DNA Watson-Crick Complementarity in Computer Science

(Thesis format: Integrated)

by

Shinnosuke Seki

Graduate Program
in
Computer Science

A thesis submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

THE UNIVERSITY OF WESTERN ONTARIO
THE SCHOOL OF GRADUATE AND POSTDOCTORAL STUDIES

CERTIFICATE OF EXAMINATION

Supervisor	Examiners
Dr. Lila Kari	Dr. Juhani Karhumäki
Supervisory Committee	Dr. Stuart Rankin
	Dr. Sheng Yu
	Dr. Lucian Ilie

The thesis by

Shinnosuke Seki

entitled:

DNA Watson-Crick Complementarity in Computer Science

is accepted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Date ______________________

Chair of the Thesis Examination Board

Abstract and Keywords

Genetic information is encoded over the four nucleotide alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ in the form of DNA helix (double-stranded structure). This structure consists of DNA strands with opposite orientation (called Watson and Crick strands), bonded via the Watson-Crick complementarity A-T, C-G. During DNA replication, each of these strands serves as a template for the reproduction of the complementary strand so as to produce two identical copies of the original DNA helix. Thus, we can say that the Watson and Crick strands are "equivalent" with respect to the information they encode.

The Watson-Crick complementarity is mathematically modeled as an antimorphic involution θ. Hence, we can formalize the above-mentioned equivalence by the equivalence between a word and its image under θ. This generalization enables us to extend the notions of periodicity and power (repetition) to those of pseudo-periodicity and pseudo-power. We call any word in $u\{u, \theta(u)\}^*$ a pseudo-power of u.

With the notion of pseudo-power, we extend two problems of significance which involve power of words, that is, the Fine and Wilf's theorem and the Lyndon-Schützenberger equation. The first theorem answers the question of how long prefix a pseudo-power of u and that of v should share to imply that u and v are pseudo-powers of some common word. Onto the length of this prefix, we provide an upper

bound $2\max(|u|,|v|)+\min(|u|,|v|)-\gcd(|u|,|v|)$, and later improve it slightly. We also investigate its lower bound by constructing words u, v which cannot be written as pseudo-powers of a common word, but some of whose pseudo-powers can share a prefix of length quite close to the upper bound.

The extended Lyndon-Schützenberger equation is of the form

$$\alpha(u,\theta(u)) = \beta(v,\theta(v))\gamma(w,\theta(w)),$$

where $\alpha(u,\theta(u)) \in \{u,\theta(u)\}^\ell$, $\beta(v,\theta(v)) \in \{v,\theta(v)\}^n$, and $\gamma(w,\theta(w)) \in \{w,\theta(w)\}^m$ for some $\ell, n, m \geq 1$. We ask the question of under what conditions on ℓ, n, m, this equation implies that $u, v, w \in \{t,\theta(t)\}^+$ for some word t. The strongest condition we obtained so far is $\ell \geq 4$, $m, n \geq 3$.

Keywords: Watson-Crick complementarity, antimorphic involution, Fine and Wilf's theorem, Lyndon-Schützenberger equation

"Simplicity is the final achievement.

After one has played a vast quantity of notes and more notes, it is simplicity that emerges as the crowning reward of art."

\- Frédéric Chopin

Co-Authorship

This thesis mainly consists of seven of my research articles published in journals, presented at conferences, and/or being under review as of August 13, 2010. All of them are co-authored by my supervisor Prof. Lila Kari, and four of them have other co-author(s).

"On a special class of primitive words" (Chapter 3) is co-authored by Dr. Elena Czeizler. She initiated this research project and I joined it. Many of the results in this paper owe much to her, but I made substantial contributions to the proof of one of its most important results "the extended Fine and Wilf's theorem" as well as to the proofs of several other combinatorial results.

"An extension of the Lyndon Schützenberger result to pseudoperiodic words" (Chapter 4) is co-authored by Dr. Elena Czeizler and Dr. Eugen Czeizler. They brought up the problem of how to solve the extended Lyndon-Schützenberger equation, and provided a concise solution for the cases when the extended Fine and Wilf's theorem is applicable. I solved the equation in the case when the theorem cannot be applied, and hence, other combinatorial arguments are required.

"Properties of pseudo-primitive words and their applications" (Chapter 5) is co-authored by Dr. Benoît Masson. All of the proofs in this paper were proved by myself, and proofread by him. Examples are totally attributed to his heavy works and an excellent example generator coded by him.

"Duplication in DNA sequences" (Chapter 8) is co-authored by Prof. Masami Ito and Mr. Zachary Kincaid. Since this work was completed more than two years ago, I cannot remember precisely which part had been done by whom. However, all the authors agree to that each of them made significant contributions to it.

Acknowledgements

First and the foremost, I wish to express my special thanks to my supervisor Prof. Lila Kari. All of the works contained in this thesis as well as the others have been accomplished by her appropriate problem presentation, constructive criticisms and discussion, words of encouragement, and engaging smile. She gave me lots of opportunities to build up my collaboration-ship, which has, in turn, provided my scientific life with quite a few valuable experiences. Aside from the scientific mentorship, she is a great conversational partner, and our discussions on philosophy, linguistics, history, music, etc. open my eyes for new worlds, and enrich my daily life further. My doctoral program has been supported financially by the Natural Sciences and Engineering Research Council of Canada Discovery Grant R2824A01 and Canada Research Chair Award to her.

Throughout the doctoral program, I have been blessed with many opportunities to collaborate with brilliant researchers. Among them, firstly, let me express my heartfelt thanks to the excellent co-authors (listed in the alphabetical order of their last names): Dr. Ehsan Chiniforooshan, Dr. Elena Czeizler, Dr. Eugen Czeizler, Prof. Mark Daley, Dr. David Doty, Prof. Oscar H. Ibarra, Prof. Masami Ito, Prof. Lila Kari, Mr. Zachary Kincaid, Prof. Satoshi Kobayashi, Prof. Kalpana Mahalingam, Dr. Benoît Masson, Prof. Peter Sosík, and Dr. Zhi Xu. The studies present in this thesis could not be accomplished if it were not for the fruitful discussion with them, their constructive comments and suggestions, and their friendship. In particular, my special thanks are due to Dr. Elena Czeizler. She initiated me to combinatorics on words, and led me to the door to the fruitful area of biologically-inspired combinatorics on words. Not to mention the two collaborative papers of ours, my further extensions of these could not have been accomplished, had it not been for her kind support and constructive comments.

Gracious mentors are also irreplaceable assets. Prof. Helmut Jürgensen spared a great deal of his precious time for our discussion, which led me to quite a few significant results. He, Prof. Masami Ito, and Prof. Satoshi Kobayashi kindly provided me with opportunities for me to give a lecture at their institutes: Universität Potsdam, Kyoto Sangyo University, and The University of Electro-Communications. I appreciate Prof. Henning Bordihn, Prof. Shikishima-Tsuji Kayoko, Dr. Peter Leupold, and Prof. Koki Abe for the fruitful discussion with them at these institutes. In the class room as well as through conference organization, I learned a lot from Prof. Sheng Yu. His class "advanced automata theory" was one of the most stimulating lecture I have ever taken. A document I assembled as its assignment became a must-read whenever I am working on deterministic context-free languages, and brought several important results between autumn 2009 and spring 2010. Writing my part of a book chapter "DNA Computing: Foundations and Implications" owes much to the thoughtful advices from Prof. Gheorghe Păun. Most of the works included in this thesis could not be completed without the valuable comments and encouragement I received from Prof. Arto Salomaa on his annual visits on our research group. Prof. Zoltán Ésik indirectly helped my works on duplication by his concise proofs. From Prof. Kathleen Hill, I received valuable comments on the mechanisms of how information processing on nucleotides works. It should also be mentioned that I am indebted to the anonymous referees who have read or are reading my papers and especially my proofs, which are apt to get lengthy, with their untiring perseverance.

I would like to take this opportunity to appreciate Prof. Lucian Ilie, Prof. Juhani Karhumäki, Prof. Stuart A. Rankin, Prof. Grzegorz Rozenberg, and Prof. Sheng Yu that they have kindly agreed to be my thesis examiners and have carefully read the thesis and given detailed comments, suggestions, and sincere encouragement, which improved the quality of this thesis noticeably.

My acknowledgements go far beyond the ones related to research. Ahead of the commencement of the Ph.D. program in Canada, there had been ambivalence and hesitation of long standing. Encouragement I had received during that period was priceless. Mentorship and friendship I have received during my Ph.D. program should be mentioned. For those, words are not enough to describe my appreciation to Ms. Naoko Aoyama, Prof. Koki Abe, Mr. Andrew J. Brown, Mr. Gang Du, Ms. Modelly Ze H, Dr. Hayato Hoshihara (Online co., ltd.), Mr. Atsushi Kijima, Prof. Satoshi Kobayashi, Prof. Yuji Kobayashi, Prof. Naoto Koyama, Dr. Thomas Margoni, Ms. Nanae Mitsuo, Mr. Tomoki Murota, Mr. Toshiaki Nagasawa, Ms. Saori Namatame, Mr. Kazuhiko Ooi, Mr. Michihiko Saeki (Online co., ltd.), Ms. Kuniyo

Saito, Mr. Masanobu Saito, late Prof. Tatsuhiko Saito, Ms. Mie Sarashina, Mr. Akio Shimazaki, Mr. Minoru Shoda (Online co., ltd.), Mr. Daichi Sugasaki, Prof. Mitsugu Suzuki, Ms. Miwa Uneme, Ms. Crystal Wong, Mr. Takashi Yabuki (Online co., ltd.), Prof. Takashi Yokomori, to name a few, and beyond anybody else to Ms. Aya Hitomi for her emotional support. Online co., ltd. also gave a generous financial support to my Ph.D. program. I also wish to express my gratitude to all the staffs at the Department of Computer Science. Above all, they relieved me from clerical or technical affairs so that I could focus on my research.

For the last but never the least, my heartfelt appreciation goes to my family. Thanks to their ready consent, I could make my mind to come to Canada to complete my Ph.D. degree. Without their physical and mental support and care, I could never have kept my enthusiasm to the research while living alone afar off my motherland. Having studied far from Tokyo for long four years, I might have caused lots of worries to them. I wish I could go back to Tokyo even for a while to stay with them after completing this thesis.

Shinnosuke Seki

To the memory of late grandfathers, Tomoe Seki and Sei-ichi Tanaka,

this little thesis which they had been waiting for forever

is most affectionately dedicated.

May their souls rest in peace.

Shinnosuke Seki

Contents

Preface

The idea of extending the notion of identity, being inspired by DNA double helix, has been originally proposed by Dr. Elena Czeizler during her stay as a postdoctoral fellow at our research group 3 years ago. At a research meeting in November 2007, she described her premiere results on the extended Fine and Wilf's theorem; it is a quite challenging task to express how deeply the author was impressed then. She illustrated how gracefully an inspiration taken from biology fuses with a well-established theory in mathematics. It was a glimpse of true enlightenment.

We quickly adopted this extended identity and a resulting extended notion of power (repetition). Our subsequent prolificacy bears eloquent testimony to how mathematically beautiful and well-defined the notion is. In fact, it was not long before we succeeded in extending another significant theory on Lyndon-Schützenberger equation. Even after she left our group, the author has kept working on this promising mine, and obtain novel results and refinements of our premiere works as well as invent useful tools for ease in access to this beautiful mathematical object. Since considerable amounts of positive results have been achieved, now the author assembles this thesis mainly based on these achievements by adding his other related works done during his Ph.D. program, which has been supervised by Prof. Lila Kari at the Department of Computer Science, the University of Western Ontario from September 2006 to August 2010 (scheduled).

This thesis is composed of two main parts except the introduction and the conclusion: Part II presents the above-mentioned main contributions in combinatorics on words, that is, a biologically-inspired extension of the notion of identity of strings and the extended Fine and Wilf's theorem and Lyndon-Schützenberger equation. Meanwhile, Part III is an ensemble of the author's works in formal language theory, which shed light rather on the dynamic information processing by modeling the mechanism as language operations such as duplication, parallel insertion and deletion.

It is most regrettable that this thesis cannot report the achievements done through fruitful collaboration-ships in the year 2010 with Dr. Ehsan Chiniforooshan, Prof. Mark Daley, Dr. David Doty, Prof. Oscar H. Ibarra, Prof. Peter Sosík, and Dr. Zhi Xu. This is purely because of its submission schedule. It does not present any results that are required for his co-authors' proposal for Ph.D. degree, either. As such, this thesis is never a "complete" collection of all his works. His website (`http://www.csd.uwo.ca/~sseki` as of August 13, 2010) is kept updated; if the readers find his works interesting and visit this website, that is more than he can dream of.

Shinnosuke Seki

during the season of fresh green leaves

at London, Ontario, Canada, 2010

List of Tables

List of Figures

List of Abbreviations, Symbols, and Nomenclature

The following is a list of symbols in heavy use with their typical usage within this thesis.

i, j, k, m, n, p, q	integers
A	automaton
L	language
X, Y	variables (unknown languages)
R	regular language
$\equiv$	equivalence relation
$\mathfrak{F}$	the set of all tuples of words
θ	antimorphic involution
gcd	the greatest common divisor
lcm	the least common multiple
max	maximum value
min	minimum value

Σ	alphabet
u, v, w, r, t	words
λ, ϵ	the empty word
$\lvert u \rvert$	the length of u
$\rho(u)$, $\sqrt{u}$	the primitive root of u
$\rho_\theta(u)$	the θ-primitive root of u
$u \wedge v$	the maximal common prefix of u and v
$\mathrm{Pref}_k(w)$	the prefix of w of length k
$\mathrm{Suff}_k(w)$	the suffix of w of length k

Σ^*	the sets of all words over Σ
Σ^+	the sets of all non-empty words over Σ
Σ^n	the sets of all non-empty words of length n over Σ
$\Sigma^{\leq n}$	the sets of all non-empty words of length at most n over Σ
$\Sigma^{\geq n}$	the sets of all non-empty words of length at least n over Σ
L^c	the complement of a language L
$A \setminus B$	the relative complement of B in A
$\mathrm{Pref}(w)$	the set of all prefixes of w
$\mathrm{Suff}(w)$	the set of all suffixes of w
$\mathrm{PPref}(w)$	the set of all proper prefixes of w
$\mathrm{PSuff}(w)$	the set of all proper suffixes of w
$\mathrm{Cp}(w)$	the set of all cyclic permutations of w
$\mathrm{L}_{\mathrm{d}}^{\theta}(w)$	the set of all proper θ-borders of w
$\mathrm{D}_{\theta}(i)$	the set of all words with i θ-borders
$\mathrm{L}_{\mathrm{cd}}^{\theta}(w)$	the set of all θ-pk-borders of w
$\mathrm{K}_{\theta}(i)$	the set of all words with i θ-pk-borders

$\heartsuit$	duplication
$\spadesuit$	repeat deletion
$L^{\heartsuit(C)}$	duplication of L controlled by C
$L^{\spadesuit(C)}$	duplication of L controlled by C
$\mathrm{Dup}(C)$	the set of all squares of words in C
$[w]_{\equiv}$	equivalence class with w as its representative
$\leftarrowtail_F$	parallel insertion based on the p-schema F
$\rightarrowtail_F$	parallel deletion based on the p-schema F
$\Sigma^*/\equiv$	the quotient set of Σ^* by $\equiv$

WK	Watson-Crick
FIN	the class of all finite languages
REG	the class of all regular languages
CFL	the class of all context-free languages
CSL	the class of all context-sensitive languages
NFA	non-deterministic finite automaton
NCM	the class of non-deterministic counter machines or that of languages accepted by such machines.
NCM(k)	the class of 1-reversal-bounded non-deterministic machines with k-counters or that of languages accepted by such machines.
NPCM	the class of non-deterministic pushdown counter machines or that of languages accepted by such machines.
D	indicator of deterministic property, which can replace N in the above notations.

Part I

Introduction

Chapter 1

Introduction

Last century has seen remarkable developments in molecular biology. Since the discovery of the double helical structure of DNA by Watson and Crick [56], several fundamental processes occurring in living organisms had been elucidated such as DNA replication, translation, transcription, and protein synthesis. The enormous progress in the understanding of DNA *in vivo* and the manipulation of DNA *in vitro* has even led to the appearance of a completely new field of research that combines molecular biology and computation: DNA computing [1].

Computation consists of two essential components: one is the method of representing information (encoding), and the other is the mechanism to manipulate the encoded information (processing). These are central to bio-computation, too. This thesis mainly focuses on the first.

The understanding of the genetic information processing mechanism has grown tremendously as exemplified before, whereas the study of the problem of how to encode information on biomolecules is in its infancy. Mechanisms of information processing in general ought to be designed so as to get the most out of the intrinsic properties of information encoding media, as being exemplified by the fact that arithmetic operations are implemented favorably on bits instead of digits because a bi-stable environment (two voltage levels) is the optimal way of encoding information electronically. Therefore, better understanding of DNA as an information encoding medium will ultimately lead us to the better way of encoding data as biomolecules.

Information encoding is a topic handled mainly in coding theory [27], which in turn relies heavily on combinatorics on words [5, 40]. Accumulated knowledge in these fields has proved to be useful in information encoding onto biomolecules, at least for DNA computing purposes. Various particularities exhibited by biological information encoding, however, prevent us from applying this knowledge *in situ* to biomolecular encoding. For instance, two representative encoding biomolecules, DNA nucleotides and proteins, consist of more than two kinds of units (not binary), and hence, known results on binary codes require some base conversion to be compatible with codes on these molecules, see, e.g., [2]. Moreover, the selective chemical bond between DNA molecules called *Watson-Crick complementarity* has to be taken into account.

The primary aim of this thesis, and hence, its major contribution, is to generalize two notions of significance in combinatorics on words, namely, *power* and *primitivity* of word, so as to reflect the particularities of DNA encoded information, specifically the Watson-Crick complementarity. These generalizations further enable us to extend two landmark results in this field, the *Fine and Wilf's theorem* [16] and *Lyndon-Schützenberger equation* [41] (Chapters 3-6). Although results reported here are credited with no more than the purely mathematical value of being generalizations of significant classical notions and theorems, continuation of research in this direction has also the potential to enrich our understanding of biomolecular information and computation.

Our secondary aim is to model three bio-operations mathematically, and analyze these models. These operations are *pseudoknot formation* (studied in Chapter 7), *duplication* (Chapter 8), and *parallel insertion/deletion* (Chapter 9).

1.1 Preliminaries

1.1.1 Preliminaries in molecular biology

This subsection contains a brief description of basic molecular biology notions of DNA structure. Keywords include nucleotide, DNA molecule in single and double strand forms, Watson-Crick complementarity, and DNA replication. An abundance of literatures exists on these topics, e.g., [3, 11, 21, 38, 55].

A *DNA* (deoxyribonucleic acid) molecule is a linear chain (*DNA single strand*) of *nucleotides* bonded by strong covalent bonds. A nucleotide consists of a sugar-phosphate unit and one of the bases (Adenine, Cytosine, Guanine, and Thymine). Since nucleotides are distinguished from each other only by their bases, the nucleotides can be symbolized by A, C, G, T.

A DNA molecule has an orientation, from the 5'-end to the 3'-end, arising from some chemical properties of nucleotides. By convention, a DNA molecule is written in the 5' to 3' orientation, that is, AGGTCT stands for 5'-AGGTCT-3'.

DNA single strands may interact with each other as follows. A can bind to T via two hydrogen bonds, while C can bind to G via three hydrogen bonds. Given two single-stranded DNA molecules w_1, w_2, if w_1 can be obtained from w_2 by *reversing* w_2 and *replacing each nucleotide with its complement nucleotide in the above-mentioned sense* (A $\rightarrow$ T, C $\rightarrow$ G, and vice versa), then w_1 and w_2 are said to be *Watson-Crick (WK-) complementary* to each other. Two complementary DNA single strands (one of which is called the Watson strand and the other is called the Crick strand; there is no biological difference between Watson strand and Crick strand) with opposite orientation can bind to each other and form a stable *DNA double strand* resembling a helical ladder (also-called *DNA double helix*). For example, AGGTCT and AGACCT are WK-complementary to each other, and can bind to each other by forming bonds between their individual bases as

5'-AGGTCT-3'
3'-TCCAGA-5' .

It is not always the case that once two WK-complementary single strands are poured into a test tube, they certainly bind to each other and form a double helix. This tends to happen at low temperatures, while at high temperatures, this is quite unlikely. The temperature at which the probability for two WK-complementary DNA single strands to form the double helix is 50% is called their melting temperature. The reason why it gets harder to keep the double helix at high temperatures is that the higher the temperature gets, the larger the amount of energy available to break the weak hydrogen bonds between bases becomes. More generally, the notion of Gibbs free energy can be defined for any given DNA structure informally[1] as the amount of energy required to break all hydrogen bonds in the structure.

The process of *DNA (semi-conservative) replication* begins with the unwinding of Watson and Crick strands of a DNA double strand. To the unwound two DNA single strands, short DNA single strands called DNA primers stick, and form partially double-stranded DNA molecules. Extending the 3'-end of primers result in two identical copies of the original double strand. This extension is accomplished by an enzyme called DNA polymerase, which adds free nucleotides to the 3'-end of newly-formed strands.

[1]The formal definition of Gibbs free energy is as:

$$G(p, T) = U + pV - TS,$$

where U is the internal energy, p is pressure, V is volume, T is the temperature, and S is the entropy. Although this formal definition will not be required in the rest of this thesis, the interested readers are referred to, e.g., [47].

1.1.2 Preliminaries in formal language theory

Next we provide the reader with basic concepts and notation in combinatorics on words and formal language theory. References [22, 24, 40, 49, 57] contain further details.

An *alphabet* is a set of letters, denoted by Σ. For example, the English language alphabet is $\{a, b, c, \ldots, y, z\}$ of size 26, the protein alphabet consists of

$$\{\mathtt{A}, \mathtt{R}, \mathtt{N}, \mathtt{D}, \mathtt{C}, \mathtt{Q}, \mathtt{E}, \mathtt{G}, \mathtt{H}, \mathtt{I}, \mathtt{L}, \mathtt{K}, \mathtt{M}, \mathtt{F}, \mathtt{P}, \mathtt{S}, \mathtt{T}, \mathtt{W}, \mathtt{Y}, \mathtt{V}\}$$

(size 20), and the alphabet of DNA molecules is $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ of size 4. A (finite) word over Σ is a sequence of letters in Σ, and the set of all such words is denoted by Σ^*. The catenation of two words $u, v \in \Sigma^*$ is denoted by $u \cdot v$ or simply by uv. The *length* of a word $w \in \Sigma^*$ is the number of letters occurring in w and is denoted by $|w|$. For instance, $|\mathtt{AGGTCT}| = 6$. The *empty word* is the word of length 0, denoted by λ (or ϵ), and we denote $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$, the set of non-empty words. A word $u \in \Sigma^*$ is called an *infix* or a *factor* of a word $w \in \Sigma^*$ if $w = xuy$ for some $x, y \in \Sigma^*$; if either x or y is not empty, then the infix is said to be *proper*. An integer $p \geq 1$ is called a *period* of a word w if $p \leq |w|$ and any of two letters of the word which are separated by $p - 1$ letters are the same. For example, the word "abcabcab" has three periods 3, 6, and 8.

A subset of Σ^* is called a *language*. A language is said to be *regular* (*context-free*) if it is accepted by a finite automaton (resp. pushdown automaton). For details of language acceptors including these two automata, and the hierarchy they form, see [24].

For a language $L \subseteq \Sigma^*$, $L^* = \{w_1 w_2 \cdots w_n \mid n \geq 0, w_1, \ldots, w_n \in L\}$. For a singleton language $\{w\}$, we use the notation w^* instead of $\{w\}^*$. A word in w^* is called a *power* of w. In particular, $w^2 (= ww)$ and $w^3 (= www)$ are called the *square* and *cube* of w, respectively. A non-empty word which is not a power of another (strictly shorter) word is said to be *primitive*. For a non-empty word $w \in \Sigma^+$, a primitive word $u \in \Sigma^+$ such that $w \in u^*$ is called the *primitive root* of w, and is denoted by $\rho(w)$. It is well-known that any non-empty word has a *unique* primitive root. Note that two words are powers of a common word if and only if their primitive roots are the same.

On Σ^*, one can define the identity function id_{Σ^*} to be the function with domain and codomain Σ^* which satisfies $\mathrm{id}_{\Sigma^*}(w) = w$ for all $w \in \Sigma^*$. A mapping $\theta : \Sigma^* \to \Sigma^*$ is called an *antimorphic involution* if both the following conditions are satisfied:

1. for any $u, v \in \Sigma^*$, $\theta(uv) = \theta(v)\theta(u)$ (antimorphism);

2. the composition of θ with itself becomes the identity, i.e., $\theta \circ \theta = \mathrm{id}_{\Sigma^*}$ (involution).

The antimorphic involution is known to be a proper model of WK-complementarity; the antimorphic and involutive properties correspond to reversing and the complementary relations A-T, C-G, respectively.

We can abstract a DNA single strand as a word over the nucleotide alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$, and define an antimorphic involution τ over this alphabet as $\tau(\mathtt{A}) = \mathtt{T}$, $\tau(\mathtt{T}) = \mathtt{A}$, $\tau(\mathtt{C}) = \mathtt{G}$, and $\tau(\mathtt{G}) = \mathtt{C}$. Then τ formalizes the Watson-Crick complementarity. In fact, τ maps a Watson strand into the corresponding Crick strand as being illustrated by $\tau(\mathtt{AGGTCT}) = \tau(\mathtt{T})\tau(\mathtt{AGGTC}) = \cdots = \tau(\mathtt{T})\tau(\mathtt{C})\tau(\mathtt{T})\tau(\mathtt{G})\tau(\mathtt{G})\tau(\mathtt{A}) = \mathtt{AGACCT}$. Therefore, τ is called the DNA involution [29].

Now let us conclude this preliminaries with the introduction of two well-known results whose extensions by taking antimorphic involutions into account will be the main subjects of the research reported in this thesis: one is the Fine and Wilf's theorem and the other is the Lyndon-Schützenberger equation.

In [16], Fine and Wilf proved that if a word has two periods p, q and is of length at least $f(p, q) = p + q - \gcd(p, q)$, then $\gcd(p, q)$ is also a period of the word, where $\gcd(p, q)$ denotes the greatest common divisor of p and q. Using the notion of power, this theorem can be restated as follows.

Theorem 1.1. *For given two words u, v, if a common prefix between a power of u and a power of v is of length $f(|u|, |v|)$, then u and v are powers of the same word, that is, they share a primitive root.*

For *any* two integers p, q, a proof of Theorem 1.1 found in [5] gives a construction of two words u, v of respective lengths p, q with $\rho(u) \neq \rho(v)$ such that a power of u

and a power of v can share a common prefix of length $f(p,q)-1$. According to the terminology used by Constantinescu and Ilie in [6], the bound f given in Theorem 1.2 is said to be *strongly optimal* for the Fine and Wilf's theorem (the formal definition of strong optimality will be introduced in the context of our extension of Fine and Wilf's theorem in Section 1.2).

The *Lyndon-Schützenberger's equation* is the generic term of the equation of the form:

$$u^\ell = v^n w^m, \tag{1.1}$$

where $u, v, w \in \Sigma^+$ and $\ell, n, m \geq 1$. Lyndon and Schützenberger proposed and investigated equations of this form [41]. A question arises of under what conditions on ℓ, n, m, this equation implies that u, v, w are powers of the same word. They answered this question as: if all of ℓ, n, m are at least 2, then Eq. (1.1) implies that $u, v, w \in t^+$ for some word $t \in \Sigma^+$.

1.2 Watson-Crick complementarity and combinatorics on words

The main goal of this thesis is to study the implications of taking into account properties of DNA encoded information, and in particular, WK-complementarity, on notions and results in combinatorics on words and formal language theory. In particular, one of the main topics of our research has been to extend the Fine and Wilf's theorem and Lyndon-Schützenberger equation. Here we firstly provide formal definitions of the problems which we will address. Since these are two of the most important results in combinatorics on words, they have been the subject of a considerable number of studies since their proposal in 1960s. For details, see the introductions of Chapters 3-6.

A common notion in the Fine and Wilf's theorem and Lyndon-Schützenberger equation is the notion of power. The deeper-lying concept is, however, the *identity* because the power can be interpreted as a repetition of identical copies of a word.

In Section 1.1.1, we explained the mechanism of DNA replication through which a DNA double strand generates two copies of its own by using its Watson and Crick strands as templates. This means that the Watson and Crick strands contain the same information for theoretical but also for practical purposes. To put it another way, they are informationally equivalent. We generalize this information equivalence, for an arbitrary alphabet and given antimorphic involution θ, by regarding a word w and its complement $\theta(w)$ equivalent.

Taking this equivalence into account, we can naturally extend the definition of power as a repetition of multiple words which are equivalent to each other according to a predefined equivalence relation between words. If we define the relation as "u is equivalent to v if $u = v$," then this new definition is identical to the original. In this thesis, we employ the equivalence defined as "for a given antimorphic involution θ,

u is equivalent to v if either $u = v$ or $u = \theta(v)$." With this equivalence in mind, it makes sense to call elements of the set $w\{w, \theta(w)\}^*$ *θ-powers of w*. For instance, for the DNA involution τ and a DNA molecule `AC`, both `ACAC` and `ACGTGT` are τ-powers of `AC` because they are in $\texttt{AC}\{\texttt{AC}, \tau(\texttt{AC})\}^*$. We call the elements of $\{uu, u\theta(u)\}$ as *θ-squares of u*.

As the notion of primitivity is based on the notion of power, we can define the concept of θ-primitivity based on the notion of θ-power. We say that a non-empty word $w \in \Sigma^+$ is *θ-primitive* if $w \in t\{t, \theta(t)\}^*$ implies $w = t$. For a non-empty word $u \in \Sigma^+$, its *θ-primitive root* is defined to be a θ-primitive word $t \in \Sigma^+$ such that $u \in t\{t, \theta(t)\}^*$. One of the main results of this thesis asserts the uniqueness of θ-primitive root for any given antimorphic involution θ (Corollary 3.15 in Chapter 3). Therefore, we can say that two words are θ-powers of a common word if and only if they share their θ-primitive roots. Furthermore, we can denote the unique primitive root of a word w by $\rho_\theta(w)$.

The definitions of θ-power and θ-primitive root are followed by a question which is analogous to the one asked in the context of Fine and Wilf's theorem as follows: For $u, v \in \Sigma^+$, when a θ-power of u and a θ-power of v share a common prefix of some length, how long should this common prefix be to force u and v to share their θ-primitive root? More formally, this problem is formalized as follows:

Problem 1.1. Find a function $f : \mathbb{N}^2 \to \mathbb{N}$ such that for an antimorphic involution θ and two words u, v, if a θ-power of u and a θ-power of v share a common prefix of length $f(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$.

The theorem which we will call *extended Fine and Wilf's theorem* is rather a collection of results which prove that some specific function is an answer to Problem 1.1. Such a function is called a *bound for the extended Fine and Wilf's theorem*, or simply, a *bound*.

Using terminologies introduced by Constantinescu and Ilie in [6], we define a few criteria of optimality for this bound. A function f is called a *good bound for* P if it is an answer to Problem 1.1. A function f which is good for (p, q) for any integers p, q is called a *good bound*. With the original result proved by Fine and Wilf in mind, one can imagine that if f returns a big integer relative to $|u|$ and $|v|$, i.e., $f(|u|, |v|) \gg |u|, |v|$, then it is likely for f to be a good bound. A more challenging problem is, therefore, to make the value of f as low as possible, or in other words, to find the *optimal* bound.

For given integers p, q, a function f is said to be *optimal for* (p, q) if this function is good for (p, q), while the function f', which is defined as $f'(n, m) = f(n, m) - 1$ for any integers n, m, is not. A function f is said to be *weakly optimal* if there exist integers p, q such that f is optimal for (p, q). On the contrary, a function f is said to be *strongly optimal* if f is optimal for any integers (p, q). Then the corresponding two problems arise.

Problem 1.2. Provide a weakly optimal bound for the extended Fine and Wilf's theorem.

Problem 1.3. Provide a strongly optimal bound for the extended Fine and Wilf's theorem.

Trivially Problem 1.3 is more difficult than Problem 1.2. In fact, Problem 1.3 will remain unsettled even in this thesis. Hence, we will examine the following problem instead.

Problem 1.4. Given a bound, characterize all pairs of integers for which the bound is optimal.

The notions of θ-power and θ-primitive root also enable us to generalize the Lyndon-Schützenberger equation as follows: for $u, v, w \in \Sigma^+$ and positive integers $\ell, n, m \geq 1$,

$$u_1 u_2 \cdots u_\ell = v_1 v_2 \cdots v_n w_1 w_2 \cdots w_m, \tag{1.2}$$

where $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. Then we examine the following problem.

Problem 1.5. Given Eq. (1.2), find conditions on ℓ, n, m under which this equation implies the existence of a word $t \in \Sigma^+$ satisfying $u, v, w \in \{t, \theta(t)\}^*$.

1.3 Contributions

1.3.1 Main contributions

The two main theorems in this thesis are the following. For the first theorem, let us define for integers p, q with $p > q$ the functions:

$$
\begin{aligned}
b(p,q) &= 2\max(p,q) + \min(p,q) - \gcd(p,q), && (1.3)\\
b'(p,q) &= \begin{cases} b(p,q) & \text{if } q = \gcd(p,q), \\ b(p,q) - \lfloor \gcd(p,q)/2 \rfloor & \text{otherwise.} \end{cases} && (1.4)
\end{aligned}
$$

Theorem 1.2 (extended Fine and Wilf's theorem [8, 34]). *Let $u, v \in \Sigma^+$. If a θ-power of u and a θ-power of v share the common prefix of length $b'(|u|, |v|)$, then $u, v \in t\{t, \theta(t)\}^*$ for some word $t \in \Sigma^+$.*

Theorem 1.3 (on the extended Lyndon-Schützenberger equation [7, 33]). *An equation of the form (1.2) implies $u, v, w \in \{t, \theta(t)\}^+$ for some word $t \in \Sigma^+$ if $\ell \geq 4$ and $n, m \geq 3$.*

On the other hand, if one of ℓ, n, m is at most 2, then there exist $u, v, w \in \Sigma^+$ and an antimorphic involution θ which satisfy the following two conditions:

1. *there does not exist any word $t \in \Sigma^+$ such that $u, v, w \in \{t, \theta(t)\}^+$;*
2. *there exist $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$ for which Eq. (1.2) holds.*

These theorems will be proved in the four chapters of Part II. Chapters 3 and 5 prove Theorem 1.2, while Chapters 4 and 6 investigate the extended Lyndon-Schützenberger equation and prove Theorem 1.3.

Among them, the most significant chapter is Chapter 3. This is because it introduces the notions of θ-power and θ-primitivity of words and investigates their various algebraic properties, which turn out to be crucial for the further investigation in the other three chapters. These essential properties include:

1. the θ-primitive root of a word is unique (Corollary 3.15);
2. circular permutation does not preserve the property of θ-primitivity[2] (Proposition 3.12);
3. a θ-primitive word u can be a proper infix of a θ-square of u or of $\theta(u)$.

Property 2 contrasts with the closure property of the set of primitive words under circular permutation. Property 3 can be rephrased as: there may exist an antimorphic involution θ and a θ-primitive word u and words $u_1, u_2, u_3 \in \{u, \theta(u)\}$ such that $xu_1y = u_2u_3$ holds with some non-empty words $x, y \in \Sigma^+$. For instance, for a τ-primitive word $u = \mathtt{CGATAT}$, $u^2 = \mathtt{CG} \cdot \mathtt{ATATCG} \cdot \mathtt{ATAT} = \mathtt{CG} \cdot \tau(u) \cdot \mathtt{ATAT}$. By contrast, a primitive word can *never* be a proper infix of its square. It is these two properties of primitive words that make it possible to prove the classical Fine and

[2] Circular permutation is a composition of arbitrary number of circular shift. For example, from ATGC, we can obtain TGCA, GCAT, and CATG as well as itself by circular permutation.

Wilf's theorem and many of its variants using elegant methods based on number theory (e.g., see [5] and the references cited in Chapters 3 and 5). Due to the above-mentioned contrasts, we can no longer expect the pure number-theoretic argument to prove the extended Fine and Wilf's theorem, and hence, some groundwork facts on combinatorics on θ-primitive words had to be established, with Theorems 1.2 and 1.3 being the ultimate goals.

In Chapter 5, we prove that b' given in Eq. (1.4) works as a weakly-optimal but not strongly-optimal bound for the extended Fine and Wilf's theorem. Unlike the proof for the goodness of b (given in Eq. (1.3)), the proof for goodness of b' (Theorem 5.22) constructs all words u, v and common prefixes between θ-powers of u, v and observes that such common prefixes are strictly shorter than $b'(|u|, |v|)$ if $\rho_\theta(u) \neq \rho_\theta(v)$. In this way, we answer Problem 1.4 for the bound b' (Corollary 5.23).

One of the most intensively-studied topics appearing throughout Part II is language equations on θ-primitive words (the extended Lyndon-Schützenberger equation is included). Solving the extended Lyndon-Schützenberger equation amounts to solving a set of more tractable language equations. Language equations of our interest on the θ-primitive word include the following: For a θ-primitive word u,

$u_1, u_2, u_3, \ldots, \in \{u, \theta(u)\}$, and *non-empty* words $x, y \in \Sigma^+$ with $|x| = |y|$,

$$u_1 x = y u_2 \tag{1.5}$$

$$u_1 u_2 x = y u_3 u_4 \tag{1.6}$$

$$u_1 u_2 \cdots u_n x = y u_{n+1} \cdots u_{2n} \text{ for } n \geq 3 \tag{1.7}$$

$$u_1 u_2 = x u_3 y \tag{1.8}$$

$$u_1 u_2 u_3 = x u_4 u_5 y \tag{1.9}$$

$$u_1 u_2 \cdots u_m = x u_{m+1} \cdots u_{2m-1} y \text{ for } m \geq 4. \tag{1.10}$$

Among these, we characterize the equations of the first three forms completely in this thesis (Table 4.1 in Chapter 4). On the contrary, the fourth one can hold with non-empty x and y, as exemplified when we explained Property 3. Using the example, one can easily observe that the equations of both the fifth and the last forms can hold. Nevertheless, we prove a theorem (Theorem 6.12 in Chapter 6), which states that if $u_4 \neq u_5$, then equations of the fifth form cannot hold as long as both x and y are assumed not to be empty. As a corollary, if the equation of the last form holds with x, y being non-empty, then $u_{m+1} = u_{m+2} = \cdots = u_{2m-1}$ must hold. We view these results as equally significant to Theorems 1.2 and 1.3.

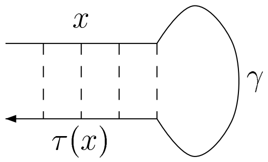

Figure 1.1: A hairpin which the word $x\gamma\tau(x)$ may form, where τ is the DNA involution.

1.3.2 Other contributions

Having abstracted the topics of primary contributions of this thesis on biological information encoding, the "static aspect" of biological computation, now let us briefly introduce the topics of our secondary contributions, which are on biological information processing, the "dynamic" aspect of biological computation[3].

Many of results in combinatorics on words have been proved useful to design sets of DNA codewords on which some bio-operations are favored, and the others are either not favored or inhibited. Better understanding of bio-operations and their properties would complement and expand the applicability of these results further. Among the most important are the computational power of bio-operations, closure properties of language classes under these operations, and language equations involving bio-operations.

Let us consider the problem of designing a codeword set whose elements cannot

[3]The usage of terminologies "static" and "dynamic" in this context is due to Luca Cardelli [4].

form any *hairpins* (see Figure 1.1). Hairpin formation of DNA single strands turned out to obstruct computational process of Adleman's first experiment of DNA computing [1] (for details of DNA computing, see Chapter 2), and this fact has driven forward the research on the hairpin-freeness design problem [26, 30, 35, 36, 45]. This problem is in more general context formulated as *negative design problem* [45, 50] (how to design strands that avoid certain bonds), and this problem has been intensively investigated on various inter- and intra-molecular interactions [2, 12, 13, 17, 19, 20, 23, 32, 39, 43]. Structure formation is one of the most basic bio-operations. By forming a structure, some part of a DNA molecule is exposed and becomes "accessible" for other molecules, whereas the other parts are "hidden". As such, it becomes possible to proceed with only intended chains of reactions to achieve various computational purposes. Reflecting the biological significance of hairpins, a large body of literature is available on hairpin-related operations such as hairpin completion and reduction [42], hairpin inversion [9, 14], control of hairpin opening for DNA memory access [28, 52, 53, 54].

In Chapter 7, we investigate pseudoknots (see Figure 1.2). Pseudoknot formation enables transfer-messenger RNA molecules (tmRNA) to change their conformation, and as such, act both as transfer RNA and as messenger RNA [15]. The pseudoknot illustrated in Figure 1.2 (Right) is of the simplest and hence prevailing form called H-type, An H-type pseudoknot is formalized as a word $v_1xv_2yv_3\tau(x)v_4\tau(y)v_5$. This formalization clarifies the *crossing-dependency*, which characterizes pseudo-

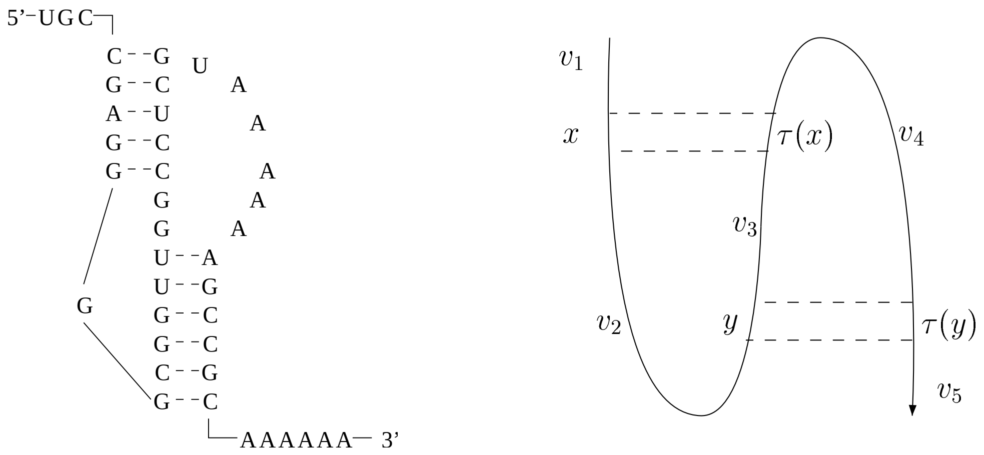

Figure 1.2: *Left:* A pseudoknot found in *E. Coli* transfer-messenger-RNA. (From Rfam [18]). *Right:* A depiction of a string modelling the pseudoknot in *Left*, as a word $v_1xv_2yv_3\tau(x)v_4\tau(y)v_5$ for the DNA involution τ. Here, $v_1 = \mathtt{UGC}$, $x = \mathtt{CGAGG}$, $v_2 = \mathtt{G}$, $y = \mathtt{GCGGUU}$, $v_3 = \mathtt{GG}$, $v_4 = \mathtt{UAAAAA}$, and $v_5 = \mathtt{AAAAAA}$.

knots. Due to the crossing-dependency, modeling pseudoknots requires strong computational power, equivalent to the context-sensitive language on the Chomsky hierarchy [24]; in contrast, hairpin can be modeled by a context-free language. As a result, fewer studies have been done on pseudoknots compared to on hairpins (see, e.g., [10, 25, 37, 44, 48, 51]). In Chapter 7, we first formalize H-type pseudoknots as θ-pseudoknot-bordered words. A word w is said to be θ-pseudoknot-bordered if $w = xy\alpha = \beta\theta(x)\theta(y)$ holds for some $x, y, \alpha, \beta \in \Sigma^*$ with $xy \neq \lambda$. This is a proper generalization of involutive borderedness of words proposed by Kari and Mahalingam [31]. Then we focus on some of the properties of pseudoknot-bordered words such as crossing-dependency structures on words.

In Chapters 8 and 9, we focus rather on other operations, often referred to as "errors". Two kinds of typical errors "duplication," and "insertion/deletion," are addressed. During DNA replication, DNA polymerase reads a template DNA single strand, and synthesizes the complementary strand of the template. However, DNA polymerase might skip and fall back, reattaching to a position that has been copied already. This type of error leads to a repetition of DNA segments during copying, and is called *duplication*. Gene duplication sometimes has life-threatening effects on organisms [46].

As a formal language operation, duplication is defined as the following unary operation:

$$u^{\heartsuit} = \{xyyz \mid u = xyz \text{ for some } x, z \in \Sigma^* \text{ and } y \in \Sigma^+\}.$$

Its dual operation, termed *repeat-deletion* $\spadesuit$, is defined as:

$$u^{\spadesuit} = \{xyz \mid u = xyyz \text{ for some } x, z \in \Sigma^* \text{ and } y \in \Sigma^+\}.$$

We can further endow duplication with a control set C such that y, the duplicated infix, must be in C, and define the *controlled duplication*. The introduction of control set is motivated by the fact that the class of regular languages[4] is not closed under uncontrolled ($C = \Sigma^*$) duplication (Proposition 8.1), whereas it is closed under repeat-deletion (Proposition 8.6). Then the following problem is examined.

[4]A language is said to be *regular* if it is accepted by some finite automaton [24].

Problem 1.6. Find a condition on C such that the class of regular languages is closed under $\heartsuit(C)$.

Chapter 8 centers around proposing several conditions on the control set C and verifying that they preserve the regularity of languages.

Insertion and deletion as well as substitution are the basic processes that induce genetic mutations. The observation that insertion (deletion) highly probably occurs at multiple points on a DNA molecule independently of each other and at the same time, makes it natural to consider parallel versions of these operations as parallel insertion and deletion. We may further endow the parallel insertion and deletion with context-sensitivity, so that the place where insertion (deletion) can occur is specified by the surrounding contexts. In Chapter 9, we propose a general framework of contextual parallel insertion and deletion called *p-schema-based insertion and deletion*, or more simply *p-schema insertion and deletion*. A *p-schema* is a set of tuples of words in Σ^*. Given a p-schema F, we define the *parallel insertion based on* F, denoted by $\longleftarrow_F$, as: for a word $u \in \Sigma^*$ and a language $L \subseteq \Sigma^*$,

$$u \longleftarrow_F L = \bigcup_{\substack{n \geq 1, u = u_1 \cdots u_n, \\ (u_1, \ldots, u_n) \in F}} u_1 L u_2 L \cdots u_{n-1} L u_n.$$

In a similar manner, for a given p-schema G, we define the *parallel deletion based* *on* G, denoted by $\rightarrowtail_G$, as: for a word $w \in \Sigma^*$ and a language $L \subseteq \Sigma^*$,

$$\begin{aligned} w \rightarrowtail_G L &= \{u_1 \cdots u_n \mid n \geq 1, x_1, \ldots, x_{n-1} \in L, \\ &\qquad (u_1, \ldots, u_n) \in G, w = u_1 x_1 u_2 x_2 \cdots u_{n-1} x_{n-1} u_n\}, \end{aligned}$$

We also consider language equations of the forms $X \leftarrowtail_F L_2 = L_3$, $L_1 \leftarrowtail_X L_2 = L_3$, $L_1 \leftarrowtail_F X = L_3$, and their deletion variants, where L_1, L_2, L_3, F are given and X is an unknown variable. Two-variables equations like $X \leftarrowtail_F Y = L_3$ are also of interest. We strictly distinguish the equations which are solvable from the ones the existence of whose solution is undecidable. For solvable equations, we propose algorithms to solve them.

1.3.3 Remarks about contributions

The primary way in which molecular biology advances the developments of computer science has been to model molecular mechanisms occurring in living organisms as computational operations or computational paradigms, and then examine the computational power of these models. From this point of view, we can say that the most important contribution which this thesis makes to computer science is to elucidate the particularities in biological computation that enable us to meaningfully expand and generalize notions and results in computer science. Among the most illustrative is the informational equivalence induced by WK-complementarity. This equivalence makes it possible to extend various notions, such as power and primitivity of words, in which the identity plays an important role by replacing the identity with the equivalence.

1.4 Thesis organization

This thesis is organized as follows:

Part II contains all the main contributions of this thesis, which consists of four chapters.

The first chapter (Chapter 3) introduces the θ-primitivity, the most significant notion in this thesis, in its Section 3.2. Then in Section 3.3, we prove some basic properties of θ-primitive words. The problem setting of the extended Fine and Wilf's theorem is formalized there, and crude bounds are proposed. Sections 3.4 and 3.5 discuss language equations which involve two words u, v and an antimorphic involution θ, and address the problem of what equations force u, v to share their θ-primitive roots. The extended Fine and Wilf's theorem in its first version, which is slightly weaker than Theorem 1.2, is proved in Section 3.6.

The second chapter (Chapter 4) formalizes the Lyndon-Schützenberger equation. In Section 4.3, we give a complete characterization of the language equations of the forms Eq. (1.5), (1.6), and (1.7). This characterization makes it possible to solve the Lyndon-Schützenberger equation positively under the condition $\ell \geq 5$ and $n, m \geq 3$

in Section 4.4.

In the latter half of Part II, we strengthen the results obtained in Chapters 3 and 4 so as to prove Theorems 1.2 and 1.3. Chapter 5 is devoted to the extended Fine and Wilf's theorem. In Section 5.3, the goodness of the bound b' given in Eq. (1.4) is verified (Theorem 5.22) and its weak and strong optimality is discussed (Corollary 5.23). A relation between the optimality of b' and Sturmian words is discussed in Section 5.4. In contrast, Chapter 6 aims at proving Theorem 1.3. We firstly advance our knowledge of Properties 2 and 3 by developing various useful lemmas in Section 6.3. Mainly based on these tools, in Section 6.4, we prove Theorem 1.3, which is stronger than the positive result proved in Chapter 4 on the extended Lyndon-Schützenberger equation, in a concise manner.

Part III consists of three chapters. The first chapter (Chapter 7) defines the notion of θ-pseudoknot-bordered words as a model of H-type pseudoknots. We investigate, e.g., the question of whether concatenation, the most basic (bio-)operation, preserves the property of θ-pseudoknot-borderedness in Section 7.4.

The second chapter (Chapter 8) deals with duplication and repeat-deletion. Section 8.3 proves closure properties of language classes in Chomsky hierarchy under these operations. In Section 8.4, we solve language equations of the form $X^{\heartsuit} = L$ and $X^{\spadesuit} = L$, where L is a given language and X is unknown. Section 8.5 defines controlled duplication, and Section 8.6 analyzes Problem 1.6. In Section 8.7, we briefly discuss relationship between duplication and the primitivity of words.

The final chapter (Chapter 9) proposes the framework of p-schema insertion and deletion in Section 9.3. Closure properties of classes of languages recognized by counter machines under p-schema insertion/deletion are investigated in details in Section 9.4; those of regular languages and context-free languages follow them as corollaries. Section 9.5 is devoted to the research on the language equations involving p-schema insertion and deletion.

Bibliography

[1] L. M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187):1021–1024, November 1994.

[2] M. Arita and S. Kobayashi. DNA sequence design using templates. *New Generation Computing*, 20:263–277, 2002.

[3] C. Calladine and H. Drew. *Understanding DNA: the Molecule and How It Works*. Academic Press, 2 edition, 1997.

[4] L. Cardelli. Personal communication.

[5] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 329–438. Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[6] S. Constantinescu and L. Ilie. Generalized Fine and Wilf's theorem for arbitrary number of periods. *Theoretical Computer Science*, 339(1):49–60, 2005.

[7] E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schützenberger result to pseudoperiodic words. In V. Diekert and D. Nowotka, editors, *Proc. DLT09*, volume 5583 of *Lecture Notes in Computer Science*, pages 183–194, Berlin, 2009. Springer-Verlag.

[8] E. Czeizler, L. Kari, and S. Seki. On a special class of primitive words. *Theoretical Computer Science*, 411(3):617–630, 2010.

[9] M. Daley, O. Ibarra, and L. Kari. Closure and decidability properties of some language classes with respect to ciliate bio-operations. *Theoretical Computer Science*, 306:19–38, 2003.

[10] R. Dirks and N. Pierce. An algorithm for computing nucleic acid base-pairing probabilities including pseudoknots. *Journal of Computational Chemistry*, 25:1295–1304, 2004.

[11] K. Drlica. *Understanding DNA and Gene Cloning: A Guide for the Curious*. Wiley and Sons, 1996.

[12] A. Dyachkov, A. Macula, W. Pogozelski, T. Renz, V. Rykov, and D. Torney. New t-gap insertion-deletion-like metrics for DNA hybridization thermodynamic modeling. *Journal of Computational Biology*, 13(4):866–881, 2006.

[13] A. Dyachkov, A. Macula, V. Rykov, and V. Ufimtsev. DNA codes based on stem similarities between DNA sequences. In M. Garzon and H. Yan, editors, *DNA Computing 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 146–151. Springer, 2008.

[14] A. Ehrenfeucht, T. Harju, I. Petre, D. Prescott, and G. Rozenberg. *Computation in Living Cells: Gene Assembly in Ciliates*. Springer, 2004.

[15] B. Felden, H. Himeno, A. Muto, J. P. McCutcheon, J. F. Atkins, and R. F. Gesteland. Probing the structure of the escherichia coli 10Sa RNA (tmRNA). *RNA*, 3(1):89–103, 1997.

[16] N. J. Fine and H. S. Wilf. Uniqueness theorem for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, February 1965.

[17] A. G. Frutos, Q. Liu, A. J. Thiel, A. M. Sanner, A. E. Condon, L. M. Smith, and R. M. Corn. Demonstration of a word design strategy for DNA computing on surfaces. *Nucleic Acids Research*, 25(23):4748–4757, 1997.

[18] S. G.-Jones, S. Moxon, M. Marshall, A. Khanna, S. R. Eddy, and A. Bateman. Rfam: Annotating non-coding RNAs in complete genomes. *Nucleic Acids Research*, 33:D121–D124, 2005.

[19] M. Garzon, P. Neathery, R. Deaton, R. Murphy, D. Franceschetti, and S. Stevens Jr. A new metric for DNA computing. In J. Koza, K. Deb, M. Dorigo, D. Vogel, M. Garzon, H. Iba, and R. Riolo, editors, *Genetic Programming 1997*, pages 479–490. Morgan Kaufmann, 1997.

[20] M. Garzon, V. Phan, S. Roy, and A. Neel. In search of optimal codes for DNA computing. In C. Mao and T. Yokomori, editors, *DNA Computing 12*, volume 4287 of *Lecture Notes in Computer Science*, pages 143–156. Springer, 2006.

[21] L. Gonick and M. Wheelis. *The Cartoon Guide to Genetics*. Collins, updated edition, 1991.

[22] M. A. Harrison. *Introduction to Formal Language Theory*. Addison-Wesley, 1978.

[23] T. Head. Relativised code concepts and multi-tube DNA dictionaries. In C. Calude and G. Păun, editors, *Finite Versus Infinite: Contributions to an Eternal Dilemma*, pages 175–186. Springer, 2000.

[24] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, 1979.

[25] H. Jabbari, A. Condon, A. Pop, C. Pop, and Y. Zhao. HFold: RNA pseudoknotted secondary structure prediction using hierarchical folding. In *WABI 2007*, volume 4645 of *Lecture Notes in Computer Science*, pages 323–334. Springer, 2007.

[26] N. Jonoska and K. Mahalingam. Languages of DNA based code words. In J. Chen and J. Reif, editors, *DNA Computing 9*, volume 2943 of *Lecture Notes in Computer Science*, pages 61–73. Springer, 2004.

[27] H. Jürgensen and S. Konstantinidis. Codes. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 511–608. Springer-Verlag, 1997.

[28] A. Kameda, M. Yamamoto, A. Ohuchi, S. Yaegashi, and M. Hagiya. Unravel four hairpins! *Natural Computing*, 7:287–298, 2008.

[29] L. Kari, R. Kitto, and G. Thierrin. Codes, involutions and dna encoding. In W. Brauer, H. Ehrig, J. Karhumäki, and A. Salomaa, editors, *Formal and Natural Computing*, volume 2300 of *Lecture Notes in Computer Science*, pages 376–393. Springer-Verlag, Berlin, 2002.

[30] L. Kari, S. Konstantinidis, and P. Sosík. On properties of bond-free DNA languages. *Theoretical Computer Science*, 334(1-3):131–159, 2005.

[31] L. Kari and K. Mahalingam. Involutively bordered words. *International Journal of Foundations of Computer Science*, 18(5):1089–1106, 2007.

[32] L. Kari, K. Mahalingam, and G. Thierrin. The syntactic monoid of hairpin-free languages. *Acta Informatica*, 44:153–166, 2007.

[33] L. Kari, B. Masson, and S. Seki. Properties of pseudo-primitive words and their applications. Submitted, available at `http://hal.archives-ouvertes.fr/hal-00458695/fr/`, 2009.

[34] L. Kari and S. Seki. An improved bound for an extension of Fine and Wilf theorem. submitted, 2009.

[35] A. Kijima and S. Kobayashi. Efficient algorithm for testing structure freeness of finite set of biomolecular sequences. In A. Carbone and N. Pierce, editors, *DNA Computing 11*, volume 3892 of *Lecture Notes in Computer Science*, pages 171–180. Springer, 2006.

[36] S. Kobayashi. Testing structure freeness of regular sets of biomolecular sequences (extended abstract). In C. Ferreti, G. Mauri, and C. Zandron, editors, *DNA 10*, volume 3384, pages 192–201. Springer, 2005.

[37] S. Kobayashi and S. Seki. An efficient multiple alignment method for RNA secondary structures including pseudoknots. In *Prpc. 2nd International Workshop on Natural Computing (IWNC)*, pages 179–188, 2009.

[38] B. Lewin. *Genes IX*. Johns and Bartlett Publishers, 2007.

[39] M. Li, H. J. Lee, A. Condon, and R. M. Corn. DNA word design strategy for creating sets of non-interacting oligonucleotides for DNA microarrays. *Langmuir*, 18:805–812, 2002.

[40] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications*. Addison-Wesley, 1983.

[41] R. C. Lyndon and M. P. Schützenberger. The equation $a^m = b^n c^p$ in a free group. *Michigan Mathematical Journal*, 9:289–298, 1962.

[42] F. Manea, V. Mitrana, and T. Yokomori. Two complementary operations inspired by the DNA hairpin formations: Completion and reduction. *Theoretical Computer Science*, 410(4-5):417–425, 2009.

[43] A. Marathe, A. Condon, and R. Corn. On combinatorial DNA word design. *Journal of Computational Biology*, 8(3):201–219, 2001.

[44] H. Matsui, K. Sato, and Y. Sakakibara. Pair stochastic tree adjoining grammars for aligning and predicting pseudoknot RNA structures. In *2004 IEEE Computational Systems Bioinformatics Conference*, pages 1–11, 2004.

[45] G. Mauri and C. Ferretti. Word design for molecular computing: A survey. In J. Chen and J. Reif, editors, *DNA Computing 9*, volume 2943 of *Lecture Notes in Computer Science*, pages 37–46. Springer, 2004.

[46] S. Mirkin. Expandable DNA repeats and human disease. *Nature*, 447(7147):932–940, 2007.

[47] P. Perrot. *A to Z of Thermodynamics*. Oxford University Press, 1998.

[48] B. Rastegari and A. Condon. Parsing nucleic acid pseudoknotted seconday structure: Algorithm and applications. *Journal of Computational Biology*, 14(1):16–32, 2007.

[49] G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages*, volume 1. Springer-Verlag, Berlin Heidelberg, 1997.

[50] J. Sager and D. Stefanovic. Designing nucleotide sequences for computation: A survey of constraints. In A. Carbone and N. Pierce, editors, *DNA Computing 11*, volume 3892 of *Lecture Notes in Computer Science*. Springer, 2006.

[51] S. Seki and S. Kobayashi. A grammatical approach to the alignment of structure-annotated strings. *IEICE Transactions on Information and Systems*, E88-D(12):2727–2737, 2005.

[52] J. S. Shin and N. A. Pierce. Rewritable memory by controllable nanopatterning of DNA. *Nanoletters*, 4:905–909, 2004.

[53] M. Takinoue and A. Suyama. Molecular reactions for a molecular memory based on hairpin DNA. *Chem-Bio Informatics Journal*, 4:93–100, 2004.

[54] M. Takinoue and A. Suyama. Hairpin-DNA memory using molecular addressing. *Small*, 2(11):1244–1247, 2006.

[55] P. Turner, A. McLenna, A. Bates, and M. White. *Instant Notes in Molecular Biology*. Garland Publishing Inc., 2 edition, 2000.

[56] J. D. Watson and F. H. C. Crick. A structure for deoxyribose nucleic acid. *Nature*, 171:737–738, 1953.

[57] S. S. Yu. *Languages and Codes*. Tsang Hai Book Company Co., Taichung, Taiwan, 2005.

Chapter 2

DNA computing

As outlined so far, the contributions of this thesis are purely mathematical. Nevertheless, it is the bio-molecular information encoding and processing that have motivated the research reported in this thesis, as emphasized throughout the previous chapter. In this chapter, we take a look back at the history of DNA computing, or more generally, of molecular computation. This will augment the mathematical contribution of this thesis with potential applications to experimental fields such as DNA computing or molecular biology.

2.1 Preamble

Among the significant scientific and technological achievements of the last century, those in physics and biology deserve special mention: nuclear power, quantum mechanics, space exploration, and molecular biology. They necessitated massive computation capability, which was well beyond the unaided human computation power. The compelling nature of this need gave information technology impetus to grow explosively. As a result, throughout the latter half of the century, information technology and the computer industry have kept expanding tremendously.

The limits of miniaturization at atomic levels may eventually challenge this rapid growth. In order to overcome these limits, novel computational paradigms have been proposed, which make use of knowledge in physics and biology: *quantum computing* and *molecular computing*. (For further details of quantum computing, the reader is referred to [24].) It comes as something of a surprise how early basic concepts of the paradigm of molecular computing appeared. In 1959, Feynman introduced the notion of nano-machine for the first time in his talk at Caltech [15]. John von Neumann, the inventor of cellular automata, regarded the mechanism of self-reproducing as a common feature between biological organisms and computers [56]. An idea of cell molecular computer was discussed by Vaintsvaig and Liberman in *Biofizika* in 1973 [55]. The capability of macromolecules such as proteins to process information has been investigated by Conrad since 1974 [8, 9]. The biological significance of the information theory by Claude Shannon, the father of information theory, has been recently pointed out by Schneider [49].

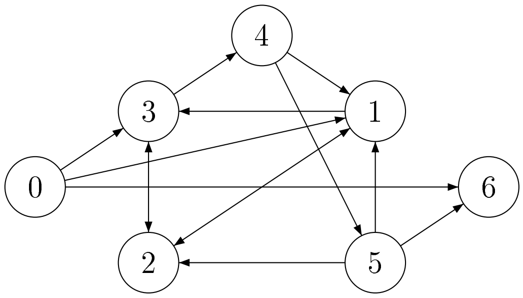

Figure 2.1: The seven-vertex instance of the Hamiltonian Path Problem solved by Adleman's experiment in 1994. It contains a Hamiltonian path $0 \to 1 \to 2 \to 3 \to 4 \to 5 \to 6$.

2.2 DNA computing: The first experiment

Being supported by conspicuous developments in molecular biology such as the discovery of DNA double helix structure by Watson and Crick [58], Polymerase Chain Reaction (PCR) [43], and gel electrophoresis, all of these early attempts have finally come to fruition in the breakthrough experiment by Leonard Adleman in 1994, who first succeeded in having DNA molecules solve an instance of an NP-complete problem, namely the directed Hamiltonian Path Problem (HPP) in a polynomial time [2]. HPP asks whether a given directed graph, with its start node v_s and end node v_e being specified, has a path from v_s to v_e which visits all other nodes exactly once[1]. The instance of HPP (a directed graph $G = (V, E)$) actually solved in his experiment is illustrated in Figure 2.1. The algorithm mainly consists of two phases:

[1]HPP is a well-known NP-complete problem; indeed it is one of the Karp's 21 NP-complete problems [31]. Thus, at least in theory, any NP-complete problem can be solved by DNA computing in a polynomial time at the cost of exponential space needed for the computation. Lipton gave a general framework for solving NP-complete problems using DNA [37, 38] (for the details of NP-completeness, the readers can consult [4]). Abstract DNA computer models with Turing universality are known [1, 5, 45, 59].

encoding an input (the given graph G) into DNA molecules, and performing a series of bio-operations such as ligation, PCR, affinity purification, and gel electrophoresis (for details of these operations, see [28]) on these information-encoding molecules, i.e., having these molecules interact with each other spontaneously.

Each of the vertices is encoded into a *carefully-designed* 20-mer DNA single strand. A directed edge $v_1 \to v_2$ is also encoded into a DNA single strand of length 20-mer whose first half is identical to the second half of the sequence encoding the source vertex v_1, and whose second half is identical to the first half of the sequence encoding the target vertex v_2. For instance, the DNA sequences for the vertex 3 and the directed edges $2 \to 3$ and $3 \to 4$ were encoded respectively as:

$$
\begin{aligned}
O_3 &= 5' - \texttt{GCTATTCGAGCTTAAAGCTA} - 3' \\
O_{2\to3} &= 5' - \texttt{GTATATCCGAGCTATTCGAG} - 3' \\
O_{3\to4} &= 5' - \texttt{CTTAAAGCTAGGCTAGGTAC} - 3'
\end{aligned}
$$

The edges $O_{2\to3}$ and $O_{3\to4}$ can interact with the Watson-Crick (WK-) complement of O_3 via the hydrogen bonds $\texttt{A} - \texttt{T}$ and $\texttt{C} - \texttt{G}$ as:

$$
\begin{array}{c}
5' - \overbrace{\texttt{GTATATCCGAGCTATTCGAG}}^{\text{edge } 2\to3}\overbrace{\texttt{CTTAAAGCTAGGCTAGGTAC}}^{\text{edge } 3\to4} - 3' \\
3' - \underbrace{\texttt{CGATAAGCTCGAATTTCGAT}}_{\text{complement of vertex 3}} - 5'
\end{array}
$$

Multiple copies of DNA sequences encoding edges and complements of vertices float in a test tube and adhere to each other, which amounts to generating the space of all candidate solutions. Among them, with high probability, is a totally double-stranded DNA sequence[2] of length $20 \times 7 = 140$mer which begins with the sequence for vertex 0, ends with the one for vertex 6, and the sequences for the others appear on it exactly once, which is the encoding of the Hamiltonian path. By increasing the number of copies of each DAN sequence in the test tube, the probability for this specific sequence to form increases. Any other DNA sequences in the test tube will be filtered out through a series of bio-operations, and finally the existence of the HPP sequence (output) can be detected by gel electrophoresis. Thus, the structure of this first DNA computing experiment can be abstractly described as follows:

information encoding The input is encoded into *carefully-designed* DNA sequences;

generating the solution space These sequences interact with each other *in vitro* in a *massive parallel manner*;

bio-operations Performing various bio-operations in an appropriate order to extract the correct solution.

A bio-algorithm to solve a 20-variable 3-SAT[3] problem by Braich et al. [6], another

[2]Note that both of its ends require special treatment to make it *totally* double stranded.

[3]SAT is an abbreviation of the satisfiability problem, the first problem proved to be NP-complete by Cook and Levin independently around 1971 [10, 34]. Its restricted version, 3-SAT, was also proved to be NP-complete by them. See also [4, 25].

significant milestone in DNA computing research, basically follows this idea[4].

In particular, the massive parallelism is the key principle that enabled the DNA computer to generate the whole solution space of the HPP instance in only one time-step (as opposed to exponential blow-up at the time-complexity in the classical electronic implementation of the algorithm). In Adleman's experiment, he spent a week to complete this computation, out of which generating solution space took one day and the rest of the time was devoted to extracting the solution from the test tube.

2.3 Information encoding in DNA computing

Needless to say, DNA computing is not free of challenges. Two major challenges are the space complexity trade-off, and information-encoding sequence design. In the above-mentioned brute force approach, the full-solution space has to be constructed. Hartmanis calculated that DNA sequences required to construct the full-solution space for an HPP instance with 200 vertices would outweigh the Earth [22]. The problem of how to avoid constructing the full-solution space has been addressed by many researchers. Morimoto, Arita, and Suyama [42] proposed a bio-algorithm in which, instead of generating all candidates before its computation phase, each path

[4] It is believed that as long as we rely on the brute-force algorithm, a DNA computer cannot solve 3-SAT with more than 70 variables [36]. The computational capacity of DNA computing for 3-SAT problem was *theoretically* strengthened so as to be able to solve 120-variables 3-SAT problem based on breadth-first search [60].

is stepwise extended from the start vertex (the vertex 0 in Figure 2.1) by checking whether a newly-appended vertex has never occurred on the path every time a vertex is appended and the path elongates. Another noteworthy example of this approach is an algorithm based on ligase chain reaction (LCR) by Wang et al. [57], which solves an n-variable m-clause SAT using the operations split, ligation, LCR, and merge m, n, m, and m times, respectively.

The other challenge lies in designing optimal information-encoding sequences. As exemplified previously in DNA computing for HPP, sequences encoding vertices and edges are supposed to bind to each other so as to form a double-stranded DNA sequence via WK-complementarity `A-T` and `C-G`. However, it is most likely that a sequence of 20mers contains all of the four kinds of nucleotides; in addition, a DNA sequence is bendable. As a result, a single-strand DNA sequence may fold into itself like hairpin if it contains two complementary subsequences, and would prevent it from interacting with other sequences. Such *intramolecular structures* (also called *secondary structures*) as well as *intermolecular structures* (structure formed by multiple sequences hybridizing among themselves) are actually preferred in general to their single-stranded form in terms of *(Gibbs) free energy*[5] [41]. The lower the free energy is, the more stable the structure gets. Hence, single-stranded sequence(s) tends to form an intra- or inter-molecular structure which achieves the minimum

[5]Informally speaking, the free energy of a structure of a DNA sequence is the sum of the energies required to melt all of its bonds.

free energy. This tendency is often made use of in order to predict an intramolecular structure of a given single-stranded DNA/RNA sequence [61, 62].

As a matter of course, the formation of inter- or intra-molecular structures can be useful in the molecular computation. The hairpin is the most typical intramolecular structure, and hence, employed to implement: read-only and rewritable DNA molecular memories [27, 52, 53, 54], logic circuits [7, 21, 44]. Whiplash PCR [20, 48], etc.. Therefore, the problem which would be more practical once being solved is not to prevent sequences from forming any intra- or inter-molecular structure (negative design), but to design sequences which form only desirable intra- or inter-molecular structures (positive design: these terminologies are from [40, 47]). The positive design problem is generally highly-related to a specific experiment and in the most general setting this problem is known to be equivalent to the independent set problem [16], and hence, NP-hard.

On the other hand, we can provide a general framework for the negative design problem by constructing a set of DNA sequences (encoding set) which do not allow for any undesired intra- or inter-molecular structures. Sager and Stefanovic in [47] proposed three conditions which such an encoding set must satisfy:

1. no sequence in the library forms any undesired intra-molecular structure (like hairpin);
2. no sequence in the encoding set hybridizes with a sequence in the encoding set

in any undesirable manner;

3. no sequence in the encoding set hybridizes with the complement of a sequence in the encoding set in any undesirable manner;

Note that these conditions are never sufficient conditions. The most significant constraint from the viewpoint of thermodynamics is the uniform melting temperature. Whether a DNA sequence and its complement interact with each other or not strongly depends on the temperature; i.e., if the temperature is below the melting temperature of the sequence. In the Adleman-Lipton model of DNA computing, the computation is the series of interactions between DNA sequences via WK-complementarity so that making the melting temperature uniform is important for obtaining uniform hybridization efficiency. One key to control the melting temperature is the `GC` content. Other constraints include forbidden sub-sequences and repeated bases. The existence of some specific sequence of bases may throw off the computation. In the experiment which employs a restriction enzyme[6], if its recognition site appeared on an encoding sequence, then the sequence would be cut by the enzyme. The repetition of bases is also known to be hazardous for DNA computing [50, 51].

Methods based on thermodynamics give us the most accurate solutions to the

[6]A *restriction enzyme* is an enzyme which recognizes a specific sequence of a double stranded DNA molecule and cut the DNA molecule at that location in a specific way, leaving either blunt ends or staggered sticky ends. Such a specific sequence is called the *recognition site* of the restriction enzyme.

negative design problem, but at the cost of massive computational power [11, 12]. One way to reduce this computational load which has attracted the attention of researchers for decades is to utilize a great deal of knowledge in information theory, coding theory, communication theory, combinatorics, computational complexity theory, automata theory, and formal language theory. The negative design problem has been blessed with this approach, and as a result, a big amount of research results have been obtained [3, 13, 14, 16, 18, 19, 23, 26, 29, 30, 32, 33, 35, 39, 40].

One of the most remarkable examples which illustrate how useful theories in computer science can be in negative design problem, and hence in DNA computing, is the method of designing DNA sequences based on templates by Arita and Kobayashi [3]. Since DNA molecules are words over the quaternary alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ of nucleotides, one can encode each nucleotide by 2bits, for example, as: $\mathtt{A} \to 10$, $\mathtt{C} \to 00$, $\mathtt{G} \to 01$, and $\mathtt{T} \to 11$. As such, two binary sequences, called the *template* and *map*, can encode a DNA single strand in such a manner that the template 1100 and the map 0101 encode `ATCG`. As a map, Arita and Kobayashi employ an error-correcting code so that the resulting set of sequences achieves a Hamming distance which is larger than a threshold. A challenging problem arises here of the size of search space of all sets of templates of length n (super-exponential 2^{2^n}). In general, sequence design problem cannot avoid enormous size of the search space, which requires a massive computational power. Computer simulation is one promising approach to this challenge. For example, Garzon, Blain, and Neel proposed a simulation tool (virtual test tube) called *EdnaCo* in [17]. *EdnaCo* provides us with the reliable prediction of designed coding molecules' behaviours in a test tube. Encoding sets will prove essential to implement various complex biomolecular mechanisms such as DNA self-assembly system [7, 46, 59].

As mentioned already, our contributions in this thesis have been purely theoretical. Nevertheless, we can observe their potential to be applied to practical problems in molecular biology such as encoding set design problem. Recall Conditions 2 and 3 among the above-mentioned three conditions which an encoding set must satisfy. As they suggest, it is usually the case that once a condition is taken into account as a criterion an encoding set must satisfy, one should also consider its complement variant. It would be convenient if one can deal with these two "complementary" conditions uniformly as:

- no sequence in the encoding set hybridizes with either a sequence in the encoding set or its complement in any undesirable manner.

Handling of this unified condition cannot help but involve cases analyses. Our results on the extensions of Fine and Wilf's theorem and Lyndon-Schützenberger equation and on the language equations as well as their proofs would highly probably alleviate the burden of such case studies.

Bibliography

[1] L. Adleman. On constructing a molecular computer. In R. J. Lipton and E. B. Baum, editors, *DNA Based Computers*, volume 27 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 1–22. American Mathematical Society, 1996.

[2] L. M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187):1021–1024, November 1994.

[3] M. Arita and S. Kobayashi. DNA sequence design using templates. *New Generation Computing*, 20:263–277, 2002.

[4] S. Arora and B. Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.

[5] D. Beaver. A universal molecular computer. In R. J. Lipton and E. B. Baum, editors, *DNA Based Computers*, volume 27 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 29–36. American Mathematical Society, 1996.

[6] R. Braich, N. Chelyapov, C. Johnson, P. Rothemund, and L. Adleman. Solution of a 20-variable 3-SAT problem on a DNA computer. *Science*, 296:499–502, 2002.

[7] E. Chiniforooshan, D. Doty, L. Kari, and S. Seki. Scalable, time-responsive, digital, energy-efficient molecular circuits using DNA strand displacement. submitted, 2010.

[8] M. Conrad. On design principles for a molecular computer. *Communications of the ACM*, 28(5):464–480, 1985.

[9] M. Conrad. Molecular computing paradigms. *Computer*, 25(11):6–9, 1992.

[10] S. A. Cook. The complexity of theorem proving procedure. In *3rd Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[11] R. Deaton, J. Chen, H. Bi, and J. Rose. A software tool for generating non-crosshybridizing libraries of DNA oligonucleotides. In M. Hagiya and A. Ohuchi, editors, *DNA Based Computer 8*, volume 2568 of *Lecture Notes in Computer Science*, pages 252–261. Springer, 2003.

[12] R. Dirks and N. Pierce. An algorithm for computing nucleic acid base-pairing probabilities including pseudoknots. *Journal of Computational Chemistry*, 25:1295–1304, 2004.

[13] A. Dyachkov, A. Macula, W. Pogozelski, T. Renz, V. Rykov, and D. Torney. New t-gap insertion-deletion-like metrics for DNA hybridization thermodynamic modeling. *Journal of Computational Biology*, 13(4):866–881, 2006.

[14] A. Dyachkov, A. Macula, V. Rykov, and V. Ufimtsev. DNA codes based on stem similarities between DNA sequences. In M. Garzon and H. Yan, editors, *DNA Computing 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 146–151. Springer, 2008.

[15] R. P. Feynman. There's plenty of room at the bottom, 1959. talk at the annual meeting of the American Physical Society at the California Institute of Technology (Caltech).

[16] A. G. Frutos, Q. Liu, A. J. Thiel, A. M. Sanner, A. E. Condon, L. M. Smith, and R. M. Corn. Demonstration of a word design strategy for DNA computing on surfaces. *Nucleic Acids Research*, 25(23):4748–4757, 1997.

[17] M. Garzon, D. Blain, and A. Neel. Virtual test tubes. *Natural Computing*, 3(4):461–477, 2004.

[18] M. Garzon, P. Neathery, R. Deaton, R. Murphy, D. Franceschetti, and S. Stevens Jr. A new metric for DNA computing. In J. Koza, K. Deb, M. Dorigo, D. Vogel, M. Garzon, H. Iba, and R. Riolo, editors, *Genetic Programming 1997*, pages 479–490. Morgan Kaufmann, 1997.

[19] M. Garzon, V. Phan, S. Roy, and A. Neel. In search of optimal codes for DNA computing. In C. Mao and T. Yokomori, editors, *DNA Computing 12*, volume 4287 of *Lecture Notes in Computer Science*, pages 143–156. Springer, 2006.

[20] M. Hagiya, M. Arita, D. Kiga, K. Sakamoto, and S. Yokoyama. Toward parallel evaluation and learning of boolean μ-formulas with molecules. In H. Rubin and D. Wood, editors, *DNA Based Computers III*, volume 48 of *DIMACS Series in Discrete Mathematics*, pages 57–72, 2000.

[21] M. Hagiya, S. Yaegashi, and K. Takahashi. Computing with hairpins and secondary structures of DNA. In *Nanotechnology: Science and Computation*, pages 293–308. Springer, 2006.

[22] J. Hartmanis. On the weight of computations. *Bulletin of the EATCS*, 55:136–138, 1995.

[23] T. Head. Relativised code concepts and multi-tube DNA dictionaries. In C. Calude and G. Păun, editors, *Finite Versus Infinite: Contributions to an Eternal Dilemma*, pages 175–186. Springer, 2000.

[24] M. Hirvensalo. *Quantum Computing*. Springer, 2 edition, 2004.

[25] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.

[26] N. Jonoska and K. Mahalingam. Languages of DNA based code words. In J. Chen and J. Reif, editors, *DNA Computing 9*, volume 2943 of *Lecture Notes in Computer Science*, pages 61–73. Springer, 2004.

[27] A. Kameda, M. Yamamoto, A. Ohuchi, S. Yaegashi, and M. Hagiya. Unravel four hairpins! *Natural Computing*, 7:287–298, 2008.

[28] L. Kari. DNA computing — the arrival of biological mathematics. *The Mathematical Intelligencer*, 19(2):9–22, 1997.

[29] L. Kari, S. Konstantinidis, and P. Sosík. On properties of bond-free DNA languages. *Theoretical Computer Science*, 334(1-3):131–159, 2005.

[30] L. Kari, K. Mahalingam, and G. Thierrin. The syntactic monoid of hairpin-free languages. *Acta Informatica*, 44:153–166, 2007.

[31] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, New York, 1972.

[32] A. Kijima and S. Kobayashi. Efficient algorithm for testing structure freeness of finite set of biomolecular sequences. In A. Carbone and N. Pierce, editors, *DNA Computing 11*, volume 3892 of *Lecture Notes in Computer Science*, pages 171–180. Springer, 2006.

[33] S. Kobayashi. Testing structure freeness of regular sets of biomolecular sequences (extended abstract). In C. Ferreti, G. Mauri, and C. Zandron, editors, *DNA 10*, volume 3384, pages 192–201. Springer, 2005.

[34] L. A. Levin. Universal sequential search problems. *PINFTRANS: Problems of Information Transmission (translated from Problemy Peredachi Informatsii (Russian)*, 9, 1973.

[35] M. Li, H. J. Lee, A. Condon, and R. M. Corn. DNA word design strategy for creating sets of non-interacting oligonucleotides for DNA microarrays. *Langmuir*, 18:805–812, 2002.

[36] R. Lipton. Using DNA to solve NP-complete problems. *Science*, 268:542–545, 1995.

[37] R. J. Lipton. DNA solution of hard computational problems. *Science*, 268(5210):542–545, 1995.

[38] R. J. Lipton and E. B. Baum, editors. *DNA Based Computers*, volume 27 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, 1996.

[39] A. Marathe, A. Condon, and R. Corn. On combinatorial DNA word design. *Journal of Computational Biology*, 8(3):201–219, 2001.

[40] G. Mauri and C. Ferretti. Word design for molecular computing: A survey. In J. Chen and J. Reif, editors, *DNA Computing 9*, volume 2943 of *Lecture Notes in Computer Science*, pages 37–46. Springer, 2004.

[41] J. S. McCaskill. The equilibrium partition function and base pair binding probability for rna secondary structure. *Biopolymers*, 29:1105–1119, 1990.

[42] N. Morimoto, M. Arita, and A. Suyama. Stepwise generation of Hamiltonian Path with molecules. In D. Lundh, B. Olsson, and A. Narayanan, editors, *Biocomputing and Emergent Computation*, pages 184–192, 1997.

[43] K. B. Mullis and F. A. Faloona. Specific synthesis of DNA in vitro via a polymerase-catalyzed chain reaction. *Methods in Enzymology*, 155:335–350, 1987.

[44] L. Qian and E. Winfree. A simple DNA gate motif for synthesizing large-scale circuits. In G. Goos, J. Hartmanis, and J. van Leeuwen, editors, *DNA 14*, volume 5347 of *Lecture Notes in Computer Science*, pages 70–89, 2008.

[45] P. W. K. Rothemund. A DNA and restriction enzyme implementation of turing machines. In R. J. Lipton and E. B. Baum, editors, *DNA Based Computers*, volume 27 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 75–119. American Mathematical Society, 1996.

[46] P. W. K. Rothemund and E. Winfree. The program-size complexity of self-assembled squares (extended abstract). In *STOC*, pages 459–468, 2000.

[47] J. Sager and D. Stefanovic. Designing nucleotide sequences for computation: A survey of constraints. In A. Carbone and N. Pierce, editors, *DNA Computing 11*, volume 3892 of *Lecture Notes in Computer Science*. Springer, 2006.

[48] K. Sakamoto, D. Kiga, K. Komiya, H. Gouzu, S. Yokoyama, S. Ikeda, H. Sugiyama, and M. Hagiya. State transitions by molecules. *Biosystems*, 52(1-3):81–91, 1999.

[49] T. D. Schneider. Claude Shannon: Biologist. *IEEE Engineering in Medicine and Biology Magazine*, 25(1):30–33, 2006.

[50] D. Sen and W. Gilbert. Formation of parallel four-stranded complexes by guanine-rich motifs in DNA and its implications for meiosis. *Nature*, 334(6180):364–366, 1988.

[51] D. Sen and W. Gilbert. A sodium-pottasium switch in the formation of four-stranded G4-DNA. *Nature*, 344(6265):410–414, 1990.

[52] J. S. Shin and N. A. Pierce. Rewritable memory by controllable nanopatterning of DNA. *Nanoletters*, 4:905–909, 2004.

[53] M. Takinoue and A. Suyama. Molecular reactions for a molecular memory based on hairpin DNA. *Chem-Bio Informatics Journal*, 4:93–100, 2004.

[54] M. Takinoue and A. Suyama. Hairpin-DNA memory using molecular addressing. *Small*, 2(11):1244–1247, 2006.

[55] M. N. Vaintsvaig and E. A. Liberman. Formal description of cell molecular computer. *Biofizika*, 18:939–942, 1973.

[56] J. von Neumann. *Theory of Self-reproducing Automata*. U. Illinois Press, 1966. Edited and Completed by A. W. Burks.

[57] X. Wang, Z. Bao, J. Hu, S. Wang, and A. Zhan. Solving the SAT problem using a DNA computing algorithm based on ligase chain reaction. *Biosystems*, 91:117–125, 2008.

[58] J. D. Watson and F. H. C. Crick. A structure for deoxyribose nucleic acid. *Nature*, 171:737–738, 1953.

[59] E. Winfree, X. Yang, and N. Seeman. Universal computation via self-assembly of DNA: Some theory and experiments. In *Proc. DNA-Based Computers II*, volume 44 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 191–213. American Mathematical Society, 1996.

[60] H. Yoshida and A. Suyama. Solution to 3-SAT by breadth-first search. In E. Winfree and D. Gifford, editors, *DNA Based Computers V*, volume 54 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 9–22, 2000.

[61] M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Research*, 31(13):3406–3415, 2003.

[62] M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9(1):133–148, 1981.

Part II

Results in Combinatorics on Words

Chapter 3

An extension of Fine and Wilf's theorem

What follows is the contents of "On a special class of primitive words"[1] by Elena Czeizler, Lila Kari, and Shinnosuke Seki, which was published in Theoretical Computer Science as:

E. Czeizler, L. Kari, and S. Seki.

On a special class of primitive words.

Theoretical Computer Science, 411(3):617-630, 2010.

The conference version of this paper was presented at Mathematical Foundations of Computer Science (MFCS 2008):

E. Czeizler, L. Kari, and S. Seki.

[1]A version of this chapter has been published.

On a special class of primitive words.

In *MFCS 2008*, volume 5162 of *Lecture Notes in Computer Science*, pages 265-277. Springer, 2008.

Summary: When representing DNA molecules as words, it is necessary to take into account the fact that a word u encodes basically the same information as its Watson-Crick complement $\theta(u)$, where θ denotes the Watson-Crick complementarity function. Thus, an expression which involves only a word u and its complement can be still considered as a repeating sequence. In this context, we define and investigate the properties of a special class of primitive words, called pseudo-primitive words relative to θ or simply θ-primitive words, which cannot be expressed as such repeating sequences. For instance, we prove the existence of a unique θ-primitive root of a given word, and we give some constraints forcing two distinct words to share their θ-primitive root. Also, we present an extension of the well-known Fine and Wilf theorem, for which we give an optimal bound.

On a special class of primitive words

Elena Czeizler[1], Lila Kari[2], and Shinnosuke Seki[2]

[1] Department of IT, Åbo Akademi University, Turku 20520, Finland.

[2] Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

3.1 Introduction

Encoding information as DNA strands as in, e.g., DNA Computing, brings up for investigation new features based on the specific biochemical properties of DNA molecules. Recall that single-stranded DNA molecules can be viewed as words over the quaternary alphabet of bases $\{\mathtt{A}, \mathtt{T}, \mathtt{C}, \mathtt{G}\}$. Moreover, one of the main properties of DNA molecules is the Watson-Crick complementarity of the bases A and T and respectively G and C. Because of this property two Watson-Crick complementary single DNA strands with opposite orientation bind together to form a DNA double strand, in a process called base-pairing. Recently, there were several approaches to generalize notions from classical combinatorics on words in order to incorporate this major characteristic of DNA molecules, see, e.g., [8, 9, 6]. Along these lines, in this paper, we generalize the concept of *primitivity* and define *pseudo-primitive words*.

The notion of periodicity plays an important role in various fields of theoretical computer science, such as algebraic coding theory, [13], and combinatorics on words, [11, 2]. An integer $p \geq 1$ is a period of a word if any of two letters on the word which are distant from each other by p letters are the same. The well-known periodicity theorem by Fine and Wilf states that if a word has two periods p, q and is of length at least $p+q-\gcd(p,q)$, then $\gcd(p,q)$ is also a period of the word, where $\gcd(p,q)$ is the greatest common divisor of p and q [7]. This theorem can be rephrased as: if a power of a word u and a power of a word v share the same prefix of length $|u|+|v|-\gcd(|u|,|v|)$, then u and v are powers of a same word t. This description elucidates the relationship between the Fine and Wilf theorem and the notion of primitivity. A word is called *primitive* if it cannot be decomposed as a power of another word. Investigating the primitivity of a word is often the first step when analyzing its properties. Moreover, how a word can be decomposed and whether two words are powers of a common word are two questions which were widely investigated in language theory, see, e.g., [11, 2, 16].

While, in classical combinatorics on words we look for repetitions of the form u^i for some word u and some $i \geq 2$, when dealing with DNA molecules (i.e., their abstract representation as words) we have to take into account the fact that a word u encodes the same information as its complement $\theta(u)$, where θ denotes the Watson-Crick complementarity function, or its mathematical formalization as an arbitrary antimorphic involution. In other words, we can extend the notion of power to *pseudo-power relative to* θ or simply θ*-power*. A θ-power of u is a word of the form $u_1u_2\cdots u_n$ for some $n \geq 1$, where $u_1 = u$ and for any $2 \leq i \leq n$, u_i is either u or $\theta(u)$. In this context, we define θ*-primitive words* as strings which cannot be a θ-power of another word. Also, we define the θ*-primitive root* of a word w as the shortest word u such that w is a θ-power of u. In classical combinatorics on words, there exist two equivalent definitions for the *primitive root* of a word w as the shortest word u such that w is a power of u, or the unique primitive word u such that w is a power of u. The first main contribution of this paper is to propose such equivalent definitions for the θ-primitive root of a word, that is, we prove that the θ-primitive root of a word w is the unique θ-primitive word u such that w is a θ-power of u. In the process of obtaining this result, we also prove an extension of the Fine and Wilf theorem. Until now, several extensions of this theorem were proved, see, e.g., [1, 3, 4, 12, 14, 15]. In this paper, we look at the case when a θ-power of u and a θ-power of v share a same prefix. If the prefix is longer than a given bound, then we prove that u and v are θ-powers of a same word, that is, they share their θ-primitive root. Our bound is twice the length of the longer word (u or v) plus the length of the other word minus the greatest common divisor of the lengths of u and v. Moreover, we show that this bound is optimal.

The paper is organized as follows. In Section 3.2, we fix our terminology and recall some basic results. In Section 3.3 we investigate some basic properties of θ-primitive words. In particular, we give an extension of the Fine and Wilf theorem which implies immediately that we can define the θ-primitive root of a word in the two equivalent ways. In Section 3.4, we present several constraints forcing two words to share their θ-primitive root. In Section 3.5, we investigate some connections between the θ-primitive words that we introduced here and the θ-palindrome words, which were proposed and investigated in [9, 6]. In Section 3.6, we present the optimal bound for our extension of the Fine and Wilf theorem.

3.2 Preliminaries

Let Σ be a finite alphabet. We denote by Σ^* the set of all finite words over the alphabet Σ, by ϵ the empty word, and by Σ^+ the set of all nonempty finite words over Σ. The *length* of a word w, denoted by $|w|$, is the number of letters occurrences, i.e., if $w = a_1 \ldots a_n$ with $a_i \in \Sigma$, $1 \leq i \leq n$, then $|w| = n$. For a letter $a \in \Sigma$, let $|w|_a$ denote the number of occurrences of a in w. Therefore, $|w| = \sum_{a\in\Sigma} |w|_a$. We say that u is a *prefix* (resp. a *suffix*) of v, if $v = ut$ (resp. $v = tu$) for some $t \in \Sigma^*$. For any integer $0 \leq k \leq |v|$, we use the notation $\mathrm{pref}_k(v)$ ($\mathrm{suff}_k(v)$) for the prefix (resp. suffix) of length k of a word v, and $\mathrm{Pref}(v)$ ($\mathrm{Suff}(v)$) for the set of all prefixes (resp. all suffixes) of v. In particular $\mathrm{pref}_0(v) = \epsilon$ for any word $v \in \Sigma^*$. An integer $p \geq 1$ is a *period* of a word $w = a_1 \ldots a_n$, with $a_i \in \Sigma$ for all $1 \leq i \leq n$, if $a_i = a_{i+p}$ for all $1 \leq i \leq n - p$.

A word $w \in \Sigma^+$ is called *primitive* if it cannot be written as a power of another word; that is, $w = u^n$ implies $n = 1$ and $w = u$. For a word $w \in \Sigma^+$, the shortest $u \in \Sigma^+$ such that $w = u^n$ for some $n \geq 1$ is called the *primitive root* of the word w and is denoted by $\rho(w)$. The following result gives an alternative, equivalent way for defining the primitive root of a word.

Theorem 3.1. *For each word $w \in \Sigma^*$, there exists a unique primitive word $t \in \Sigma^+$ such that $\rho(w) = t$, i.e., $w = t^n$ for some $n \geq 1$.*

The next result illustrates another property of primitive words.

Proposition 3.2. *Let $u \in \Sigma^+$ be a primitive word. Then u cannot be a factor of u^2 in a nontrivial way, i.e., if $u^2 = xuy$, then necessarily either $x = \epsilon$ or $y = \epsilon$.*

We say that two words u and v *commute* if $uv = vu$. The following result characterizes the commutation of two words in terms of primitive roots.

Theorem 3.3. *For $u, v \in \Sigma^*$, the following conditions are equivalent: i) u and v commute; ii) u and v satisfy a non-trivial relation, i.e., an equation where the two sides are not graphically identical; iii) u and v have the same primitive root.*

For two words u and v, we denote by $u \wedge v$ the *maximal common prefix of u and v*. The following result from [2] will be very useful in our future considerations.

Theorem 3.4. *Let $X = \{x, y\} \subseteq \Sigma^*$ such that $xy \neq yx$. Then, for each words $u, v \in X^*$ we have*

$$u \in xX^+,\ v \in yX^+,\ |u|, |v| \geq |xy \wedge yx|,\ \Rightarrow u \wedge v = xy \wedge yx.$$

The following result is an immediate consequence.

Corollary 3.5. *Let $X = \{x, y\} \subseteq \Sigma^*$, $u \in xX^*$, and $v \in yX^*$ such that $|u|, |v| \geq |xy|$. If $|u \wedge v| \geq |xy|$, then $\rho(x) = \rho(y)$.*

Two words u and v are said to be *conjugate* if there exist words x and y such that $u = xy$ and $v = yx$. In other words, v can be obtained via a cyclic permutation of u. The next result characterizes the conjugacy of two words.

Theorem 3.6. *Let $u, v \in \Sigma^+$. Then the following conditions are equivalent: i) u and v are conjugate; ii) there exists a word z such that $uz = zv$; moreover, this holds if and only if $u = pq$, $v = qp$, and $z = (pq)^i p$, for some $p, q \in \Sigma^*$ and $i \geq 0$; iii) the primitive roots of u and v are conjugate.*

Note that conjugacy is an equivalence relation, the *conjugacy class* of a word w consisting of all conjugates of w. The following is a well-known result.

Proposition 3.7. *If w is a primitive word, then its conjugacy class contains $|w|$ distinct primitive words.*

The following result, known as the Fine and Wilf theorem in its form for words, see [2, 11], illustrates a fundamental periodicity property of words. As usual, $\gcd(n, m)$ denotes the *greatest common divisor of n and m.*

Theorem 3.8. *Let $u, v \in \Sigma^*$, $n = |u|$, $m = |v|$, and $d = \gcd(n, m)$. If two powers u^i and v^j of u and v have a common prefix of length at least $n + m - d$, then u and v are powers of a common word. Moreover, the bound $n + m - d$ is optimal.*

A mapping $\theta : \Sigma^* \to \Sigma^*$ is called a *morphism* (an *antimorphism*) if for any words $u, v \in \Sigma^*$, $\theta(uv) = \theta(u)\theta(v)$ (resp. $\theta(uv) = \theta(v)\theta(u)$). A mapping $\theta : \Sigma^* \to$

Σ^* is called an *involution* if, for all words $u \in \Sigma^*$, $\theta(\theta(u)) = u$. Watson-Crick complementarity is a typical example of antimorphic involutions; in fact, it is defined as the antimorphic involution θ satisfying $\theta(\mathtt{A}) = \mathtt{T}$, $\theta(\mathtt{T}) = \mathtt{A}$, $\theta(\mathtt{C}) = \mathtt{G}$, and $\theta(\mathtt{G}) = \mathtt{C}$, which is called the Watson-Crick involution.

For a mapping $\theta : \Sigma^* \to \Sigma^*$, a word $w \in \Sigma^*$ is called *θ-palindrome* if $w = \theta(w)$, see [9, 6]. We say that a word $w \in \Sigma^+$ is a *pseudo-power of a non-empty word $t \in \Sigma^+$ relative to θ*, or simply *θ-power of t*, if $w \in t\{t, \theta(t)\}^*$. Conversely, t is called a *pseudo-period of w relative to θ*, or simply *θ-period of w* if $w \in t\{t, \theta(t)\}^*$. Hence t is a θ-period of w if and only if w is a θ-power of t. We call a word $w \in \Sigma^+$ *θ-primitive* if there exists no non-empty word $t \in \Sigma^+$ such that w is a θ-power of t and $|w| > |t|$. We define the *θ-primitive root of w*, denoted by $\rho_\theta(w)$, as the shortest word t such that w is a θ-power of t.

3.3 Properties of θ-primitive words

In this section, we consider $\theta : \Sigma^* \to \Sigma^*$ to be either a morphic or an antimorpic involution, other than the identity function. We start by looking at some basic properties of θ-primitive words.

Proposition 3.9. *If a word $w \in \Sigma^+$ is θ-primitive, then it is also primitive. Moreover, the converse is not always true.*

Proof. Suppose that w is a θ-primitive word but not primitive. Then there exists

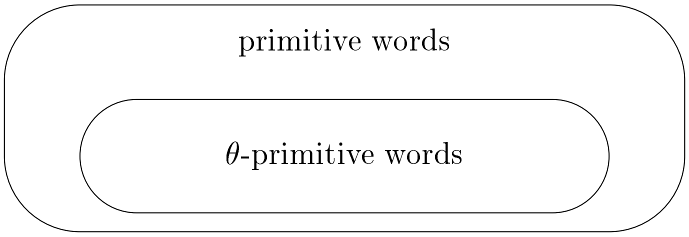

Figure 3.1: The sets of primitive and θ-primitive words

some $t \in \Sigma^+$ such that $w = t^n$ with $n \geq 2$. By definition of θ-power, w is a θ-power of t. However, this contradicts the θ-primitivity of w because $|t| < |w|$. For the converse, since θ is not the identity function, there exists a letter a such that $\theta(a) \neq a$. Then, if we take $w = a\theta(a)$, it is obvious that w is primitive, but not θ-primitive. $\square$

Thus, the class of θ-primitive words is strictly included in the set of primitive ones, as illustrated in Figure 3.1.

Proposition 3.10. *The θ-primitive root of a word is θ-primitive.*

Proof. Let $w \in \Sigma^+$ and $t = \rho_\theta(w)$ be its θ-primitive root, that is, w is a θ-power of t. Suppose, now that t is not θ-primitive. Then there exists a word $s \in \Sigma^*$ such that t is a θ-power of s and $|s| < |t|$. Note that $\theta(t)$ is a θ-power of either s or $\theta(s)$. Thus, w is a θ-power of s. However, this contradicts that t being the θ-primitive root of w because $|s| < |t|$. $\square$

We also obtain the following result as an immediate consequence.

Corollary 3.11. *The θ-primitive root of a word is primitive.*

Contrary to the case of primitive words, a conjugate of a θ-primitive word need not be θ-primitive, as shown by the following two examples.

Example 1. Let $\theta : \{\mathtt{A}, \mathtt{T}, \mathtt{C}, \mathtt{G}\}^* \to \{\mathtt{A}, \mathtt{T}, \mathtt{C}, \mathtt{G}\}^*$ be the Watson-Crick involution defined in Section 3.2. Then the word $w = \mathtt{GCTA}$ is θ-primitive, while its conjugate $w' = \mathtt{AGCT} = \mathtt{AG}\theta(\mathtt{AG})$ is not.

Example 2. Let $\theta : \{a, b, c, d\}^* \to \{a, b, c, d\}^*$ be a morphic involution defined by $\theta(a) = c$, $\theta(c) = a$, $\theta(b) = d$, and $\theta(d) = b$. Then the word $w = abadcb$ is θ-primitive, while its conjugate $w' = babadc = (ba)^2\theta(ba)$ is not.

So, we can formulate the following result.

Proposition 3.12. *The class of θ-primitive words is not necessarily closed under circular permutations.*

Fine and Wilf's result on words (Theorem 3.8) constitutes one of the fundamental periodicity properties of words. Thus, a natural question is whether we can obtain an extension of this result when for two words u, v, instead of taking a power of u and a power of v, we look at a θ-power of u and a θ-power of v. First, we analyze the case when θ is a morphic involution; it turns out that in this case we can obtain the same bound as in Theorem 3.8. However, since the proof of this result is analogous to the one for Theorem 3.8, see for instance [11], we will not include it here.

Theorem 3.13. *Let $\theta : \Sigma^* \to \Sigma^*$ be a morphic involution, $u, v \in \Sigma^+$ with $n = |u|$, $m = |v|$, and $d = \gcd(n, m)$, $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$, and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$. If the two θ-powers $\alpha(u, \theta(u))$ and $\beta(v, \theta(v))$ have a common prefix of length at least $n + m - d$, then there exists a word $t \in \Sigma^+$ such that $u, v \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Moreover, the bound $n + m - d$ is optimal.*

However, as illustrated by the following example, if the mapping θ is an antimorphic involution, then the bound given by Theorem 3.13 is not enough anymore.

Example 3. Let $\theta : \{a, b\}^* \to \{a, b\}^*$ be the mirror mapping defined as follows: $\theta(a) = a, \theta(b) = b$, and $\theta(w_1 \ldots w_n) = w_n \ldots w_1$, where $w_i \in \{a, b\}$ for all $1 \leq i \leq n$. Obviously, θ is an antimorphic involution on $\{a, b\}^*$. Let now $u = (ab)^k b$ and $v = ab$. Then, u^2 and $v^k \theta(v)^{k+1}$ have a common prefix of length $2|u| - 1 > |u| + |v| - \gcd(|u|, |v|)$. However u and v do not have the same θ-primitive root, that is, $\rho_\theta(u) \neq \rho_\theta(v)$.

Before stating an analogous result also for the case of antimorphic involutions, we introduce the mapping $\varphi : \Sigma^* \times \Sigma \to \mathbb{N}$ defined as $\varphi(u, a) = |u|_a + |u|_{\theta(a)}$, that is, the number of occurrences of the letters a and $\theta(a)$ in the word u. Note that for any letter a and any word u, $\varphi(u, a) = \varphi(u, \theta(a)) \leq |u|$, with equality only when $u \in \{a, \theta(a)\}^*$. We will call this mapping the *characteristic function on the alphabet* Σ. Moreover, $\operatorname{lcm}(n, m)$ denotes, as usual, the *least common multiple of n and m.*

Theorem 3.14. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution, $u, v \in \Sigma^+$, and*

$\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$, $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ be two θ-powers sharing a common prefix of length at least $\mathrm{lcm}(|u|, |v|)$. *Then, there exists a word $t \in \Sigma^+$ such that $u, v \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. In particular, if $\alpha(u, \theta(u)) = \beta(v, \theta(v))$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. The proof of this result uses the techniques from [4]. First, we can suppose, without loss of generality that $\gcd(|u|, |v|) = 1$ and thus $\mathrm{lcm}(|u|, |v|) = |u||v|$. Otherwise, i.e., $\gcd(|u|, |v|) = d \geq 2$, we consider a new alphabet $\Sigma' = \Sigma^d$, where the new letters are words of length d in the original alphabet, and we look at the words u and v as elements of $(\Sigma')^+$. In the larger alphabet $\gcd(|u|, |v|) = 1$, and if we can prove the theorem there it immediately gives the general proof. Let now $|u| = n$ and $|v| = m$. If we denote by $\alpha'(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta'(v, \theta(v)) \in v\{v, \theta(v)\}^*$ the prefixes of length $\mathrm{lcm}(n, m) = nm$ of $\alpha(u, \theta(u))$ and $\beta(v, \theta(v))$, respectively, then we actually have $\alpha'(u, \theta(u)) = \beta'(v, \theta(v))$.

Since the mapping θ is an involution, we can easily notice that for any word w and any letter a, $\varphi(w, a) = \varphi(\theta(w), a)$. Moreover, since $\alpha'(u, \theta(u)) = \beta'(v, \theta(v))$, whenever, for a letter a, $\varphi(u, a) > 0$, we also have that $\varphi(v, a) > 0$.

Suppose now that there exist two letters a and b such that $\{a, \theta(a)\} \cap \{b, \theta(b)\} = \emptyset$, $\varphi(u, a) > 0$, and $\varphi(u, b) > 0$. Then, since $n = |u| = \sum_{c \in \Sigma} |u|_c$, we have that $\varphi(u, a) < n$. Let us look next at the number of occurrences of a and $\theta(a)$ in the two sides of the equality $\alpha'(u, \theta(u)) = \beta'(v, \theta(v))$. Since $|\alpha'(u, \theta(u))| = |\beta'(v, \theta(v))| = nm$, where $|u| = n$, and $|v| = m$, we obtain $m\varphi(u, a) = n\varphi(v, a)$. However this contradicts the fact that $\gcd(n, m) = 1$ and $\varphi(u, a) < n$. So, there exists a letter $a \in \Sigma$ such that $u \in \{a, \theta(a)\}^+$. Since $\alpha'(u, \theta(u)) = \beta'(v, \theta(v))$, this implies that also $v \in \{a, \theta(a)\}^+$. Thus, $\rho_\theta(u) = \rho_\theta(v)$. ☐

Note that, in many cases there is a big gap between the bounds given in Theorems 3.13 and 3.14. Moreover, Theorem 3.14 does not give the optimal bound for the general case when θ is an antimorphic involution. In Section 3.6, we show that this optimal bound for the general case is $2|u| + |v| - \gcd(|u|, |v|)$, where $|u| > |v|$, while for some particular cases we obtain bounds as low as $|u| + |v| - \gcd(|u|, |v|)$. As an immediate consequence of Theorems 3.13 and 3.14, we obtain the following result.

Corollary 3.15. *For any word $w \in \Sigma^+$ there exists a unique θ-primitive word $t \in \Sigma^+$ such that $w \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(w) = t$.*

Let us note now that, maybe even more importantly, just as in the case of primitive words, this result provides us with an alternative, equivalent way for defining the θ-primitive root of a word w, i.e., *the θ-primitive word t such that $w \in t\{t, \theta(t)\}^*$*. This proves to be a very useful tool in our future considerations.

Moreover, we also obtain the following two results as immediate consequences of Theorems 3.13 and 3.14.

Corollary 3.16. *Let $u, v \in \Sigma^+$ be two words such that $\rho(u) = \rho(v) = t$. Then $\rho_\theta(u) = \rho_\theta(v) = \rho_\theta(t)$.*

Corollary 3.17. *If we have two words $u, v \in \Sigma^+$ such that $u \in v\{v, \theta(v)\}^*$, then $\rho_\theta(u) = \rho_\theta(v)$.*

3.4 Relations imposing θ-periodicity

It is well-known, due to Theorem 3.3, that any non-trivial equation over two distinct words forces them to be powers of a common word, i.e., to share a common primitive root. Thus, a natural question is whether this would be the case also when we want two distinct words to be θ-powers of a common word, i.e., to share a common θ-primitive root. From [8], we already know that the equation $uv = \theta(v)u$ imposes $\rho_\theta(u) = \rho_\theta(v)$ only when θ is a morphic involution. In this section, we give several examples of equations over $\{u, \theta(u), v, \theta(v)\}$ forcing $\rho_\theta(u) = \rho_\theta(v)$ in the case when $\theta : \Sigma^* \to \Sigma^*$ is an antimorphic involution.

The first equation we look at is very similar to the commutation equation of two words, but it involves also the mapping θ.

Theorem 3.18. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ and $u, v \in \Sigma^+$. If $uv\theta(v) = v\theta(v)u$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. Since $uv\theta(v) = v\theta(v)u$, we already know, due to Theorem 3.3, that there exists a primitive word $t \in \Sigma^+$ such that $u = t^i$ and $v\theta(v) = t^j$, for some $i, j \geq 0$. If $j = 2k$ for some $k \geq 0$, then we obtain immediately that $v = \theta(v) = t^k$, i.e., $\rho(u) = \rho(v) = t$. Thus, $\rho_\theta(u) = \rho_\theta(t) = \rho_\theta(v)$. Otherwise, i.e., $j = 2k + 1$, we can write $v = t^k t_1$ and $\theta(v) = t_2 t^k$, where $t = t_1 t_2$ and $|t_1| = |t_2| > 0$. Hence, $\theta(v) = \theta(t_1)\theta(t)^k = t_2 t^k$, which implies $t_2 = \theta(t_1)$. In conclusion, $u, v \in t_1\{t_1, \theta(t_1)\}^*$, for some word $t_1 \in \Sigma^+$, i.e., $\rho_\theta(u) = \rho_\theta(t_1) = \rho_\theta(v)$. □

Example 4. Let $\theta : \{a,b\}* \to \{a,b\}^*$ be defined as $\theta(a) = b$ and $\theta(b) = a$, and let $u = ab$ and $v = aba$. Then $uv\theta(v) = v\theta(v)u = (ab)^4$ and $\rho_\theta(u) = \rho_\theta(v) = a$.

Next, we modify the previous equation, such that on one side, instead of $v\theta(v)$, we take its conjugate $\theta(v)v$.

Theorem 3.19. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ and $u, v \in \Sigma^+$. If $v\theta(v)u = u\theta(v)v$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. If we concatenate the word $\theta(v)$ to the right on both sides of the equation $v\theta(v)u = u\theta(v)v$, then we obtain $(v\theta(v))(u\theta(v)) = (u\theta(v))(v\theta(v))$. Due to Theorem 3.3, this means that there exists a primitive word $t \in \Sigma^+$ such that $v\theta(v) = t^i$ and $u\theta(v) = t^j$, for some $i, j \geq 0$, $j \geq \lceil i/2 \rceil$. If $i = 2k$ for some $k \geq 0$, then $\theta(v) = v = t^k$ and thus also $u = t^{j-k}$, i.e., $\rho(u) = \rho(v) = t$. Henceforth, $\rho_\theta(u) = \rho_\theta(t) = \rho_\theta(v)$. Otherwise, i.e., $j = 2k + 1$, we can write $v = t^k t_1$ and $\theta(v) = t_2 t^k$, where $t = t_1 t_2$ and $|t_1| = |t_2| > 0$. Hence, we achieve again $t_2 = \theta(t_1)$, which implies that $v \in t_1\{t_1, \theta(t_1)\}^*$. Moreover, since $u\theta(v) = t^j$, we also obtain $u = t^{j-k-1} t_1 \in t_1\{t_1, \theta(t_1)\}^*$. Thus, $\rho_\theta(u) = \rho_\theta(t_1) = \rho_\theta(v)$. □

Example 5. Using Σ defined in Example 4, let $u = a$ and $v = aba$. Then $v\theta(v)u = u\theta(v)v = abababa$ and $\rho_\theta(u) = \rho_\theta(v) = a$.

The next result gives an example of a more intricate equation which also imposes θ-periodicity.

Theorem 3.20. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ and $u, v \in \Sigma^*$. If $u^2v = vu\theta(u)$, then $u = \theta(u)$ and $\rho(u) = \rho(v)$.*

Proof. Since $u^2v = vu\theta(u)$, due to Theorem 3.6, there exist some words $z, t \in \Sigma^*$ and some integer $k \geq 0$ such that $u^2 = zt$, $u\theta(u) = tz$, and $v = (zt)^k z$. This representation clarifies that $u\theta(u)$ can be obtained by cyclically permuting u^2. Note that this operation preserves the property of the word being square. Thus, $u\theta(u) = w^2$ for some $w \in \Sigma^*$, and in fact we have $u = \theta(u)$ because θ is length-preserving. As a result, the given equation becomes $u^2v = vu^2$ so that $\rho(u) = \rho(v)$. □

Observe that the primitive root of a θ-palindromic word is θ-palindrome. As such, Theorem 3.20 means that $u^2v = vu\theta(u)$ implies $v = \theta(v)$. Examples of u and v satisfying $u^2v = vu\theta(u)$ are hence quite trivial like $u = w^i$ and $v = w^j$ for some θ-palindrome w and $i, j \geq 0$.

Next, we look at the case when both uv and vu are θ-palindromic words, which also proves to be enough to impose that $u, v \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$.

Theorem 3.21. *Let $u, v \in \Sigma^*$ be two words such that both uv and vu are θ-palindrome and let $t = \rho(uv)$. Then, $t = \theta(t)$ and either $\rho(u) = \rho(v) = t$ or $u = (t_1\theta(t_1))^i t_1$ and $v = \theta(t_1)(t_1\theta(t_1))^j$, where $t = t_1\theta(t_1)$ and $i, j \geq 0$.*

Proof. The equality $uv = \theta(uv)$ immediately implies that $t = \theta(t)$. Moreover, if u and v commute, then $\rho(u) = \rho(v) = \rho(uv) = t$. Assume now that u and v do not commute. Since $\rho(u) \neq \rho(v)$ and $uv = t^n$ for some $n \geq 1$, we can write $u = t^i t_1$ and $v = t_2 t^{n-i-1}$ for some $i \geq 0$ and $t_1, t_2 \in \Sigma^+$ such that $t = t_1 t_2$. Thus, $vu = t_2 t^{n-1} t_1 = (t_2 t_1)^n$ and since $vu = \theta(vu)$ we obtain that also $t_2 t_1$ is θ-palindrome, i.e., $t_2 t_1 = \theta(t_2 t_1) = \theta(t_1)\theta(t_2)$. Now, if $|t_1| = |t_2|$, then $t_2 = \theta(t_1)$ and thus $t = t_1 \theta(t_1)$, $u = t^i t_1$, and $v = \theta(t_1) t^{n-i-1}$. Otherwise, either $|t_1| > |t_2|$ or $|t_1| < |t_2|$. We consider next only the case $|t_1| > |t_2|$, the other one being similar. Since $t_2 t_1 = \theta(t_1)\theta(t_2)$, we can write $\theta(t_1) = t_2 x$ and $t_1 = x\theta(t_2)$ for some word $x \in \Sigma^+$ with $x = \theta(x)$. Then, since $t = \theta(t)$ we have that $t = t_1 t_2 = x\theta(t_2)t_2 = \theta(x\theta(t_2)t_2) = \theta(t_2)t_2 x$. Hence, x and $\theta(t_2)t_2$ commute, which contradicts the primitivity of t. □

Example 6. With θ defined in Example 4, let $u = aba$ and $v = babab$. Then both uv and vu are θ-palindrome. For such u and v, $t = \rho(uv) = ab = a\theta(a)$.

As an immediate consequence we obtain the following result.

Corollary 3.22. *For $u, v \in \Sigma^*$, if $uv = \theta(uv)$ and $vu = \theta(vu)$, then $\rho_\theta(u) = \rho_\theta(\theta(v))$. In particular, there exists some $t \in \Sigma^+$ such that $u, v \in \{t, \theta(t)\}^*$.*

3.5 On θ-primitive and θ-palindromic words

In this section, we investigate some word equations under which a θ-primitive word must be θ-palindrome. Throughout this section we consider $\theta : \Sigma^* \to \Sigma^*$ to be an

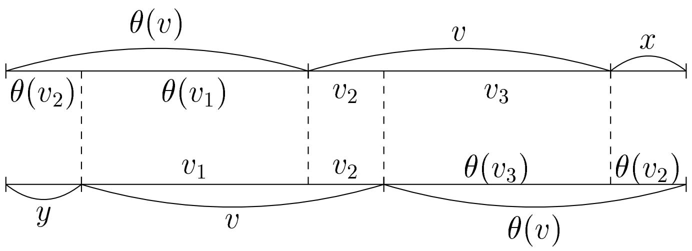

Figure 3.2: The equation $\theta(v)vx = yv\theta(v)$

antimorphic involution over the alphabet Σ.

Theorem 3.23. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ and $v \in \Sigma^+$ be a θ-primitive word. If $\theta(v)vx = yv\theta(v)$ for some words $x, y \in \Sigma^*$ with $|x|, |y| < |v|$, then v is θ-palindrome and $x = y = \epsilon$.*

Proof. Assume there exist some words $x, y \in \Sigma^*$ with $|x|, |y| < |v|$, such that $\theta(v)vx = yv\theta(v)$, as illustrated in Figure 3.2.

Then, we can write $v = v_1v_2 = v_2v_3$, with $v_1, v_2, v_3 \in \Sigma^*$, $y = \theta(v_2) = x$, $v_1 = \theta(v_1)$, $v_3 = \theta(v_3)$. Since $v_1v_2 = v_2v_3$, we can write $v_1 = pq$, $v_3 = qp$, $v_2 = (pq)^ip$, and $v = (pq)^{i+1}p$ for some words $p, q \in \Sigma^*$ and some $i \geq 0$. Thus, $pq = \theta(pq)$ and $qp = \theta(qp)$, which, due to Theorem 3.21, leads to one of the following two cases. First, if $p = t^kt_1$ and $q = \theta(t_1)t^j$, where $k, j \geq 0$ and $t = t_1\theta(t_1)$ is the primitive root of pq, then we obtain that $v = t^{(k+j+1)(i+1)+k}t_1$ with $(k+j+1)(i+1)+k \geq 1$, which contradicts the θ-primitivity of v. Second, if $\rho(p) = \rho(q) = t$, then also $v \in \{t\}^*$ where $t = \theta(t)$. Thus, $v = \theta(v)$, and the initial identity becomes $v^2x = yv^2$. However, since v is θ-primitive and thus also primitive, we immediately obtain, due to Proposition 3.2, that $x = y = \epsilon$. $\square$

In other words, the previous result states that if v is a θ-primitive word, then $\theta(v)v$ cannot overlap with $v\theta(v)$ in a nontrivial way. However, the following example shows that this is not the case anymore if we look at the overlaps between $\theta(v)v$ and v^2, or between $v\theta(v)$ and v^2, respectively, even if we consider the larger class of primitive words.

Example 7. Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ, $p, q \in \Sigma^+$ such that $\rho(p) \neq \rho(q)$, $p = \theta(p)$, and $q = \theta(q)$, and let $v = p^2q^2p$ and $u = pq^2p^2$. It is easy to see that u and v are primitive words. In addition, if we take $\Sigma = \{a, b\}$, the mapping θ to be the mirror image, $p = a$, and $q = b$, then u and v are actually θ-primitive words. Since $\theta(v) = pq^2p^2$ and $\theta(u) = p^2q^2p$, we can write $xv^2 = v\theta(v)y$ and $y\theta(u)u = u^2z$ where $x = p^2q^2$, $y = pq^2p$, and $z = q^2p^2$. Thus, for primitive (resp. θ-primitive) words u and v, $v\theta(v)$ can overlap with v^2 and $\theta(u)u$ with u^2 in a nontrivial way.

Maybe even more surprisingly, the situation changes again if we try to fit v^2 inside $v\theta(v)v$, as shown by the following result.

Theorem 3.24. *Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution over the alphabet Σ and $v \in \Sigma^+$ be a primitive word. If $v\theta(v)v = xv^2y$ for some words $x, y \in \Sigma^*$, then v is θ-palindrome and either $x = \epsilon$ and $y = v$ or $x = v$ and $y = \epsilon$.*

Proof. Suppose that $v\theta(v)v = xv^2y$ for some words $x, y \in \Sigma^*$, as illustrated in

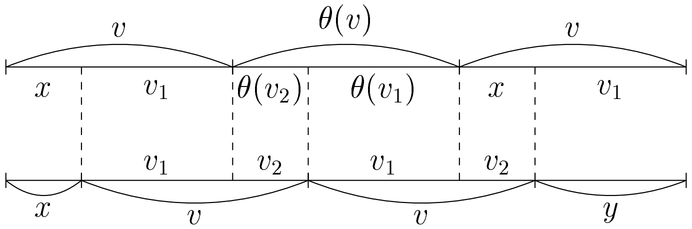

Figure 3.3: The equation $v\theta(v)v = xv^2y$

Figure 3.3.

If we look at this identity from left to right, then we can write $v = xv_1 = v_1v_2$, with $v_1, v_2 \in \Sigma^*$ such that $|x| = |v_2|$ and $\theta(v) = \theta(v_2)\theta(v_1)$. Then, if we look at the right sides of this identity, then we immediately obtain that $x = v_2$ and $v_1 = y$. Thus, $v = xy = yx$, implying that $x, y \in \{t\}^*$, for some primitive word t. However, since v is primitive, this means that either $x = \epsilon$ and $y = v$ or $x = v$ and $y = \epsilon$. Moreover, in both cases we also obtain $v = \theta(v)$. $\square$

3.6 A shorter bound for the Fine and Wilf theorem (antimorphic case)

Throughout this section we take $\theta : \Sigma^* \to \Sigma^*$ to be an antimorphic involution, $u, v \in \Sigma^+$ with $|u| > |v|$, $\alpha(u, \theta(u))$ be a θ-power of u, and $\beta(v, \theta(v))$ be a θ-power of v. Recall that $\alpha(u, \theta(u))$ starts with u and $\beta(v, \theta(v))$ starts with v. We start our analysis with the case when v is θ-palindrome.

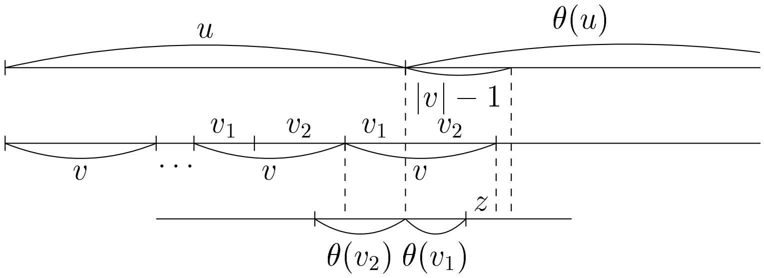

Figure 3.4: The common prefix of $u\theta(u)$ and v^n of length $|u| + |v| - 1$

Theorem 3.25. *Let u and v be two words with $|u| > |v|$ and $v = \theta(v)$. If there exist two θ-powers $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $|u| + |v| - \gcd(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. First, we can suppose, without loss of generality that $\gcd(|u|, |v|) = 1$. Otherwise, i.e., $\gcd(|u|, |v|) = d \geq 2$, we consider a new alphabet $\Sigma' = \Sigma^d$, where the new letters are words of length d in the original alphabet, and we look at the words u and v as elements of $(\Sigma')^+$. In the larger alphabet $\gcd(|u|, |v|) = 1$, and if we can prove the theorem there it immediately gives the general proof.

Since $v = \theta(v)$, $\beta(v, \theta(v)) = v^n$ for some $n \geq 2$. Moreover, if $v \in \Sigma$, then trivially $u \in v\{v, \theta(v)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. So, suppose next that $|v| \geq 2$ and, since $\gcd(|u|, |v|) = 1$, $u = v^i v_1$, where $i \geq 1$ and $v = v_1 v_2$ with $v_1, v_2 \in \Sigma^+$.

If $\alpha(u, \theta(u)) = u^2\alpha'(u, \theta(u))$, then u^2 and v^n have a common prefix of length at least $|u|+|v|-\gcd(|u|, |v|)$, which, due to Theorem 3.8, implies that $\rho(u) = \rho(v) = t$, for some primitive word $t \in \Sigma^+$, and thus $\rho_\theta(u) = \rho_\theta(t) = \rho_\theta(v)$.

Otherwise, $\alpha(u, \theta(u)) = u\theta(u)\alpha'(u, \theta(u))$ for some $\alpha'(u, \theta(u)) \in \{u, \theta(u)\}^*$. Now, we have two cases depending on $|v_1|$ and $|v_2|$. We present here only the case when $|v_1| \leq |v_2|$, see Figure 3.4, the other one being symmetric. Now, since θ is an antimorphism, $\theta(\mathrm{suff}_{|v|-1}(u)) = \mathrm{pref}_{|v|-1}(\theta(u))$. So, we can write $v_2 = \theta(v_1)z$ for some $z \in \Sigma^*$, since $|v_1| \leq |v_2| \leq |v| - 1 = |v| - \gcd(|u|, |v|)$. Now, to the left of the border-crossing v there is at least one occurrence of another v, so we immediately obtain $z = \theta(z)$, as $v_2 = \theta(v_1)z$ and $\theta(v_2) = \theta(z)v_1$. Then, $v = v_1\theta(v_1)z = zv_1\theta(v_1) = \theta(v)$ which implies, due to Theorem 3.18, that $\rho_\theta(v_1) = \rho_\theta(z)$. So, since $v = v_1\theta(v_1)z$ and $u = v^i v_1 = (v_1\theta(v_1)z)^i v_1$, we obtain $\rho_\theta(u) = \rho_\theta(v)$. $\square$

Let us look next at the case when u is θ-palindrome.

Theorem 3.26. *Let u and v be two words with $|u| > |v|$ and $u = \theta(u)$. If there exist two θ-powers $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $|u| + |v| - \gcd(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. As in the previous proof, we can suppose without loss of generality that $\gcd(|u|, |v|) = 1$. Also, since $u = \theta(u)$, we actually have $\alpha(u, \theta(u)) = u^n$ for some $n \geq 2$. Moreover, since u starts with v and $u = \theta(u)$, we also know that u ends with $\theta(v)$. Now, if $v \in \Sigma$, then trivially $u \in v\{v, \theta(v)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. So, we can suppose next that $|v| \geq 2$ and thus, since $\gcd(|u|, |v|) = 1$, we have $u = \beta'(v, \theta(v))v'$, where $\beta'(v, \theta(v))$ is a prefix of $\beta(v, \theta(v))$ and $v' \in \Sigma^+$, $v' \in \mathrm{Pref}(v) \cup \mathrm{Pref}(\theta(v))$.

Case 1: We begin our analysis with the case when the border between the first two u's falls inside a v, as illustrated in Figure 3.5. Then, we can write $v = v_1v_2 =$

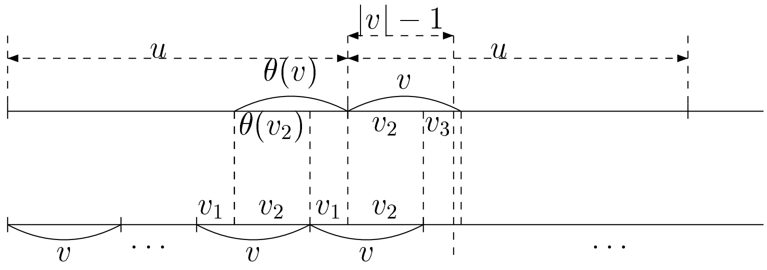

Figure 3.5: The common prefix of u^2 and $\beta(v, \theta(v))$ of length $|u| + |v| - 1$

$v_2 v_3$ where $v_1, v_2, v_3 \in \Sigma^+$, implying that $v_1 = xy$, $v_3 = yx$, and $v_2 = (xy)^j x$ for some $j \geq 0$ and $x, y \in \Sigma^*$. Moreover, since u ends with $\theta(v)$, we also have $v_1 = \theta(v_1)$, i.e., $xy = \theta(y)\theta(x)$. If $x = \epsilon$, then $v_1, v_2, v_3, v \in \{y\}^*$, which implies that also $u \in y\{y, \theta(y)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v) = \rho_\theta(y)$; moreover, since $\gcd(|u|, |v|) = 1$ we actually must have $y \in \Sigma$. Similarly, we also obtain $\rho_\theta(u) = \rho_\theta(v)$ when $y = \epsilon$. So, from now on we can suppose that $x, y \in \Sigma^+$.

Let us consider next the case when, before the border-crossing v we have an occurrence of another v, as illustrated in Figure 3.5. Then, we have that $v_2 = \theta(v_2)$, i.e., $(xy)^j x = (\theta(x)\theta(y))^j \theta(x)$. If $j \geq 1$, then this means that $x = \theta(x)$ and $y = \theta(y)$. Then, the equality $xy = \theta(y)\theta(x)$ becomes $xy = yx$. So, there exists a word $t \in \Sigma^+$ such that $x, y \in \{t\}^*$, and thus also $v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, $j = 0$ and we have $x = \theta(x)$. But then, the equality $xy = \theta(y)\theta(x)$ becomes $xy = \theta(y)x$, implying that $x = p(qp)^n$ and $y = (qp)^m$ for some $m \geq 1$, $n \geq 0$, and some words p and q with $p = \theta(p)$ and $q = \theta(q)$, see [8]. Since u^2 and $\beta(v, \theta(v))$ share a common prefix of length at least $|u| + |v| - \gcd(|u|, |v|) =$

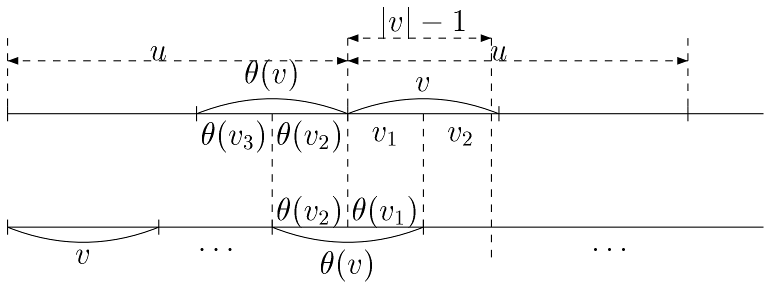

Figure 3.6: The common prefix of u^2 and $\beta(v, \theta(v))$ of length $|u| + |v| - 1$

$|u| + |v| - 1$, v_3 and some $\beta'(v, \theta(v))$ share a prefix of length $|v_3| - 1$. Furthermore, as $v_3 = yx = (qp)^m p(qp)^n$, $v = v_1 v_2 = p(qp)^{m+n} p(qp)^n$, and $\theta(v) = (pq)^n p(pq)^{m+n} p$, this means that independently of what follows to the right the border-crossing v, either v or $\theta(v)$, we have two expressions over p and q sharing a common prefix of length at least $|p| + |q|$. So, due to Corollary 3.5, $p, q \in \{t\}^*$ for some $t \in \Sigma^+$, which implies that also $x, y, v \in \{t\}^+$ and $u \in \{t, \theta(t)\}^+$, i.e., $\rho_\theta(u) = \rho_\theta(v)$.

Now, suppose that before the border-crossing v we have an occurrence of $\theta(v)$. If $|u| < 2|v| + |v_1|$, then, since $\beta(v, \theta(v))$ starts with v, we must have $v = \theta(v)$, in which case we can use Theorem 3.25 to conclude that $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, $|u| \geq 2|v| + |v_1|$ and since $u = \theta(u)$, u ends either with $v\theta(v)$ or with $\theta(v)\theta(v)$. In the first case, we obtain $v_3 = \theta(v_3)$, i.e., $yx = \theta(yx)$, which together with $xy = \theta(xy)$ imply, due to Corollary 3.22, that $x, y \in \{t, \theta(t)\}^*$, for some $t \in \Sigma^+$ and thus, $\rho_\theta(u) = \rho_\theta(v)$. In the second case, we obtain $v_1 = v_3$, i.e., $xy = yx$. So, $x, y \in \{t\}^*$, and thus also $v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$.

Case 2: Let us consider now the case when the border between the first two u's falls inside $\theta(v)$, as illustrated in Figure 3.6. Then, we can write again $v = v_1 v_2 = v_2 v_3$ where $v_1, v_2, v_3 \in \Sigma^+$, which implies that $v_1 = xy$, $v_3 = yx$, and $v_2 = (xy)^j x$ for some $j \geq 0$ and $x, y \in \Sigma^*$. Just as before, if $x = \epsilon$ or $y = \epsilon$, we immediately obtain that $\rho_\theta(u) = \rho_\theta(v)$. So, we can suppose that $x, y \in \Sigma^+$. Moreover, $v_1 = \theta(v_1)$, i.e., $xy = \theta(xy)$. Now, if the border-crossing $\theta(v)$ is preceded by an occurrence of v, then we also have $v_3 = \theta(v_3)$, i.e., $yx = \theta(yx)$. Then, due to Corollary 3.22, there exists some $t \in \Sigma^+$ such that $x, y \in \{t, \theta(t)\}^*$, implying that $\rho_\theta(u) = \rho_\theta(v)$, since $v = (xy)^{j+1}x$ and $u = \beta'(v, \theta(v))\theta(v_2)$. If, on the other hand, the border-crossing $\theta(v)$ is preceded by another $\theta(v)$, then we immediately obtain $v_1 = v_3$, i.e., $xy = yx$. So, $x, y \in \{t\}^*$, for some $t \in \Sigma^+$, and thus also $v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. □

Although the previous two results give a very short bound, i.e., $|u| + |v| - \gcd(|u|, |v|)$, this is not enough in the general case, as illustrated in Example 3. Nevertheless, we can prove that, independently of how the θ-power $\alpha(u, \theta(u))$ starts, $2|u| + |v| - \gcd(|u|, |v|)$ is enough to impose $\rho_\theta(u) = \rho_\theta(v)$. The first case we consider is when $\alpha(u, \theta(u))$ starts with u^2.

Theorem 3.27. *Given two words $u, v \in \Sigma^+$ with $|u| > |v|$, if there exist two θ-powers $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $2|u| + |v| - \gcd(|u|, |v|)$ and, moreover, $\alpha(u, \theta(u)) = u^2\alpha'(u, \theta(u))$ for some $\alpha'(u, \theta(u)) \in \{u, \theta(u)\}^+$, then $\rho_\theta(u) = \rho_\theta(v)$.*

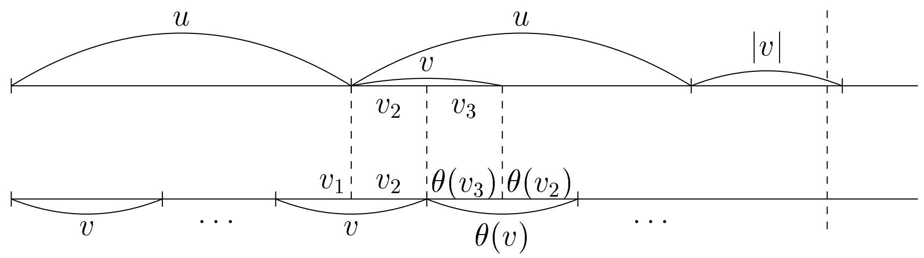

Figure 3.7: The prefix of $u^2\alpha'(u,\theta(u))$ and $\beta(v,\theta(v))$ of length $2|u|+|v|-1$

Proof. Just as we did before, we can suppose, without loss of generality, that $\gcd(|u|,|v|)=1$. Now, if $v\in\Sigma$, then trivially $u\in v\{v,\theta(v)\}^*$, i.e., $\rho_\theta(u)=\rho_\theta(v)$. So, we can suppose next that $|v|\geq 2$ and thus, since $\gcd(|u|,|v|)=1$, we have $u=\beta'(v,\theta(v))v'$, where $\beta'(v,\theta(v))$ is a prefix of $\beta(v,\theta(v))$ and $v'\in\Sigma^+$ is a prefix of either v or $\theta(v)$.

<u>*Case 1*</u>*:* Let us look first at the case when the border between the first two u's falls inside v, i.e., $u=\beta'(v,\theta(v))v_1$ for some $v_1\in\Sigma^+$ such that $v=v_1v_2$ and $\beta'(v,\theta(v))\in v\{v,\theta(v)\}^*$ is a prefix of $\beta(v,\theta(v))$. Moreover, if this border-crossing v is followed to the right by another v, then $v^2=v_1vv_2$, since $\mathrm{pref}_{|v|}(u)=v$. Thus, $v_1v_2=v_2v_1$, meaning that there exists a primitive word $t\in\Sigma^+$ such that $v_1,v_2\in\{t\}^+$ and thus $v\in\{t\}^+$. Moreover, since $u=\beta'(v,\theta(v))v_1$, we also have $u\in t\{t,\theta(t)\}^*$, i.e., $\rho_\theta(u)=\rho_\theta(v)$. Otherwise, the border-crossing v is followed to the right by $\theta(v)$, as illustrated in Figure 3.7. Thus, we can write $v=v_1v_2=v_2v_3$ with $v_1,v_2,v_3\in\Sigma^+$, $|v_1|=|v_3|$, and $v_3=\theta(v_3)$. But then, Theorem 3.6 implies that there exist some $i\geq 0$ and some $x,y\in\Sigma^*$ such that $v_1=xy$, $v_3=yx$, $v_2=(xy)^ix$,

and $v = (xy)^{i+1}x$. If $x = \epsilon$, then we have that $v_1, v_2, v_3, v \in \{y\}^+$, which implies that also $u \in y\{y, \theta(y)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Similarly, we also obtain $\rho_\theta(u) = \rho_\theta(v)$ when $y = \epsilon$. So, from now on we can suppose that $x, y \in \Sigma^+$.

Suppose first that $i \geq 1$. If we take $|\beta'(v, \theta(v))| = k|v|$ with $k \geq 1$, then the length of the first u is $|u| = k|v| + |v_1| = k(i+1)|xy| + k|x| + |xy|$. Since the second u starts with $v_2 = (xy)^i x$, using length arguments, we must have that its right end will fall inside either v or $\theta(v)$, after exactly $2|xy|$ characters. If the right end of the second u falls inside $\theta(v) = (yx)^{i+1}\theta(x)$, then $\mathrm{suff}_{|xy|}(u) = yx$. But, the first u ended with $v_1 = xy$. So, $xy = yx$, which implies that there exists a primitive word $t \in \Sigma^+$ such that $x, y \in \{t\}^*$, and thus also $v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(t) = \rho_\theta(v)$. Otherwise, the right end of the second u falls inside v, i.e., $\mathrm{suff}_{2|xy|}(u) = xyxy$. Actually, depending on what precedes to the left this second border-crossing v, either v or $\theta(v)$, we have $\mathrm{suff}_{|x|+2|xy|}(u) \in \{xxyxy, \theta(x)xyxy\}$. Next, we look at the suffix of the first u and we have again two cases depending on what precedes the first border-crossing v. If there is a v to the left of this border-crossing v, then $\mathrm{suff}_{|x|+2|xy|}(u) = xyxv_1$, and thus we obtain immediately that $xy = yx$. So, in this case there exists a primitive word $t \in \Sigma^+$, such that $v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, there is a $\theta(v)$ to the left of the border-crossing v, i.e., $\mathrm{suff}_{|x|+2|xy|}(u) = yx\theta(x)v_1$. Thus, in this case we obtain that either $yx\theta(x) = xxy$ or $yx\theta(x) = \theta(x)xy$. However, in both cases, due to Theorems 3.19 and 3.20, we obtain $x, y \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$, which

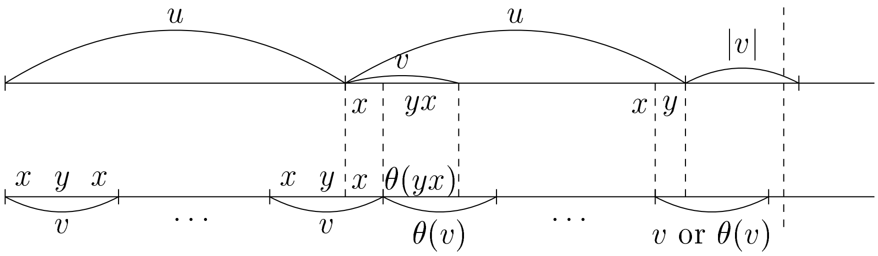

Figure 3.8: The prefix of $u^2\alpha'(u,\theta(u))$ and $\beta(v,\theta(v))$ of length $2|u|+|v|-1$

immediately implies $\rho_\theta(v) = \rho_\theta(u)$.

Suppose next that $i = 0$, i.e., $v_1 = xy$, $v_3 = yx$, $v_2 = x$, $v = xyx$, and $\theta(yx) = yx$, as illustrated in Figure 3.8. Now, if we compute the length of the first u, then we have $|u| = k|v| + |xy|$ for some $k \geq 1$. Since the second u starts with $v_2 = x$, we must have that its right end will fall inside either v or $\theta(v)$, after exactly $|y|$ characters. Now, we have two cases depending on what occurs to the left of this second border-crossing point.

Firstly, if there is a v occurring before this border-crossing point, then $\mathrm{suff}_{2|xy|}(u) = xyxy$. Next, we turn again to look at the suffix of the first u. Depending on whether there is v or $\theta(v)$ to the left of the first border-crossing v, we have $\mathrm{suff}_{2|xy|}(u) \in \{yxxy, \theta(y)\theta(x)xy\}$. Thus, either $yx = xy$ or $\theta(xy) = xy$. However, since also $\theta(yx) = yx$, we obtain that either $x, y \in \{t\}^*$ or $x, y \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$, and thus $\rho_\theta(u) = \rho_\theta(v)$.

Secondly, if $\theta(v) = \theta(x)\theta(y)\theta(x)$ occurs to the left of the second border-crossing point, since $\mathrm{suff}_{|xy|}(u) = v_1 = xy$, then we obtain immediately that $x = \theta(x)$. But,

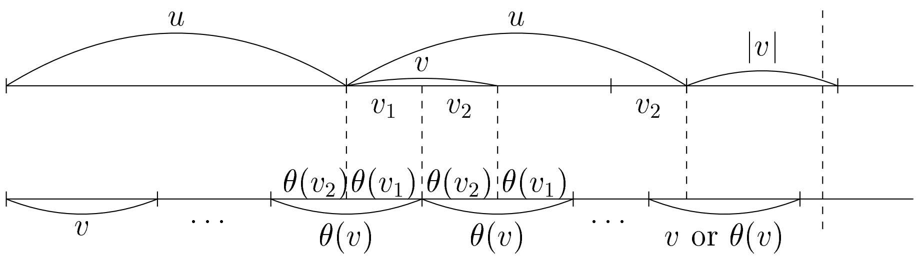

Figure 3.9: The prefix of $u^2\alpha'(u,\theta(u))$ and $\beta(v,\theta(v))$ of length $2|u|+|v|-1$

we already knew that $yx = \theta(yx)$, i.e., $yx = x\theta(y)$, which implies $x = p(qp)^j$ and $y = (pq)^k$ for some $j \geq 0$, $k \geq 1$, and some words p and q such that $p = \theta(p)$ and $q = \theta(q)$, see [8]. Now, since $\alpha(u,\theta(u))$ and $\beta(v,\theta(v))$ have a common prefix of length $2|u|+|v|-\gcd(|u|,|v|) = 2|u|+|v|-1$, we can also look at the prefix of length $|v|-1$ of the third word from $\alpha(u,\theta(u))$, which is either u or $\theta(u)$. However, in all cases, after we reduce the common prefix, we have two distinct expressions over p and q of length longer than $|p|+|q|$, which implies, due to Corollary 3.5, that $pq = qp$. Thus, also in this case $\rho_\theta(u) = \rho_\theta(v)$.

Case 2: Consider now the case when the border between the first two u's falls inside $\theta(v)$. If this border-crossing $\theta(v)$ is followed to the right by another $\theta(v)$, as illustrated in Figure 3.9, then there exist some $v_1, v_2 \in \Sigma^+$ such that $v = v_1v_2$, $v_1 = \theta(v_1)$, and $v_2 = \theta(v_2)$. Thus, obviously $v, \theta(v), u, \theta(u) \in \{v_1, v_2\}^+$, i.e., $\alpha(u,\theta(u))$ and $\beta(v,\theta(v))$ are actually two expressions over $\{v_1, v_2\}$ having a common prefix of length $2|u|+|v|-\gcd(|u|,|v|) = 2|u|+|v|-1$. Moreover, since $|u| = k|v|+|v_2|$ for some $k \geq 1$ and the second u begins with v_1, its right end cuts a v or $\theta(v)$ after exactly

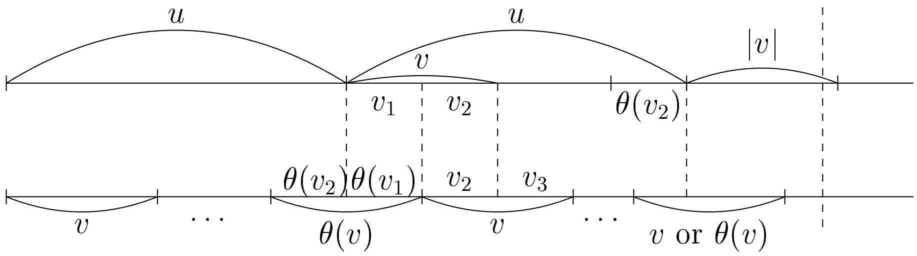

Figure 3.10: The prefix of $u^2\alpha'(u, \theta(u))$ and $\beta(v, \theta(v))$ of length $2|u| + |v| - 1$

$(2|v_2| \bmod |v|) \neq |v_2|$ characters. Thus, the two expressions over $\{v_1, v_2\}$ have to differ at some point, and moreover, after we eliminate the common prefix we remain with two distinct expressions over v_1 and v_2 of length longer than $|v_1| + |v_2|$, which implies, due to Corollary 3.5, that $v_1v_2 = v_2v_1$. Thus, also in this case $\rho_\theta(u) = \rho_\theta(v)$.

Hence, the border-crossing $\theta(v)$ is followed to the right by v, as illustrated in Figure 3.10. Then, we can write $v = v_1v_2 = v_2v_3$ for some $v_1, v_2, v_3 \in \Sigma^+$ with $|v_1| = |v_3|$ and $v_1 = \theta(v_1)$. Thus, due to Theorem 3.6, there exist some words $x, y \in \Sigma^*$ and some $i \geq 0$ such that $v_1 = xy$, $v_3 = yx$, $v_2 = (xy)^i x$, and $v = (xy)^{i+1}x$. Again, if either $x = \epsilon$ or $y = \epsilon$, then we obtain immediately that $\rho_\theta(u) = \rho_\theta(v)$. So, from now on, we can suppose that $x, y \in \Sigma^+$. Moreover, since u ends with $\theta(v_2)$, we also know that $\theta(u)$ starts with $v_2 = (xy)^i x$.

Suppose first that $i \geq 1$. Then, the length of the first u is $|u| = k|v| + |v_2| = k|v| + i|xy| + |x|$ for some $k \geq 1$. Since the second u starts with $\theta(v_1) = xy$, its right end will cut either v or $\theta(v)$ after exactly $|x| + (i-1)|xy|$ characters. If this second border point falls inside v, since both u and $\theta(u)$ start with xy, we obtain

$xy = yx$. That is, there exists a primitive word $t \in \Sigma^+$ such that $x, y, v \in \{t\}^+$ and $u \in t\{t, \theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, this second border point cuts $\theta(v) = \theta(x)(xy)^{i+1}$ after exactly $|x| + (i-1)|xy|$ characters. Then, since u ends with $\theta(v_2) = \theta(x)(xy)^i$, depending on whether to the left of this second border-crossing $\theta(v)$ we have either v or $\theta(v)$, we obtain either $yx\theta(x) = \theta(x)xy$ or $xy\theta(x) = \theta(x)xy$. In the first case, Theorem 3.19 implies $x, y \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$, while in the latter one we obtain $x = \theta(x)$ and $\rho(x) = \rho(y)$. Since $v = (xy)^{i+1}x$ and $u = \beta'(v, \theta(v))\theta(v_2)$, we conclude again that $\rho_\theta(u) = \rho_\theta(v)$.

Otherwise, we have $i = 0$, i.e., $v_1 = xy$, $v_3 = yx$, $v_2 = x$, $v = xyx$, and $\theta(v) = \theta(x)xy$. Using again length arguments, we notice that the right end of the second u cuts either v or $\theta(v)$ after exactly $2|x|$ characters.

Let us look first at the case when this second border point falls inside $\theta(v)$. Then $x = \theta(x)$, as u ends with $\theta(v_2) = \theta(x)$. Since $\alpha(u, \theta(u))$ and $\beta(v, \theta(v))$ have a common prefix of length $2|u| + |v| - \gcd(|u|, |v|) = 2|u| + |v| - 1$, we can also look at the prefix of length $|v| - 1$ of the third word from $\alpha(u, \theta(u))$, which is either u or $\theta(u)$. Since u ends with $\theta(v_2) = \theta(x)$, we know that both u and $\theta(u)$ start with x. Furthermore, since $\theta(xy) = xy$, we actually have two distinct expressions over $\{x, y\}^+$, one starting with x and the other with y, having a common prefix longer than $|x| + |y|$, implying, due to Corollary 3.5, that $xy = yx$. So, also in this case $\rho_\theta(u) = \rho_\theta(v)$.

Next, we turn to the case when the second border point falls inside v and we

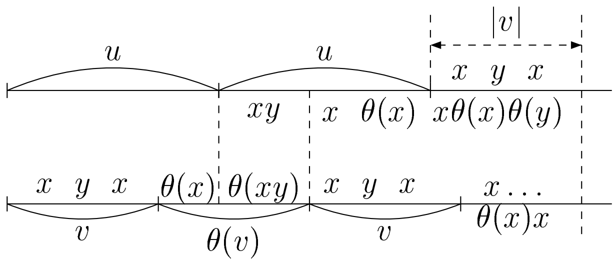

Figure 3.11: The prefixes of $u^2\alpha'(u,\theta(u))$ and $\beta(v,\theta(v))$ of length $2|u|+|v|$

analyze two cases depending on the length of u. Firstly, if $|u| > 2|v|$, then the first u starts either with v^2 or with $v\theta(v)$ and we look at the prefix of the second u, see Figure 3.10. In the former case, we obtain immediately that $xy = yx$, which implies that there exists a primitive word $t \in \Sigma^+$ such that $x, y, v \in \{t\}^+$ and $u \in t\{t,\theta(t)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. In the latter case, we obtain $yx = \theta(yx)$, which together with $xy = \theta(xy)$ implies, due to Corollary 3.22, that $x, y \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$, and thus also $\rho_\theta(u) = \rho_\theta(v)$.

Secondly, if $|u| < 2|v|$, then we actually must have $u = v\theta(v_2) = xyx\theta(x)$, as illustrated in Figure 3.11. Since $\alpha(u,\theta(u))$ and $\beta(v,\theta(v))$ have a common prefix of length $2|u|+|v|-1$, after eliminating the common prefix, we obtain one of the following four equations, depending on whether the third block of $\alpha(u,\theta(u))$ is u or $\theta(u)$, and the fourth block of $\beta(v,\theta(v))$ is v or $\theta(v)$.

- If we have $\theta(x)xy\ \mathrm{pref}_{|x|-1}(x) = yxx\ \mathrm{pref}_{|x|-1}(yx)$, then $\mathrm{pref}_{|x|}(yx) = \theta(x)$, and thus we obtain $\mathrm{pref}_{|x|-1}(x) = \mathrm{pref}_{|x|-1}(\theta(x))$. Now, if we denote $x = x_1 \ldots x_n$ with $x_1, \ldots, x_n \in \Sigma$, then the equation $\mathrm{pref}_{|x|-1}(x) = \mathrm{pref}_{|x|-1}(\theta(x))$ becomes

$x_1 \ldots x_{n-1} = \theta(x_n) \ldots \theta(x_2)$. Depending on whether $|x|$ is even or odd, this equality implies $x = x_1 \ldots x_k \theta(x_1 \ldots x_k)$ or $x = x_1 \ldots x_k x_{k+1} \theta(x_1 \ldots x_k)$ with $x_{k+1} = \theta(x_{k+1})$. However, on both cases, we obtain $x = \theta(x)$. Then, from the initial equation $\theta(x)xy\ \mathrm{pref}_{|x|-1}(x) = yxx\ \mathrm{pref}_{|x|-1}(yx)$ we obtain $x^2 y = yx^2$, which implies $\rho(x) = \rho(y)$. Hence, also $\rho_\theta(u) = \rho_\theta(v)$.

- If $\theta(x)xy = yx\theta(x)$, then, due to Theorem 3.19, we immediately obtain $x, y \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$, and thus $\rho_\theta(u) = \rho_\theta(v)$.

- If $\theta(x)x\ \mathrm{pref}_{|xy|-1}(\theta(x)\theta(y)) = yxx\ \mathrm{pref}_{|x|-1}(yx)$, then we can write $yx = \theta(x)z$ for some word $z \in \Sigma^+$ with $|z| = |y|$. If we substitute this equation into the initial one, we obtain $x\theta(z)\ \mathrm{pref}_{|x|-1}(x) = zx\ \mathrm{pref}_{|x|-1}(\theta(x))$, which implies that $x_1 \ldots x_{n-1} = \theta(x_n) \ldots \theta(x_2)$, where $x = x_1 \ldots x_n$ with $x_1, \ldots, x_n \in \Sigma$. Just as before we can derive again $x = \theta(x)$. Since $xy = \theta(xy)$, we can write $xy = \theta(y)x$ which implies that $x = p(qp)^j$ and $y = (qp)^k$, for some $j \geq 0$, $k \geq 1$, and some words p and q such that $p = \theta(p)$ and $q = \theta(q)$, see [8]. Then, using these relations, the initial equation becomes a nontrivial identity over p and q of length more than $|p| + |q|$. Thus, due to Corollary 3.5, there exists a primitive word t such that $p, q, x, y \in \{t\}^+$. So, $\rho_\theta(u) = \rho_\theta(v)$.

- If $\theta(x)x\ \mathrm{pref}_{|xy|-1}(\theta(x)\theta(y)) = yx\theta(x)\ \mathrm{pref}_{|x|-1}(x)$, then we can write again $yx = \theta(x)z$ for some word $z \in \Sigma^+$ with $|z| = |y|$. Thus, the initial equation becomes $x\theta(z) = z\theta(x)$. If in the equation $xy = \theta(y)\theta(x)$ we concatenate $x\theta(x)$ both to the left and to the right, then we derive $x\theta(x)xyx\theta(x) = x\theta(yx)\theta(x)x\theta(x)$. Substituting $yx = \theta(x)z$ and $\theta(yx) = \theta(z)x$, we derive $x\theta(x)x\theta(x)z\theta(x) = x\theta(z)x\theta(x)x\theta(x)$. Now, since $x\theta(z) = z\theta(x)$, this becomes $(x\theta(x))^2x\theta(z) = x\theta(z)(x\theta(x))^2$, which implies that there exists a primitive word $t \in \Sigma^+$ such that $x\theta(x), x\theta(z) \in \{t\}^*$. If $x\theta(x) = t^{2j}$ for some $j \geq 0$, then $x = \theta(x) = t^j$, $t = \theta(t)$, $z, y \in \{t\}^*$, and thus $\rho_\theta(u) = \rho_\theta(v)$. If $x\theta(x) = t^{2j+1}$ for some $j \geq 1$, then we actually have $x = t^j t_1$, $\theta(x) = \theta(t_1)t^j$, and $\theta(z) = \theta(t_1)t^k$ where $t = t_1\theta(t_1)$ and $k \geq 0$. Now, from the equation $yx = \theta(x)z$ we also obtain that $y \in \{t_1, \theta(t_1)\}^+$. So, also in this case we can conclude that $\rho_\theta(u) = \rho_\theta(v)$.

$\square$

Next, let us look at the case when $\alpha(u, \theta(u))$ starts with $u\theta(u)u$.

Theorem 3.28. *Given two words $u, v \in \Sigma^+$ with $|u| > |v|$, if there exist two θ-powers $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $2|u|+|v|-\gcd(|u|, |v|)$ and, moreover, $\alpha(u, \theta(u)) = u\theta(u)u\alpha'(u, \theta(u))$ with $\alpha'(u, \theta(u)) \in \{u, \theta(u)\}^*$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. Let us suppose again, just as we did before, that $\gcd(|u|, |v|) = 1$. If we denote $u' = u\theta(u)$, then $u'u'$ and $\beta(v, \theta(v))$ have a common prefix of length $|u'|+|v|-\gcd(|u'|, |v|) = |u'| + |v| - 1$ and, moreover, $u' = \theta(u')$. Thus, due to Theorem 3.26, $\rho_\theta(v) = \rho_\theta(u')$; let this θ-primitive root be t. Then, $u\theta(u) = \gamma(t, \theta(t))$, for some

θ-power $\gamma(t, \theta(t)) \in t\{t, \theta(t)\}^+$, which implies, due to Theorem 3.14, that $\rho_\theta(u) = t = \rho_\theta(v)$. □

The only case which remains to be considered now is when $\alpha(u, \theta(u))$ starts with $u\theta(u)\theta(u)$. Next, we give two intermediate results concerning θ-palindromic words, which will be very helpful in the proof of Theorem 3.31.

Lemma 3.29. *Let $w \in \Sigma^+$ and x, y, z be θ-palindromes. If $w = xy = yz$, then there exists a θ-palindromic primitive word $p \in \Sigma^+$ such that $w, x, y, z \in \{p\}^+$.*

Proof. Since $w = xy$ with $x = \theta(x)$ and $y = \theta(y)$, we know from [10], that there exist two θ-palindrome words p, q and an integer $n \geq 1$ such that $w = (pq)^n$, where pq is a primitive word, $p \neq \epsilon$, $x = (pq)^i p$, $y = q(pq)^{n-i-1}$, $y = (pq)^j p$, and $z = q(pq)^{n-j-1}$ for some integers $0 \leq i, j < n$. If $n-i-1, j \geq 1$, then $pq, qp \in \text{Pref}(y)$, i.e., $pq = qp$. Since pq is primitive, this means that $q = \epsilon$. Therefore, p is a primitive word and $w, x, y, z \in \{p\}^+$. If $n - i - 1 \geq 1$ and $j = 0$, then $q(pq)^{n-i-1} = p$, which implies that $n - i - 1 = 1$ and hence $q = \epsilon$, and we reached the same conclusion as above. If $n - i - 1 = 0$ and $j \geq 1$, then $q = (pq)^j p$, which cannot hold for any $j \geq 1$ because $p \neq \epsilon$. If both $n - i - 1$ and j are 0, then $p = q$, which contradicts the primitivity of pq. □

Lemma 3.30. *Let $w \in \Sigma^+$ and x, y, z be θ-palindromes. If $w = xy^2 = yz$, then there exists a θ-palindrome primitive word $p \in \Sigma^+$ such that $w, x, y, z \in \{p\}^+$.*

Proof. Since $w = yz$ with $y = \theta(y)$ and $z = \theta(z)$, we know from [10], that there exist two θ-palindrome words p, q and an integer $n \geq 1$ such that $w = (pq)^n$, where pq is a primitive word, $p \neq \epsilon$, $x = (pq)^i p$, $y^2 = q(pq)^{n-i-1}$, $y = (pq)^j p$, and $z = q(pq)^{n-j-1}$ for some integers $0 \leq i, j < n$. If $n - i - 1, j \geq 1$, then, just as in the proof of Lemma 3.29, $pq = qp$. Since pq is primitive, $q = \epsilon$, and hence p is primitive and $x, y, z \in \{p\}^+$. If $n - i - 1 \geq 1$ and $j = 0$, then $y = p$. Since $y^2 = q(pq)^{n-i-1}$, we have that $p^2 = q(pq)^{n-i-1}$, which means, due to Theorem 3.3, that $p, q \in \{t\}^*$ for some primitive word t. Since pq is primitive, this implies that $q = \epsilon$, $p = t$, and $x, y, z \in \{p\}^+$. If $n - i - 1 = 0$ and $j \geq 1$, then $y^2 = q$ and $y = (pq)^j p$, which are clearly contradictory. If both $n - i - 1$ and j are 0, then $y^2 = q$ and $y = p$, which contradicts the primitivity of pq. $\square$

Now, we can state the following result which considers the last case of our analysis.

Theorem 3.31. *Let $u, v \in \Sigma^+$ be two words with $|u| > |v|$. If there exist two θ-powers $\alpha(u, \theta(u)) \in u\theta(u)^2\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $2|u| + |v| - \gcd(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. Once again, we can suppose that $\gcd(|u|, |v|) = 1$ without loss of generality. If $v \in \Sigma$, then trivially $u \in v\{v, \theta(v)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. So, we can suppose next that $|v| \geq 2$ and thus, since $\gcd(|u|, |v|) = 1$, the end of both of u and $u\theta(u)$ falls inside either v or $\theta(v)$. Let $\beta(v, \theta(v)) = v_1 v_2 \ldots v_n v_{n+1} v_{n+2} \beta'(v, \theta(v))$ with $v_1 = v$,

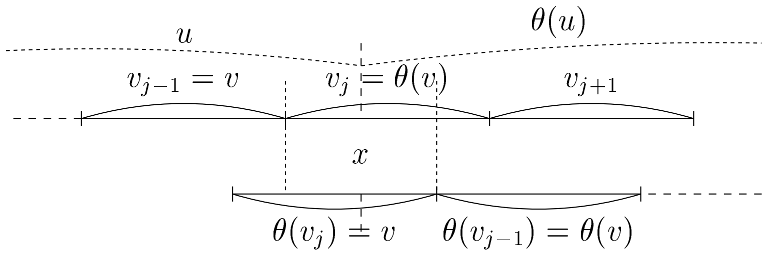

Figure 3.12: The case where n is even, $v_{j-1} = v$, $v_j = \theta(v)$, and $v_{j+1} = v$.

$v_i \in \{v, \theta(v)\}$ for all $2 \leq i \leq n+2$, and $\beta'(v, \theta(v)) \in \{v, \theta(v)\}^*$ such that the end of $u\theta(u)$ falls inside v_{n+1}. Let $u\theta(u) = v_1 \cdots v_n v'$, where $v' \in \text{Pref}(v_{n+1})$. Note that since $u\theta(u)$ is θ-palindrome, $u\theta(u) = \theta(v')\theta(v_n)\cdots\theta(v_1)$. Moreover, the end of u falls inside $v_{(n+2)/2}$ if n is even and inside $v_{(n+1)/2}$ if n is odd. So, from now on we take $j = \frac{n+2}{2}$ whenever n is even and $j = \frac{n+1}{2}$ otherwise, i.e., j is chosen such that the border between u and $\theta(u)$ falls inside v_j.

Let us consider first the case when n is even. Then x, a prefix of v_j, overlaps with a suffix of $\theta(v_j)$, see Figure 3.12, and the overlap implies $x = \theta(x)$. Note that x is a nonempty and proper prefix of v_j.

Now we focus on v_{j-1}, v_j, and v_{j+1}. Even if $j = n$, we can consider $v_{j+1} = v_{n+1} \in \{v, \theta(v)\}$. Suppose $v_{j-1}v_j = v\theta(v)$. If $v_{j+1} = \theta(v)$, then $\theta(v)^2 = x\theta(v)x'$ for some $x' \in \Sigma^+$. This means that $x, \theta(v) \in \{t\}^+$ for some primitive word t. Since $u\theta(u) = v_1 \cdots v_{j-1}x\theta(v_{j-1})\cdots\theta(v_1)$, we obtain that $u, v \in \{t, \theta(t)\}^+$. But $v \in \text{Pref}(u)$, which implies $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, $v_{j+1} = v$. Then $v\theta(v)w = wv\theta(v)$ holds for $w \in \text{Pref}(v)$ with $|w| = |x|$, which implies, due to Theorem 3.18, that

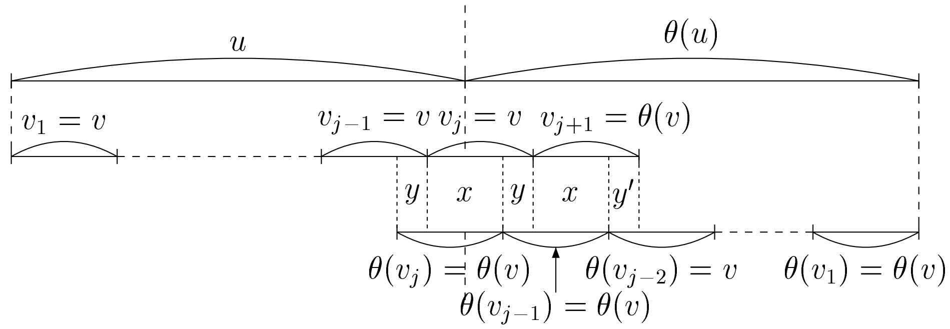

Figure 3.13: The case n being even, $v_{j-1} = v_j = v$, $v_{j+1} = \theta(v)$, and $\theta(v_{j-2}) = v$

$\rho_\theta(v) = \rho_\theta(w) = t$. Then $x \in \{t, \theta(t)\}^+$, and hence $\rho_\theta(u) = \rho_\theta(v)$. The case when $v_{j-1}v_j = \theta(v)v$ also leads to the same conclusion.

Thus, when n is even, only the cases where $v_{j-1}v_j = vv$ or $v_{j-1}v_j = \theta(v)\theta(v)$ remain unsolved yet. Moreover, using exactly the same technique, we can also prove that when n is odd, all we have to consider are the cases when $v_jv_{j+1} = vv$ or $v_jv_{j+1} = \theta(v)\theta(v)$. Although we shall discuss only the case when n is even, a similar result can also be obtained for n odd. Assume that n is even, $v_{j-1}v_j = vv$, and let $v_j = v = xy$ such that $y \in \mathrm{Pref}(\theta(v_{j-1}))$, as illustrated in Figure 3.13. Then we have $x = \theta(x)$ and $y = \theta(y)$.

Next, we claim that once assuming $v_{j-1}v_j = vv$, we only need to consider the case when $v_1v_2 \ldots v_n = v^n$, that is, in all the other cases, we obtain $\rho_\theta(u) = \rho_\theta(v)$. If $j = n$, then we are done. Otherwise, i.e., $j < n$, since $j = (n+2)/2$ we have $n > 2$, and hence also $j > 2$. Thus we can also consider v_{j-2}. Suppose first that $v_{j+1} = \theta(v)$. If $\theta(v_{j-2}) = \theta(v)$, then the nontrivial overlap between $\theta(v)^2$ and $\theta(v)$

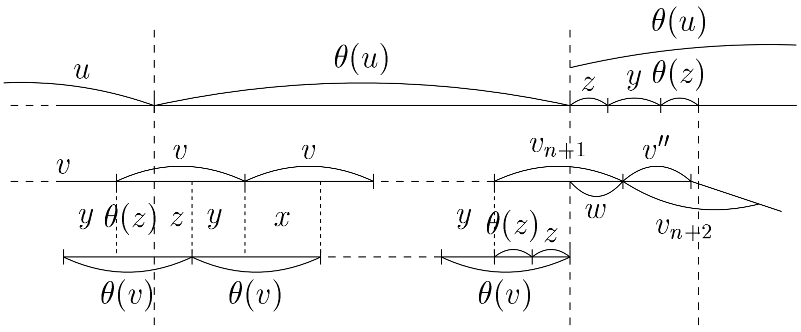

Figure 3.14: The case when n is even and $v_1 = \cdots = v_n = v$.

implies that $\rho(y) = \rho(x) = \rho(\theta(v))$, which, as shown earlier, leads to $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, let $\theta(v_{j-2}) = v$, as illustrated in Figure 3.13. Then, let $v_{j+1} = \theta(v) = xy'$ for some $y' \in \text{Pref}(v)$, which implies that $y' = \theta(y')$. Therefore, $v = xy = y'x$, which implies, due to Lemma 3.29, that $v, x \in \{t\}^+$ for some $t \in \Sigma^+$ and hence $\rho_\theta(u) = \rho_\theta(v)$.

Now suppose that $v_{j+1} = v$, and we consider $\theta(v_{j-2})$. If $j+1 = n$, then $j-2 = 1$ and thus $v_{j-2} = v_1 = v$, i.e., $v_1 v_2 \ldots v_n = v^n$. Otherwise, i.e., $j+1 < n$, suppose $\theta(v_{j-2}) = v$. Moreover, since $j+1 < n$, we can also consider v_{j+2}. But then, independently of whether v_{j+2} is v or $\theta(v)$ we obtain $\rho_\theta(u) = \rho_\theta(v)$. Repeating the whole process leaves only the case $v_{j+1} = v_{j+2} = \ldots = v_n = v$ and $\theta(v_{j-2}) = \theta(v_{j-3}) = \ldots = \theta(v_1) = \theta(v)$ unsolved. That is, when we assume $v_{j-1}v_j = vv$, all we have to consider is the case when $v_1 v_2 \ldots v_n = v^n$. On the other hand, if we start with the assumption that $v_{j-1}v_j = \theta(v)^2$, then the only case remaining to be proved is when $v_1 = v$ and $v_2 = \cdots = v_n = \theta(v)$; in all the other cases, using similar techniques as before, we obtain that $\rho_\theta(u) = \rho_\theta(v)$. However, also in this case, independently of whether v_{n+1} is v or $\theta(v)$, we can conclude that $\rho_\theta(u) = \rho_\theta(v)$. Moreover, using similar arguments as above, if n is odd, then the only case which remains to be solved is $u\theta(u) = v^n v'$. Therefore, independently of the parity of n, the only case we have to consider is when $u\theta(u) = v^n v'$.

Let us look first at the case when n is even. If $v = xy$, as illustrated in Figure 3.14, then $u\theta(u) = v^n v' = v^{n/2} x \theta(v)^{n/2}$ with $|v'| = |x|$. But, this actually means that $v' = x$ since $\theta(v) = yx$. Moreover, x can be written as $x = \theta(z)z$ for some $z \in \Sigma^+$. So, the prefix of $\theta(u)$ of length $|v|$ is $zy\theta(z)$. Let $v^n v_{n+1} v'' = \mathrm{pref}_{2|u|+|v|-1}(\beta(v, \theta(v)))$ with $v'' \in \mathrm{Pref}(v_{n+2})$, and $v_{n+1} = \theta(z)zw$ for some $w \in \Sigma^+$.

Firstly, we consider the case $v_{n+1} = \theta(v)$. Since $\theta(z) \in \mathrm{Pref}(v_{n+1})$, $\theta(z)$ is a prefix of both v and $\theta(v)$. Note that $|v''| = 2|z| - 1$ and hence $\theta(z) \in \mathrm{Pref}(v'')$. If $|y| \geq |z|$, then $\theta(z) \in \mathrm{Pref}(y)$, i.e., $z \in \mathrm{Suff}(y)$ because $v_{n+1} = y\theta(z)z$ and $\theta(z) \in \mathrm{Pref}(v_{n+1})$. In Figure 3.14, y and v'' overlap with the overlapped part of length $|z|$ so $z = \theta(z)$. Then from the equation $v_{n+1}\theta(z) = \theta(z)zzy = y\theta(z)z\theta(z)$ we derive $z^3 y = yz^3$. This means that $\rho(y) = \rho(z)$, and thus $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, i.e., $|y| < |z|$, we have $zy = w\theta(z)$. Then, $z = wt$ and $\theta(z) = ty$ for some $t \in \Sigma^+$, which implies that $w = y = \theta(y)$. Hence $\theta(v) = y\theta(z)z = \theta(z)zy$, which implies, due to Theorem 3.18 that $\rho_\theta(y) = \rho_\theta(\theta(z))$, and hence $\rho_\theta(u) = \rho_\theta(v)$.

Next we consider the case when $v_{n+1} = v$. If $v_{n+2} = v$, then Theorem 3.8 immediately implies that $\theta(u)$ and a conjugate of v, that is, $zy\theta(z)$ share the primitive root t. Since $\theta(u) = (zy\theta(z))^{j-1}z$, $z \in \{t\}^+$, and hence $t = \theta(t)$ and $y, \theta(z) \in \{t\}^+$.

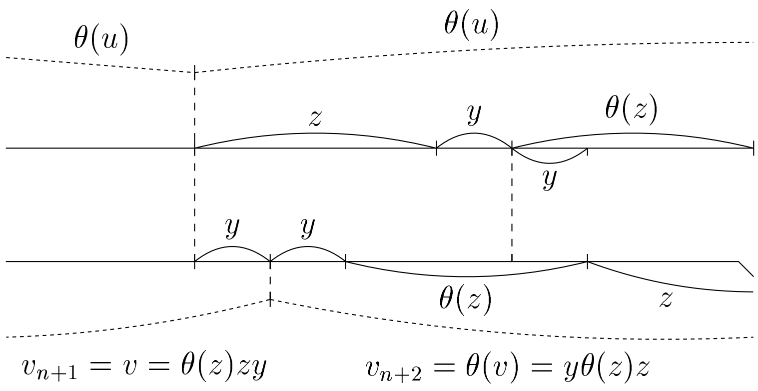

Figure 3.15: The case when n is even and $|y| < |z|$.

Thus, $u, v \in \{t\}^+$, and hence $\rho_\theta(u) = \rho_\theta(v)$. Otherwise let $v_{n+2} = \theta(v) = y\theta(z)z$. Now, we have two subcases, depending on the lengths of y and z. Firstly, if $|z| \leq |y|$, then $\mathrm{pref}_{|z|}(v'') \in \mathrm{Pref}(y)$, and hence $zy \in \mathrm{Pref}(y^2)$. Hence $\rho(y) = \rho(z)$, implying that $\rho_\theta(u) = \rho_\theta(v)$. Secondly, if $|y| < |z|$, then since $y \in \mathrm{Pref}(z)$ we also have $y \in \mathrm{Suff}(\theta(z))$. Thus, $\mathrm{pref}_{|z|-1}(\theta(z)) \in y\mathrm{Pref}(z)$, as illustrated in Figure 3.15. Moreover, since $zy^2 = y^2\theta(z)$, we actually have two distinct expressions over $\{y, z\}^+$, one starting with y and the other with z, having a common prefix of length at least $|y| + |z|$. Then, due to Corollary 3.5, we obtain $\rho(y) = \rho(z)$, which implies $\rho_\theta(u) = \rho_\theta(v)$.

Next we consider the case when n is odd and $v_1 = \cdots = v_n = v$, see Figure 3.16. Let $v = xy$ such that $x = \theta(x)$, $y = \theta(y)$, and $y = \theta(z)z$ for some $x, y, z \in \Sigma^+$. Then $u\theta(u) = v^{(n-1)/2}x\theta(z)zx\theta(v)^{(n-1)/2}$.

If $v_{n+1} = \theta(v)$, then x is a prefix of both v and $\theta(v)$ and thus $v'' = \mathrm{pref}_{|x|-1}(x)$. Hence we have $xzxz_s = v_{n+1}v'' = \theta(z)zxv''$ for some $z_s = \mathrm{pref}_{|z|-1}(\theta(z))$. Depending

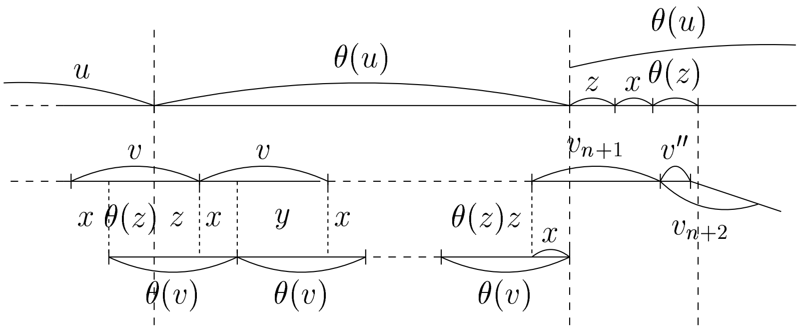

Figure 3.16: The case when n is odd and $v_1 = \cdots = v_n = v$.

on the lengths of x and z, we have the following four subcases. Let us consider the first subcase when $|x| = |z|$. Then immediately we have $x = \theta(z)$, and we are done, i.e., obviously $\rho_\theta(u) = \rho_\theta(v)$. The second subcase is when $|x| > |z|$. Then, $xzxz_s = \theta(z)zxv''$ implies that x overlaps non-trivially with xv''. Since $v'' \in \text{Pref}(x)$ and x is θ-palindrome, we can write $x = x_1x_2 = x_2x_1$ for some θ-palindromes x_1, x_2, where, moreover $x_2 = \theta(z)$. This implies that $x_1, x_2, x \in \{t\}^+$ for some $t \in \Sigma^+$, and hence $\theta(z), z, x \in \{t, \theta(t)\}^+$. Since $u, v \in \{\theta(z), z, x\}^+$, we have $\rho_\theta(u) = \rho_\theta(v)$. The third subcase is when $|x| < |z| \leq 2|x|$. Let $\theta(z) = xz_p$ for some $z_p \in \text{Pref}(z)$, which implies $z_p = \theta(z_p)$. Thus, $z = z_px$. Since $xz \in \text{Pref}(\theta(z)z)$, i.e., $xz_px \in \text{Pref}(xz_pz_px)$ and $|z_p| \leq |x|$, we have $z_p \in \text{Pref}(x)$. Now since $\theta(z)z \in \text{Pref}(xzx)$, we have $\theta(z)z = xz_pxz_p$, i.e., $z = xz_p$. Therefore, $z = xz_p = z_px$, which implies $\rho(z) = \rho(x)$ and we obtain again $\rho_\theta(u) = \rho_\theta(v)$. The fourth subcase is when $2|x| < |z|$. As in the third subcase, $\theta(z) = xz'_p$ for some θ-palindrome z'_p. Since $xzx \in \text{Pref}(\theta(z)z)$ holds in this case, let $\theta(z)z = xzxz'_s$ for some $z'_s \in \text{Pref}(\theta(z))$. By substituting $z = z'_px$ into this equation, we obtain $z = x^2z'_s$. Then $z'_s = \theta(z'_s)$. Hence, $z = z'_px = x^2z'_s$, which implies, due to Lemma 3.30, that $\rho(x) = \rho(z)$ and hence $\rho_\theta(u) = \rho_\theta(v)$.

Finally we consider the case $v_{n+1} = v = x\theta(z)z$. Then, as illustrated in Figure 3.16, $z = \theta(z)$ and thus $v_{n+1} = xz^2$. If $v_{n+2} = v$, then as above, we can employ Theorem 3.8 to conclude that $\rho_\theta(u) = \rho_\theta(v)$. Otherwise, $v_{n+2} = \theta(v) = z^2x$. Now we have four subcases depending on the lengths of x and z. The first subcase is when $|x| \leq |z|$. Note that $z^2 \in \mathrm{Pref}(zx\theta(z))$. Hence $z = xz_s$ for some $z_s \in \mathrm{Pref}(\theta(z))$, which implies that $z_s = \theta(z_s)$. Since $z = \theta(z)$, $z = xz_s = z_sx$. This means $\rho(x) = \rho(z)$ and we obtain again $\rho_\theta(u) = \rho_\theta(v)$. The second subcase is when $|z| < |x| \leq 2|z|$. Since $|v''| = |x| - 1$ and v'' is a prefix of $v_{n+2} = z^2x$, we have that $z \in \mathrm{Pref}(v'')$. Then, $z^3 \in \mathrm{Pref}(zx\theta(z))$, and we can conclude $\rho(x) = \rho(z)$ as done in the first subcase. Thus, $\rho_\theta(u) = \rho_\theta(v)$. The third subcase is when $2|z| < |x| \leq 3|z|$. Then, we actually have $z^2 \in \mathrm{Pref}(v'')$. Thus, $z^4 \in \mathrm{Pref}(zx\theta(z))$ and again we have $\rho(x) = \rho(z)$, and hence $\rho_\theta(u) = \rho_\theta(v)$. The last subcase is when $3|z| < |x|$. Recall that $v_{n+2} = z^2x$. Since $x = \theta(x)$, we can rewrite this as $v_{n+2} = z^2\theta(x)$. As $|v''| = |x| - 1$, this means that $v'' = z^2x_1$ for some $x_1 \in \mathrm{Pref}(\theta(x))$ satisfying $|x_1| = |x| - 2|z| - 1$, which is positive. Since $zx \in \mathrm{Pref}(z^2v'')$, there exists $x_2 \in \mathrm{Pref}(x_1)$ such that $zx = z^4x_2$, i.e., $x_2 \in \mathrm{Suff}(x)$. However, since $x_2 \in \mathrm{Pref}(\theta(x))$, we obtain $x_2 = \theta(x_2)$. Thus, $x = z^3x_2 = x_2z^3$, which implies, due to Lemma 3.29, that $\rho(x) = \rho(z)$, so we conclude again that $\rho_\theta(u) = \rho_\theta(v)$. $\square$

Example 8. Let $\theta : \{a, b\}^* \to \{a, b\}^*$ be the mirror involution, $u = a^2ba^3b$, and

$v = a^2ba$. Then, $\gcd(|u|,|v|) = 1$, u^3 and $v^2\theta(v)^2v$ have a common prefix of length $2|u| + |v| - 2$, but $\rho_\theta(u) \neq \rho_\theta(v)$.

Example 9. Let $\theta : \{a,b\}^* \to \{a,b\}^*$ be the mirror involution, $u = ba^2baba$, and $v = ba^2ba$. Then, $\gcd(|u|,|v|) = 1$, $u\theta(u)^2$ and v^4 have a common prefix of length $2|u| + |v| - 2$, but $\rho_\theta(u) \neq \rho_\theta(v)$.

Combining all results obtained in this section together, we have the extended Fine and Wilf theorem for an antimorphic involution θ:

Corollary 3.32. *Let $u, v \in \Sigma^*$ be two words with $|u| > |v|$. If there exist two θ-powers $\alpha(u,\theta(u)) \in u\{u,\theta(u)\}^*$ and $\beta(v,\theta(v)) \in v\{v,\theta(v)\}^*$ having a common prefix of length $2|u| + |v| - \gcd(|u|,|v|)$, then $\rho_\theta(u) = \rho_\theta(v)$. Furthermore, this bound is optimal.*

3.7 Conclusion

In this paper, we extended the notion of primitive word, being motivated by encoding information into DNA molecules. Then we investigated various relations on words u, v (word equations, extended Fine and Wilf theorem) which imply $\rho_\theta(u) = \rho_\theta(v)$. A future research topic is to generalize the extended Fine and Wilf theorem as being done for the original Fine and Wilf theorem (e.g., arbitrary number of periods, for partial words or bidimensional words). Another direction is to study relations on words which force some of the involved words to share their θ-primitive root (see [5]).

Acknowledgement

We greatfully acknowledge the anonymous reviewers for their constructive comments. This research was supported by Natural Sciences and Engineering Research Council of Canada Discovery Grant and Canada Research Chair Award to Lila Kari.

Bibliography

[1] J. Berstel and L. Boasson. Partial words and a theorem of Fine and Wilf. *Theoretical Computer Science*, 218(1):135–141, 1999.

[2] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 329–438. Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[3] S. Constantinescu and L. Ilie. Generalized Fine and Wilf's theorem for arbitrary number of periods. *Theoretical Computer Science*, 339(1):49–60, 2005.

[4] S. Constantinescu and L. Ilie. Fine and Wilf's theorem for Abelian periods. *Bulletin of the EATCS*, 89:167–170, June 2006.

[5] E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schützenberger result to pseudoperiodic words. In V. Diekert and D. Nowotka, editors, *Proc. DLT09*, volume 5583 of *Lecture Notes in Computer Science*, pages 183–194, Berlin, 2009. Springer-Verlag.

[6] A. de Luca and A. De Luca. Pseudopalindrome closure operators in free monoids. *Theoretical Computer Science*, 362:282–300, 2006.

[7] N. J. Fine and H. S. Wilf. Uniqueness theorem for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, February 1965.

[8] L. Kari and K. Mahalingam. Watson-Crick conjugate and commutative words. In *Proc. of DNA 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 273–283, 2008.

[9] L. Kari and K. Mahalingam. Watson-Crick palindromes in DNA computing. *Natural Computing*, 2009. DOI: 10.1007/s11047-009-9131-2.

[10] L. Kari, K. Mahalingam, and S. Seki. Twin-roots of words and their properties. *Theoretical Computer Science*, 410:2393–2400, 2009.

[11] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications.* Addison-Wesley, 1983.

[12] F. Mignosi, A. Restivo, and P. V. Silva. On Fine and Wilf's theorem for bidimensional words. *Theoretical Computer Science*, 292:245–262, 2003.

[13] H. J. Shyr and G. Thierrin. Disjunctive languages and codes. In *Proc. FCT77*, number 56 in Lecture Notes in Computer Science, pages 171–176, Berlin, Heidelberg, New York, 1977. Springer-Verlag.

[14] R. Tijdeman and L. Zamboni. Fine and Wilf words for any periods. *Indagationes Mathematicae*, 14(1):135–147, 2003.

[15] R. Tijdeman and L. Zamboni. Fine and Wilf words for any periods ii. *Theoretical Computer Science*, 410:3027–3034, 2009.

[16] S. S. Yu. *Languages and Codes.* Tsang Hai Book Company Co., Taichung, Taiwan, 2005.

Chapter 4

An extension of Lyndon-Schützenberger equation

This chapter consists of the contents of "An extension of the Lyndon Schützenberger result to pseudoperiodic words"[1], which is now under review of *Information and Computation* (as of August 13, 2010).

Its earlier version was presented at 13th International Conference on Developments in Language Theory (DLT 2009):

E. Czeizler, E. Czeizler, L. Kari, and S. Seki.
An extension of the Lyndon Schützenberger result to pseudoperiodic words.
In *DLT 2009*, volume 5583 of *Lecture Notes in Computer Science*, pages 183-194, Springer, 2009.

Summary: One of the particularities of information encoded as DNA strands is that a string u contains basically the same information as its Watson-Crick complement,

[1] A version of this chapter has been published.

denoted here as $\theta(u)$. Thus, any expression consisting of repetitions of u and $\theta(u)$ can be considered in some sense periodic. In this paper we give a generalization of Lyndon and Schützenberger's classical result about equations of the form $u^l = v^n w^m$, to cases where both sides involve repetitions of words as well as their complements. Our main results show that, for such extended equations, if $l \geq 5, n, m \geq 3$, then all three words involved can be expressed in terms of a common word t and its complement $\theta(t)$. Moreover, if $l \geq 5$, then $n = m = 3$ is an optimal bound. These results are established based on a complete characterization of all possible overlaps between two expressions that involve only some word u and its complement $\theta(u)$, which is also obtained in this paper.

An Extension of the Lyndon Schützenberger Result to Pseudoperiodic Words

Elena Czeizler[1], Eugen Czeizler[1], Lila Kari[2], and Shinnosuke Seki[2]

[1] Department of IT, Åbo Akademi University, Turku 20520, Finland.

[2] Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

4.1 Introduction

Periodicity and primitiveness of words are fundamental properties in combinatorics on words and formal language theory. Their wide-ranging applications include pattern-matching algorithms (see, e.g., [3, 4]) and data-compression algorithms (see, e.g., [20]). Sometimes motivated by their applications, these classical notions have been modified or generalized in various ways. A representative example is the "weak periodicity" of [5] whereby a word is called *weakly periodic* if it consists of repetitions of words with the same Parikh vector. This type of period was also called *abelian period* in [2]. Czeizler, Kari, and Seki have proposed the notion of *pseudo-primitiveness* (and pseudoperiodicity) of words in [7], motivated by the properties of information encoded as DNA strands.

DNA stores genetic information primarily in its single-stranded form as an oriented chain made up of four kinds of nucleotides: adenine ($\mathtt{A}$), cytosine ($\mathtt{C}$), guanine ($\mathtt{G}$), and thymine ($\mathtt{T}$). Thus, a single-stranded DNA can be viewed as a word over the four-letter alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. Due to the Watson-Crick complementarity property of DNA strands, whereby A is complementary to T, and C is complementary to G, single-stranded DNA molecules interact with each other. Indeed, two Watson-Crick complementary DNA single strands with opposite orientation will bind to each other by weak hydrogen bonds between their individual bases and form the well-known DNA double helix structure. In the process of DNA replication, a DNA double strand is separated into its two constituent single strands, each of which serves as a template for the enzyme DNA polymerase. Starting from one end of a DNA single strand, DNA polymerase has the ability to build up, one nucleotide at a time, a new DNA strand that is perfectly complementary to the template, resulting in two copies of the DNA double strand. Thus, two DNA strands Watson-Crick complementary to each other can be considered "equivalent" in terms of the information they encode.

The fact that one can consider a DNA strand and its Watson-Crick complement "equivalent" led to natural as well as theoretically interesting extensions of various notions in combinatorics of words and formal language theory such as pseudo-palindrome [8], pseudo-commutativity [13], as well as hairpin-free and bond-free languages (e.g., [12, 15, 19]). Watson-Crick complementarity has been modeled mathematically by an antimorphic involution θ, i.e., a function that is an antimorphism, $\theta(uv) = \theta(v)\theta(u)$, $\forall u, v \in \Sigma^*$, and an involution, $\theta(\theta(x)) = x$, $\forall x \in \Sigma^*$. The aforementioned new concepts and notions are based on extending the notion of identity between words to that of "equivalence" between a word u and $\theta(u)$, in the sense that an occurence of $\theta(u)$ will be treated as another occurrence of u, albeit disguised by the application of θ.

In [7], a word w is called *θ-primitive* if we cannot find any word x that is strictly shorter than w such that w can be written as a combination of x and $\theta(x)$. For instance, if θ is the Watson-Crick complementarity then ATCG is θ-primitive, whereas TCGA is not because TCGA $=$ TCθ(TC). The periodicity theorem of Fine and Wilf – one of the fundamental results on periodicity of words, see, e.g., [1] and [17] – was also extended as follows "For given words u and v, how long does a common prefix of a word in $\{u, \theta(u)\}^+$ and a word in $\{v, \theta(v)\}^+$ have to be, in order to imply that $u, v \in \{t, \theta(t)\}^+$ for some word t?".

In this paper, we continue the theoretical study of θ-primitive words by extending another central periodicity result, due to Lyndon and Schützenberger [18]. The original result states that, if the concatenation of two periodic words v^n and w^m can be expressed in terms of a third period u, i.e., $u^\ell = v^n w^m$, for some $\ell, m, n \geq 2$, then all three words u, v, and w can be expressed in terms of a common word t, i.e., $u, v, w \in \{t\}^+$. (See also [10] and Chapter 5 from [17] for some of its shorter proofs and [11, 16] for some other generalizations.) Replacing identity of words by the weaker notion of "equivalence" between a word u and $\theta(u)$, for a given antimorphic involution θ, we define an extended Lyndon and Schützenberger equation as

$$u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m,$$

where $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$ with $\ell, n, m \geq 2$. For this extended Lyndon and Schützenberger equation we ask the following question: "What conditions on ℓ, n, m, imply that all three words u, v, w can be written as a combination of a word and its image under θ, i.e., $u, v, w \in \{t, \theta(t)\}^+$ for some word t?". In this paper we give a partial positive answer and partial negative answer to this question.

The positive answer states that whenever $\ell \geq 5$, $n, m \geq 3$, the extended Lyndon-Schützenberger equation implies $u, v, w \in \{t, \theta(t)\}^+$ for some word t (Theorem 4.22). The negative answer states that once either n or m becomes 2, we can construct u, v, w which satisfy the extended equation, but such a word t does not exist (Examples 10 and 11). Therefore, for any $\ell \geq 5$, $n = m = 3$ is an optimal bound. In the case when $\ell = 3$ or $\ell = 4$, the problem of finding optimal bounds remains open, though the negative result holds even in these cases. Our proofs are not generalizations of the methods used in the classical case, since one of the main properties used therein, i.e., the fact that the conjugate of a primitive word is still primitive, does not hold for θ-primitiveness any more.

Prior to the proof of the positive result, we characterize all non-trivial overlaps between two expressions $\alpha(v, \theta(v)), \beta(v, \theta(v)) \in \{v, \theta(v)\}^+$ for a θ-primitive word v. Formally speaking, we show that the equality $\alpha(v, \theta(v)) \cdot x = y \cdot \beta(v, \theta(v))$ with x and y shorter than v is possible, and we provide all possible representations of the involved words v, x, y (Theorem 4.10). Note that this result is in contrast to the classical case (where the two expressions involve only a word v, but not its image under θ).

The paper is organized as follows. In Section 4.2, we fix our terminology and recall some known results. In Section 4.3, we provide the characterization of all possible overlaps of the form $\alpha(v, \theta(v)) \cdot x = y \cdot \beta(v, \theta(v))$ with $\alpha(v, \theta(v)), \beta(v, \theta(v)) \in \{v, \theta(v)\}^+$ and x, y shorter than v. Finally, in Section 4.4 we provide our extension of Lyndon and Schützenberger's result.

4.2 Preliminaries

Here we introduce notions and notation used in the following sections. For details of each, readers are referred to [1, 17].

Let Σ be a finite alphabet. We denote by Σ^* the set of all finite words over Σ, by λ the empty word, and by Σ^+ the set of all nonempty finite words. The catenation of two words $u, v \in \Sigma^*$ is denoted by either uv or $u \cdot v$. The *length* of a word $w \in \Sigma^*$, denoted by $|w|$, is the number of letters occurring in it. We say that u is a *factor* (a *prefix*, a *suffix*) of v if $v = t_1ut_2$ (resp. $v = ut_2$, $v = t_1u$) for some $t_1, t_2 \in \Sigma^*$. We denote by $\mathrm{Pref}(v)$ (resp. $\mathrm{Suff}(v)$) the set of all prefixes (resp. suffixes) of the word v. We say that two words u and v overlap if $ux = yv$ for some $x, y \in \Sigma^*$ with $|x| < |v|$. An integer $p \geq 1$ is a *period* of a word $w = a_1 \ldots a_n$, with $a_i \in \Sigma$ for all $1 \leq i \leq n$, if $a_i = a_{i+p}$ for all $1 \leq i \leq n - p$.

A word $w \in \Sigma^+$ is called *primitive* if it cannot be written as a power of another word; that is, if $w = u^n$ then $n = 1$ and $w = u$. For a word $w \in \Sigma^+$, the shortest $u \in \Sigma^+$ such that $w = u^n$ for some $n \geq 1$ is called the *primitive root* of the word w and is denoted by $\rho(w)$. It is well-known that two words u, v commute, i.e., $uv = vu$ if and only if u, v have the same primitive root. This is rephrased as the following proposition.

Proposition 4.1. *Let $u \in \Sigma^+$ be a primitive word. If $u^2 = xuy$, then either $x = \lambda$ or $y = \lambda$.*

A mapping $\theta : \Sigma^* \to \Sigma^*$ is called an *antimorphism* if for any words $u, v \in \Sigma^*$, $\theta(uv) = \theta(v)\theta(u)$. A mapping $\theta : \Sigma^* \to \Sigma^*$ is called an *involution* if θ^2 is the identity. As mentioned in the introduction, an antimorphic involution is a mathematical formalization of the Watson-Crick complementarity. Throughout this paper we will assume that θ is an antimorphic involution on a given alphabet Σ. A word $w \in \Sigma^*$ is called a *θ-palindrome*, or a *pseudo-palindrome* if θ is not specified, if $w = \theta(w)$ (see [14] and [8]).

The notions of periodic and primitive words were extended in [7] in the following way. A word $w \in \Sigma^+$ is θ-periodic if $w = w_1 \ldots w_k$ for some $k \geq 2$ and words $t, w_1, \ldots, w_k \in \Sigma^+$ such that $w_i \in \{t, \theta(t)\}$ for all $1 \leq i \leq k$. Following [8], in less precise terms, a word which is θ-periodic with respect to a given but unspecified involutory morphism θ will be also called *pseudoperiodic*. The word t in the definition of a θ-periodic word w is called a θ-period of w. We call a word $w \in \Sigma^+$ *θ-primitive* if it is not θ-periodic. The set of θ-primitive words is strictly included in the set of primitive ones, see [7]; for instance, if we take $a \neq b$ and $\theta(a) = b$, $\theta(b) = a$, then the word ab is primitive, but not θ-primitive. We define *the θ-primitive root of w*, denoted by $\rho_\theta(w)$, as the shortest word t such that $w = w_1 \dots w_k$ for some $k \geq 1$, $w_i \in \{t, \theta(t)\}$ for all $1 \leq i \leq k$, and $w_1 = t$. Note that if w is θ-primitive, then $\rho_\theta(w) = w$.

The Fine and Wilf theorem, originally formulated for sequences of real numbers in [9], illustrates another fundamental periodicity property in its form for words [1, 17]. It states that for two words $u, v \in \Sigma^*$, if a power of u and a power of v have a common prefix of length at least $|u| + |v| - \gcd(|u|, |v|)$, then u and v are powers of a common word, where $\gcd(,)$ denotes the *greatest common divisor of two arguments*. Moreover, the bound $|u| + |v| - \gcd(|u|, |v|)$ is optimal.

This theorem was extended in [7] for the case when instead of powers of two words u and v, we look at expressions over $\{u, \theta(u)\}$ and $\{v, \theta(v)\}$, respectively. The extended theorem consists of the following two variants.

Theorem 4.2 ([7]). *Let $u, v \in \Sigma^+$ be two distinct words with $|u| > |v|$. If there exist two expressions $\alpha(u, \theta(u)) \in u\{u, \theta(u)\}^*$ and $\beta(v, \theta(v)) \in v\{v, \theta(v)\}^*$ having a common prefix of length at least $2|u| + |v| - gcd(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$. Moreover, the bound $2|u| + |v| - gcd(|u|, |v|)$ is optimal.*

Theorem 4.3 ([7]). *Let $u, v \in \Sigma^+$, $\alpha(u,\theta(u)) \in u\{u,\theta(u)\}^*$, and $\beta(v,\theta(v)) \in v\{v,\theta(v)\}^*$ such that $\alpha(u,\theta(u)) = \beta(v,\theta(v))$. Then $\rho_\theta(u) = \rho_\theta(v)$.*

The next two results, also from [7], will be very useful in our considerations.

Lemma 4.4 ([7]). *For $u, v \in \Sigma^*$, if $uv = \theta(uv)$ and $vu = \theta(vu)$, then there exists a word $t \in \Sigma^+$ such that $u, v \in \{t, \theta(t)\}^*$.*

Lemma 4.5 ([7]). *Let $v \in \Sigma^+$ be a θ-primitive word. Then, $\theta(v)vx = yv\theta(v)$ for some words $x, y \in \Sigma^*$ with $|x|, |y| < |v|$, if and only if $v = \theta(v)$ and $x = y = \lambda$. Similarly, $v\theta(v)v = xv^2y$ for some $x, y \in \Sigma^*$ if and only if $v = \theta(v)$ and either $x = \lambda$ or $y = \lambda$.*

The following result will prove very useful in our future considerations.

Lemma 4.6. *Let $u \in \Sigma^+$ such that $u = xz = zy$ for some $x, y, z \in \Sigma^+$ with $x = \theta(x)$ and $y = \theta(y)$. Then $x, y, z, u \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$.*

Proof. The equation $u = xz = zy$ implies that $x = pq$, $y = qp$, and $z = (pq)^j p$ for some $p, q \in \Sigma^*$ and $j \geq 0$. Since $x = \theta(x)$ and $y = \theta(y)$, we have $pq = \theta(pq)$ and $qp = \theta(qp)$. Then, Lemma 4.4 implies that there exists a word $t \in \Sigma^+$ such that $p, q \in \{t, \theta(t)\}^*$. $\square$

When considering word equations that involve the antimorphic involution like those in the previous lemmas, one often encounters the θ-commutativity of words. A word u is said to *θ-commute* with a word v if $uv = \theta(v)u$ [13]. This is a special case of the *conjugacy* of words $ux = yu$. The solution to this equation is given as: $u = (pq)^i p$, $x = (qp)^j$, and $y = (pq)^j$ for some $i \geq 0$, $j \geq 1$ and $p, q \in \Sigma^*$ such that pq is primitive. Note that if we give up the primitivity of pq, then we can assume $j = 1$. Thus, we can characterize the solution to the θ-commutativity of words as follows.

Proposition 4.7 ([13]). *For words $u, v \in \Sigma^+$ and an antimorphic involution θ, if $uv = \theta(v)u$ holds, then $u = (rt)^i r$ and $v = (tr)^j$ for some $i \geq 0$, $j \geq 1$, and θ-palindromes $r, t \in \Sigma^*$ such that rt is primitive.*

4.3 Overlaps between θ-Primitive Words

It is well known that a primitive word v cannot occur nontrivially inside v^2, see Proposition 4.1. Thus, two expressions v^i and v^j, with $i, j \geq 1$, cannot overlap nontrivially on a sequence longer than $|v|$. A natural question is whether we can have some nontrivial overlaps between two expressions $\alpha(v, \theta(v)), \beta(v, \theta(v)) \in \{v, \theta(v)\}^+$, when $v \in \Sigma^+$ is a θ-primitive word. In this section we completely characterize all such nontrivial overlaps, and, moreover, in each case we also give the set of all solutions of the corresponding equation.

We begin our analysis by giving two intermediate results.

Theorem 4.8. *Let $v \in \Sigma^+$ be a θ-primitive word and $\alpha(v, \theta(v)), \beta(v, \theta(v)) \in \{v, \theta(v)\}^+$ such that $\alpha(v, \theta(v)) \cdot x = y \cdot \beta(v, \theta(v))$, with $x, y \in \Sigma^+$, $|x|, |y| < |v|$. Then,*

v^2 and $\theta(v)^2$ cannot occur simultaneously neither in $\alpha(v, \theta(v))$ nor in $\beta(v, \theta(v))$.

Proof. Suppose that both v^2 and $\theta(v)^2$ occur in $\alpha(v, \theta(v))$; the case when they both occur in $\beta(v, \theta(v))$ is symmetric. Moreover, since θ is an involution, we can suppose without loss of generality that v^2 occurs before $\theta(v)^2$, thus implying that $v^2\theta(v)$ is a factor in $\alpha(v, \theta(v))$. Since v (resp. $\theta(v)$) is primitive, the border between any two consecutive v's (resp. $\theta(v)$'s) falls inside a $\theta(v)$ (resp. v), see Figure 4.1.

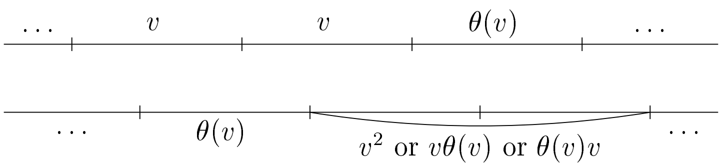

Figure 4.1: The case when $v^2\theta(v)$ is a factor in $\alpha(v, \theta(v))$

Thus, $v^2\theta(v)$ overlaps either with $\theta(v)v^2$ or with $\theta(v)v\theta(v)$ or with $\theta(v)^2v$. In all three cases the nontrivial overlap between $v\theta(v)$ and $\theta(v)v$ contradicts the θ-primitivity of v, see Lemma 4.5. ☐

Theorem 4.9. *For a θ-primitive word $v \in \Sigma^+$, let $\alpha(v, \theta(v)), \beta(v, \theta(v)) \in \{v, \theta(v)\}^+$ such that $\alpha(v, \theta(v)) \cdot x = y \cdot \beta(v, \theta(v))$ for some $x, y \in \Sigma^+$ with $|x|, |y| < |v|$. Then, for any $i \geq 1$, neither $v\theta(v)^iv$ nor $\theta(v)v^i\theta(v)$ can occur either in $\alpha(v, \theta(v))$ or in $\beta(v, \theta(v))$.*

Proof. Suppose that $v\theta(v)^iv$ occurs in $\alpha(v, \theta(v))$ for some $i \geq 1$. We assumed that $x, y \in \Sigma^+$ and $|x|, |y| < |v|$ so that the factor $v\theta(v)^iv$ contains as a proper factor $\gamma(v, \theta(v)) \in \{v, \theta(v)\}^{i+1}$, i.e., there exist some $p, q \in \Sigma^+$ such that $v\theta(v)^iv =$

Table 4.1: Characterization of possible proper overlaps of the form $\alpha(v,\theta(v))\cdot x = y\cdot\beta(v,\theta(v))$. For the second and third equations, $p,q\in\Sigma^+$. For the last three equations, $i\geq 0$, $j\geq 1$, $r,t\in\Sigma^+$ such that $r=\theta(r)$, $t=\theta(t)$, and rt is primitive. Note that the 4th and 5th equations are the same up to the antimorphic involution θ

Equation	Solution
$v^k x = y\theta(v)^k, k\geq 1$	$v=yp, x=\theta(y), p=\theta(p)$, and whenever $k\geq 2$, $y=\theta(y)$
$vx=yv$	$v=(pq)^{i+1}p$, $x=qp$, $y=pq$, with $i\geq 0$
$v\theta(v)x = yv\theta(v)$,	$v=(pq)^{i+1}p$, $x=\theta(pq)$, $y=pq$, with $i\geq 0$, $qp=\theta(qp)$
$v^{k+1}x = y\theta(v)^k v, k\geq 1$	$v=r(tr)^{i+j}r(tr)^i$, $x=(tr)^j r(tr)^i$, $y=r(tr)^{i+j}$
$v\theta(v)^k x = yv^{k+1}$, $k\geq 1$	$v=(rt)^i r(rt)^{j+i}r$, $y=(rt)^i r(rt)^j$, $x=(rt)^{j+i}r$
$v\theta(v)^i x = yv^i\theta(v)$, $i\geq 2$	$v=(rt)^n r(rt)^{m+n}r$, $y=(rt)^n r(rt)^m$, $x=(tr)^m r(tr)^n$

$p\gamma(v,\theta(v))q$. Due to Lemma 4.5 and $\theta(v)$ being primitive, $\gamma(v,\theta(v))=v^{i+1}$. Now we have $v\theta(v)^i v = pv^{i+1}q$ and hence $v\theta(v)v = pv^2q$. However, this contradicts Lemma 4.5. The other cases can be proved similarly. $\square$

As an immediate consequence of the previous two theorems, for a given θ-primitive word v, if $\alpha(v,\theta(v))\cdot x = y\cdot\beta(v,\theta(v))$ with $x,y\in\Sigma^+$, $|x|,|y|<|v|$, then $\alpha(v,\theta(v))$ and $\beta(v,\theta(v))$ can be only of the following types v^k, $v^k\theta(v)$, $v\theta(v)^k$, $\theta(v)^k$, $\theta(v)^k v$, or $\theta(v)v^k$ for some $k\geq 1$. The next result refines this characterization further.

Theorem 4.10. *Let $v\in\Sigma^+$ be a θ-primitive word. Then, the only possible proper overlaps of the form $\alpha(v,\theta(v))\cdot x = y\cdot\beta(v,\theta(v))$ with $\alpha(v,\theta(v)),\beta(v,\theta(v))\in\{v,\theta(v)\}^+$, $x,y\in\Sigma^+$ and $|x|,|y|<|v|$ are given in Table 4.1 (modulo a substi-*

tution of v by $\theta(v)$) together with the characterization of their sets of solutions.

Proof. Since θ is an involution, we can assume without loss of generality that $\alpha(v, \theta(v))$ starts with v. Then, due to the previous observation we know that $\alpha(v, \theta(v)) \in \{v^k, v^k\theta(v), v\theta(v)^k \mid k \geq 1\}$.

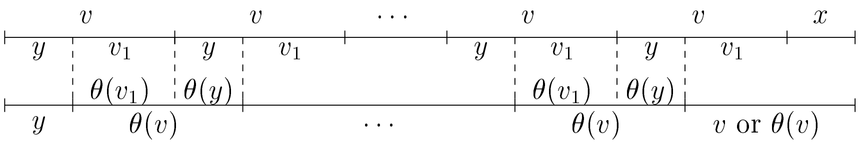

Figure 4.2: The case when $\alpha(v, \theta(v)) = v^i$

Case 1: First we consider the case when $\alpha(v, \theta(v)) = v^k$ for some $k \geq 1$. Since v is θ-primitive, $v^k x = y\beta(v, \theta(v))$, and $|y|, |x| < |v|$, the border between any two consecutive v's falls inside a $\theta(v)$, see Figure 4.2; otherwise v would occur inside v^2 which would contradict its primitivity. Thus, $\beta(v, \theta(v)) \in \{\theta(v)^k, \theta(v)^{k-1}v\}$. Then, we can write $v = yv_1$, that is, $\theta(v) = \theta(v_1)\theta(y)$.

Suppose first that $\beta(v, \theta(v)) = \theta(v)^k$. Then, we immediately obtain $v_1 = \theta(v_1)$ and in addition, if $i \geq 2$, then $y = \theta(y)$. Moreover, if we look at the end of the two sides of the equation $v^k x = y\theta(v)^k$, we also obtain that $x = \theta(y)$. Thus, a proper overlap of the form $v^k x = y\theta(v)^k$ with v being θ-primitive is possible, and, moreover, the set of all solutions of this equation is characterized by the following formulas: $v = yv_1$ and $x = \theta(y)$, where $v_1 = \theta(v_1)$ and $y = \theta(y)$ whenever $k \geq 2$.

Suppose now that $\beta(v, \theta(v)) = \theta(v)^{k-1}v$. If we look at the end of the two sides of the equation, then we obtain $v = v_1 x$. Thus, $v = yv_1 = v_1 x$. This conjugacy implies that there exist some $p, q \in \Sigma^*$ and $n \geq 0$ such that $y = pq$, $x = qp$, $v_1 = (pq)^n p$, and $v = (pq)^{n+1}p$. If q is empty, then $v = p^{n+2}$ which contradicts the primitivity of v. If p is empty, then $v = q^{n+1}$ which either contradicts the primitivity of v or, when $n = 0$, implies that $v = y$ contradicting our assumption that $|y| < |v|$. Thus, p, q have to be nonempty. The set of all solutions of this equation with $k = 1$ is characterized by $v = (pq)^{n+1}p$, $x = qp$, $y = pq$, and $n \geq 0$. Now, if $k \geq 2$, then $v_1 = \theta(v_1)$ and $y = \theta(y)$, i.e., $p = \theta(p)$ and $pq = \theta(pq) = \theta(q)p$. If $n \geq 1$, then also $q = \theta(q)$, which contradicts the primitivity of v. Thus, if $k \geq 2$, then n has to be 0. Due to Proposition 4.7, the θ-commutativity $pq = \theta(q)p$ implies $p = r(tr)^i$ and $q = (tr)^j$ for $i \geq 0$, $j \geq 1$, and two θ-palindromes r, t such that rt is primitive. In fact, r and t must be non-empty. Recall that $v = pqp = r(tr)^{i+j}r(tr)^i$. If t were empty, then $v = r^{2i+j+2}$ and would not be even primitive, a contradiction. Even if r were empty, unless $i = 0$ and $j = 1$, we reach the same contradiction; if $i = 0$ and $j = 1$, then $y = v$, which contradicts the assumption $|y| < |v|$. To conclude, a proper overlap of the form $v^k x = y\theta(v)^{k-1}v$ with v being θ-primitive and $k \geq 2$ is also possible. The set of all solutions of this equation is characterized by the following formulas: $v = r(tr)^{i+j}r(tr)^i$, $x = (tr)^j r(tr)^i$, and $y = r(tr)^{i+j}$.

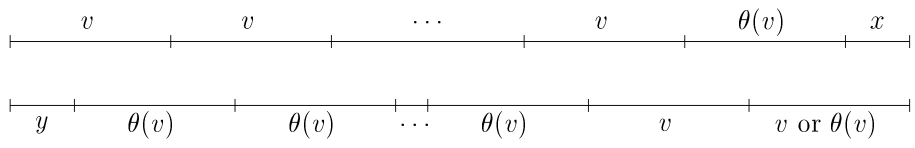

Figure 4.3: The case when $\alpha(v, \theta(v)) = v^i\theta(v)$ and $\beta(v, \theta(v))$ begins with $\theta(v)^{i-1}v$

Case 2: Suppose now that $\alpha(v, \theta(v)) = v^k\theta(v)$ for some $k \geq 1$. If $k \geq 2$, then $\beta(v, \theta(v))$ has to start with $\theta(v)^{k-1}$ because otherwise it would contradict the primitivity of v. If this $\theta(v)^{k-1}$ is followed by v, see Figure 4.3, then $v\theta(v)$ overlaps with $\theta(v)v$ with the overlap properly longer than v. Then Lemma 4.5 leads to a contradiction. Hence, $\beta(v, \theta(v))$ starts with $\theta(v)^k$, see Figure 4.4. Then, however, $\beta(v, \theta(v))$ can end neither with v due to Lemma 4.5, nor with $\theta(v)$ due to Proposition 4.1.

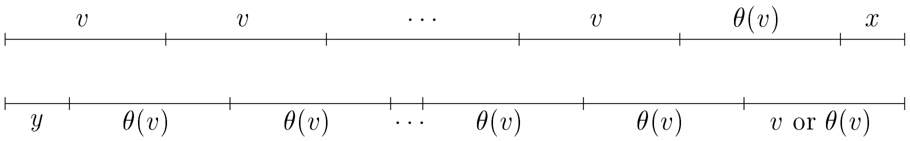

Figure 4.4: The case when $\alpha(v, \theta(v)) = v^i\theta(v)$ and $\beta(v, \theta(v))$ begins with $\theta(v)^i$

Thus k has to be 1. Proposition 4.1 and Lemma 4.5 imply that $\beta(v, \theta(v))$ starts with v. Firstly we consider the case when $\beta(v, \theta(v)) = v^2$, that is, we have $v\theta(v)x = yv^2$ (see Figure 4.5 left). Note that for any $x, y \in \Sigma^+$ with $|x|, |y| < |v|$, $v\theta(v)x = yv^2$ holds if and only if $v\theta(v)^kx = yv^{k+1}$ holds for any $k \geq 1$. Furthermore, the latter equation is the same as the equation $v^{k+1}x' = y'\theta(v)^kv$, which was considered in Case 1, up to the antimorphic involution θ. Using the result obtained in Case 1, we have that the set of all solutions of $v\theta(v)^kx = yv^{k+1}$ is characterized by the formulae $v = (rt)^ir(rt)^{j+i}r$, $x = (rt)^{j+i}r$, and $y = (rt)^ir(rt)^j$.

The remaining case for $\alpha(v, \theta(v)) = v\theta(v)$ is when $\beta(v, \theta(v)) = v\theta(v)$, see Figure 4.5 right. Then, we can write $v = yv_1 = v_1v_2$ and we obtain immediately $x = \theta(y)$ and $v_2 = \theta(v_2)$. Thus, a proper overlap of the form $v\theta(v)x = yv\theta(v)$, with

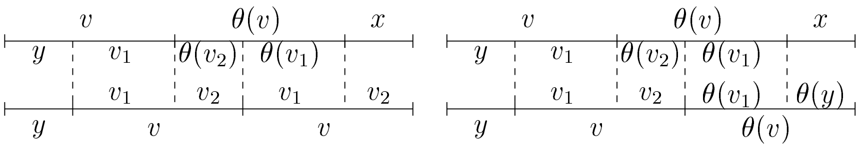

Figure 4.5: The equations: $v\theta(v)x = yv^2$ and $v\theta(v)x = yv\theta(v)$

v being θ-primitive, is possible. Furthermore, the set of all solutions of this equation is characterized by the following formulas: $v = (pq)^{i+1}p$, $y = pq$, $x = \theta(pq)$ for some $i \geq 0$ and $p, q \in \Sigma^*$ such that $qp = \theta(qp)$. We can easily check that p, q have to be non-empty as done previously.

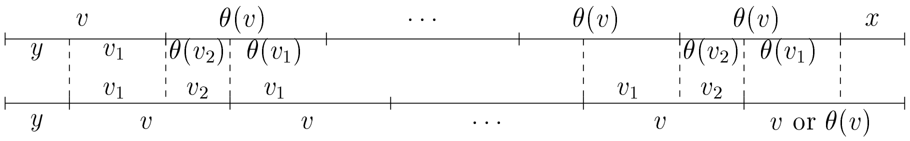

Figure 4.6: The case when $\alpha(v, \theta(v)) = v\theta(v)^i$ and $\beta(v, \theta(v))$ starts with v

Case 3: Finally we consider the case when $\alpha(v, \theta(v)) = v\theta(v)^k$ for some $k \geq 2$; the case when $k = 1$ was already considered. Since $\theta(v)$ is primitive, the border between any two $\theta(v)$'s falls inside v. If $\beta(v, \theta(v))$ starts with $\theta(v)$, then this $\theta(v)$ could not be followed by either v due to Lemma 4.5 or $\theta(v)$ due to Proposition 4.1. Therefore $\beta(v, \theta(v))$ has to begin with v, and moreover $\beta(v, \theta(v)) \in v^k\{v, \theta(v)\}$, see Figure 4.6. Actually it suffices to consider the case $\beta(v, \theta(v)) = v^k\theta(v)$ because the other was already addressed in Case 2. In this case, $x = \theta(y)$. Since $v\theta(v) \in \mathrm{Pref}(yv^2)$ and $|y| < |v|$, we have $v = yv_1 = v_1v_2$ for some $v_1, v_2 \in \Sigma^+$ with $v_1 = \theta(v_1)$ and $v_2 = \theta(v_2)$. As seen in Case 1, this equation implies $y = \theta(x) = (rt)^i r(rt)^j$, $v_1 = (rt)^i r$, and $v_2 = (rt)^{i+j}r$ for some $i \geq 0$, $j \geq 1$, and two nonempty θ-palindromes r, t such that rt is primitive. $\square$

4.4 An Extension of Lyndon and Schützenberger's Result

As an application of the obtained characterization of non-trivial overlaps, now we consider the extended Lyndon-Schützenberger equation. Let us recall first the original result by Lyndon and Schützenberger [18].

Theorem 4.11. *If words u, v, w satisfy the relation $u^\ell = v^n w^m$ for some positive integers $\ell, n, m \geq 2$, then they are all powers of a common word, i.e., there exists a word t such that $u, v, w \in \{t\}^*$.*

Let us extend the equation as follows: for $u, v, w \in \Sigma^+$ and $\ell, n, m \geq 2$,

$$u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m, \tag{4.1}$$

where $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. We call Eq. (4.1) the *extended Lyndon-Schützenberger equation* (abbreviated as exLS equation).

In light of Theorem 4.11, we ask the question of under what conditions on ℓ, n, m, the exLS equation implies that $u, v, w \in \{t, \theta(t)\}^+$ for some word $t \in \Sigma^+$. If such t

exists, we say that the triple (ℓ, n, m) *imposes θ-periodicity on u, v, w*, (or shortly, *imposes θ-periodicity*). Furthermore, we say that the triple (ℓ, n, m) *imposes θ-periodicity* if it imposes θ-periodicity on all u, v, w. Note that, if (ℓ, n, m) imposes θ-periodicity, then so does (ℓ, m, n), and vice versa. Note also that the fact that a certain triple (ℓ, n, m) imposes θ-periodicity does not imply that (ℓ', n', m') imposes θ-periodicity for $\ell' > \ell$ or $n' > n$ or $m' > m$.

The results of this section are summarized in Table 4.2. Overall, combining all the results from this section we obtain that $\ell \geq 5$, $n \geq 3$, $m \geq 3$ imposes θ-periodicity on u, v, and w (Theorem 4.22). In contrast, for $\ell \geq 3$, once either $n = 2$ or $m = 2$, (ℓ, n, m) does not always impose θ-periodicity, see Examples 10 and 11. Therefore, when $\ell \geq 5$, $(\ell, 3, 3)$ is the optimal bound. In the case when $\ell = 2$, $\ell = 3$, or $\ell = 4$, the problem of finding optimal bounds is still open.

Table 4.2: Result summary for the extended Lyndon-Schützenberger equation.

ℓ	n	m	θ-periodicity	
≥ 6	≥ 3	≥ 3	YES	Theorem 4.12
5	≥ 5	≥ 5	YES	Theorem 4.13
5	4	≥ 4	YES	Theorem 4.20
5	3	≥ 3	YES	Theorem 4.21
≥ 3	2	≥ 1	NO	Examples 10 and 11

Example 10. Let $\Sigma = \{a, b\}$ and $\theta : \Sigma^* \to \Sigma^*$ be the mirror image defined as $\theta(a) = a$, $\theta(b) = b$, and $\theta(w_1 \ldots w_n) = w_n \ldots w_1$, where $w_i \in \{a, b\}$ for all $1 \leq i \leq n$. Take now $u = a^k b^2 a^{2k}$, $v = \theta(u)^l a^{2k} b^2 = (a^{2k} b^2 a^k)^l a^{2k} b^2$, and $w = a^2$, for some $k, l \geq 1$. Then, although $\theta(u)^{l+1}u^{l+1} = v^2w^k$, there is no word $t \in \Sigma^+$ with $u, v, w \in \{t, \theta(t)\}^+$, i.e., for any $k, l \geq 1$, the triple of numerical parameters $(2l+2, 2, k)$ is not enough to impose θ-periodicity.

Example 11. Consider again $\Sigma = \{a, b\}$ and $\theta : \Sigma^* \to \Sigma^*$ be the mirror image defined in the previous example and take $u = b^2(aba)^k$, $v = u^l b = (b^2(aba)^k)^l b$, and $w = aba$ for some $k, l \geq 1$. Then, although $u^{2l+1} = v\theta(v)w^k$, there is no word $t \in \Sigma^+$ with $u, v, w \in \{t, \theta(t)\}^+$, i.e., for any $k, l \geq 1$, $(2l+1, 2, k)$ is not enough to impose θ-periodicity.

In the rest of this section, we handle the cases when (ℓ, n, m) imposes θ-periodicity. Among them, we firstly consider some cases where enough amount of repetition is available for us to apply the extended Fine and Wilf's theorem (Theorem 4.2). The next two results analyze the cases when we have triples (ℓ, n, m) with $\ell \geq 6$ and $n, m \geq 3$ and respectively $(5, n, m)$ with $n, m \geq 5$.

Theorem 4.12. *Let $u, v, w \in \Sigma^+$, $n, m \geq 3$, $\ell \geq 6$, $u_i \in \{u, \theta(u)\}$ for $1 \leq i \leq \ell$, $v_j \in \{v, \theta(v)\}$ for $1 \leq j \leq n$, and $w_k \in \{w, \theta(w)\}$ for $1 \leq k \leq m$. If $u_1 \dots u_\ell = v_1 \dots v_n\, w_1 \dots w_m$, then there exists a word $t \in \Sigma^+$ such that $u, v, w \in \{t, \theta(t)\}^+$.*

Proof. Let us suppose that $|v_1 \dots v_n| \geq |w_1 \dots w_m|$; the other case is symmetric and can be solved similarly. Then, $|v_1 \dots v_n| \geq \frac{1}{2}|u_1 \dots u_l| \geq 3|u|$, since $\ell \geq 6$. Since $n \geq 3$, this means that $u_1 \dots u_\ell$ and $v_1 \dots v_n$ share a common prefix of length larger than both $3|u|$ and $3|v|$. Thus, we can apply Theorem 4.2 to obtain that $u, v \in \{t, \theta(t)\}^+$

for some θ-primitive word $t \in \Sigma^+$. Moreover, since $u_1 \ldots u_\ell = v_1 \ldots v_n \ w_1 \ldots w_m$, this implies $w_1 \ldots w_m \in \{t, \theta(t)\}^*$. Since t is θ-primitive, Theorem 4.3 implies that also $w \in \{t, \theta(t)\}^+$. $\square$

This proof clarifies one important point: in order to prove that (ℓ, n, m) imposes θ-periodicity, it suffices to prove that two of u, v, w are in $\{t, \theta(t)\}^+$ for some t.

Theorem 4.13. *Let $u, v, w \in \Sigma^+$, $n, m \geq 5$, $u_i \in \{u, \theta(u)\}$ for $1 \leq i \leq 5$, $v_j \in \{v, \theta(v)\}$ for $1 \leq j \leq n$, and $w_k \in \{w, \theta(w)\}$ for $1 \leq k \leq m$. If $u_1u_2u_3u_4u_5 = v_1 \ldots v_n \ w_1 \ldots w_m$, then there exists a word $t \in \Sigma^+$ such that $u, v, w \in \{t, \theta(t)\}^+$.*

Proof. Since $u_1u_2u_3u_4u_5 = v_1 \ldots v_n \ w_1 \ldots w_m$ and $n, m \geq 5$, we immediately obtain that $|u| > |v|$ and $|u| > |w|$. Assume now that $n|v| \geq m|w|$; the other case is symmetric. Thus, $n|v| \geq 2|u| + \frac{1}{2}|u|$ and we take $n|v| = 2|u| + l$ for some $l \geq \frac{1}{2}|u|$.

We claim now that $l \geq |v|$. If $l \geq |u|$, then we are done since we already know that $|u| > |v|$. So, let $\frac{1}{2}|u| \leq l < |u|$. If $n \geq 6$, then $n|v| = 2|u| + l < 3|u|$ and thus $|v| < \frac{1}{2}|u| \leq l$. Thus, the only case remaining now is when $n = 5$. Then, $5|v| = 2|u| + l \geq 2|u| + \frac{1}{2}|u|$, which implies $|v| \geq \frac{1}{2}|u|$. But then we have that $4|v| \geq 2|u|$ while $5|v| = 2|u| + l$. Hence, also in this case we obtain $|v| \leq l$.

Thus, $u_1u_2u_3u_4u_5$ and $v_1 \ldots v_n$ have a common prefix of length $n|v| = 2|u| + l \geq 2|u| + |v|$. This means, due to Theorem 4.2, that there exists a θ-primitive word $t \in \Sigma^+$ such that $u, v \in \{t, \theta(t)\}^+$. As mentioned previously, now we can also say that $w \in \{t, \theta(t)\}^+$. $\square$

The triple $(5, n, m)$ also turns out to impose θ-periodicity for any $n \geq 4$ and $m \geq 7$.

Theorem 4.14. *Let $u, v, w \in \Sigma^+$, $n \geq 4$, $m \geq 7$, $u_i \in \{u, \theta(u)\}$ for $1 \leq i \leq 5$, $v_j \in \{v, \theta(v)\}$ for $1 \leq j \leq n$, and $w_k \in \{w, \theta(w)\}$ for $1 \leq k \leq m$. If $u_1u_2u_3u_4u_5 = v_1 \dots v_n\ w_1 \dots w_m$, then there exists a word $t \in \Sigma^+$ such that $u, v, w \in \{t, \theta(t)\}^+$.*

Proof. Unless the border between v_n and w_1 falls inside u_3, Theorem 4.2 concludes the existence of such t. So, assume that the border falls inside u_3. Even under this assumption, if the border between u_2 and u_3 falls inside some v_i except v_n, then Theorem 4.2 leads us to the same conclusion. Otherwise, we have that $(n-1)|v| < 2|u|$, which means $|v| < \frac{2}{n-1}|u| \leq \frac{2}{3}|u|$. Similarly, if the border between u_3 and u_4 does not fall inside w_1, we reach the existence of such t; otherwise $|w| < \frac{2}{m-1}|u| \leq \frac{1}{3}|u|$. Under the condition that v_n and w_1 straddle these respective borders, the equation cannot hold because v and w are too short. $\square$

We already know from Example 11 that for any $m \geq 1$, the triple $(5, 2, m)$ is not enough to impose θ-periodicity. So, we investigate next what would be the optimal bound for the extension of the Lyndon and Schützenberger result when the first parameter is 5. Note that, without loss of generality, we can assume $n \leq m$. Then, due to Theorem 4.13, all we have to investigate are the cases $(5, 3, m)$ for $m \geq 3$ and $(5, 4, m)$ for $m \geq 4$. The next intermediate lemma will be useful in the analysis of these cases.

Lemma 4.15. *Let $u \in \Sigma^+$ such that $u = xy$ and $y \in \mathrm{Pref}(u)$ for some θ-palindrome words $x, y \in \Sigma^+$. If $|y| \geq |x|$, then $\rho(x) = \rho(y) = \rho(u)$.*

Proof. We have $u = xy = yz$ for some $z \in \Sigma^+$ of the same length as x. The length condition implies that $x \in \mathrm{Pref}(y)$. Since $x = \theta(x)$ and $y = \theta(y)$, this means that $x \in \mathrm{Suff}(y)$ and hence $z = x$. So we have $u = xy = yx$, and hence x, y, and u share their primitive root. □

Unlike in the case of the original Lyndon-Schützenberger equation, the investigation of our extension entails the consideration of an enormous amount of cases since for each variable u_i, v_j, w_k we have two possible values. However, in almost all cases, it is enough to consider the common prefix between $u_1 \ldots u_\ell$ and $v_1 \ldots v_n$ or the common suffix between $u_1 \ldots u_\ell$ and $w_1 \ldots w_m$ to prove that either the equation imposes θ-periodicity or the equation cannot hold.

Note that for the $(5, 3, m)$ or $(5, 4, m)$ extensions of the Lyndon-Schützenberger equation, we only have to consider the case when the border between v_n and w_1 is inside u_3 because otherwise Theorem 4.2 immediately implies that $u, v, w \in \{t, \theta(t)\}^+$ for some word $t \in \Sigma^+$. Also even if the border is inside u_3, if $m|w| \geq 2|u| + |w|$, then we reach the same conclusion. Moreover, we can assume that w is θ-primitive since otherwise we would just increase the value of the parameter m. These observations justify the assumptions which will be made in the following propositions.

Proposition 4.16. *Let $u, v \in \Sigma^+$ such that v is a θ-primitive word, $u_1, u_2, u_3 \in \{u, \theta(u)\}$, and $v_1, \cdots, v_{2m+1} \in \{v, \theta(v)\}$ for some $m \geq 1$. If $v_1 \cdots v_{2m+1}$ is a proper prefix of $u_1u_2u_3$ and $2m|v| < 2|u| < (2m+1)|v|$, then $u_2 \neq u_1$ and $v_1 = \cdots = v_{2m+1}$. Moreover, $v_1 = yp$ and $u_1u_2 = (yp)^{2m}y$ for some $y, p \in \Sigma^*$ such that $y = \theta(y)$ and $p = \theta(p)$.*

Proof. Since θ is an involution, we may assume without loss of generality that $u_1 = u$ and $v_1 = v$. Note that $|v| < |u|$ and, due to the length condition, the border between u_1 and u_2 falls inside v_{m+1} while the one between u_2 and u_3 falls inside v_{2m+1}. Now, we have two cases depending on whether u_2 is equal to u_1.

Case 1: Suppose first that $u_2 \neq u_1$, i.e., $u_2 = \theta(u)$. Since $u_1u_2 = u\theta(u)$ is a θ-palindrome, $v_1 \cdots v_{2m} \in \mathrm{Pref}(u_1u_2)$ implies $\theta(v_{2m}) \cdots \theta(v_1) \in \mathrm{Suff}(u_1u_2)$. Applying Theorem 4.10 to the overlap between $v_1 \cdots v_{2m}$ and $\theta(v_{2m}) \cdots \theta(v_1)$ gives the following subcases: a) $v_1 = \cdots = v_{2m} = v$, and b) $v_1 = v, v_2 = \cdots = v_{2m} = \theta(v)$. For case b), because of the θ-primitivity of v, applying Theorem 4.10 to the overlap between $v_{2m}v_{2m+1}$ and $\theta(v_2)\theta(v_1)$ implies that v_{2m+1} can be neither v nor $\theta(v)$. Thus, this subcase is not possible.

Next, we consider the subcase a), and prove that v_{2m+1} must be v. Suppose otherwise, i.e., $v_{2m+1} = \theta(v)$, and we analyze two cases depending on whether u_3 is u or $\theta(u)$. If $u_3 = u$, then $v_{2m}v_{2m+1} = v\theta(v)$ overlaps with $\theta(v_1)v_1 = \theta(v)v$ because $v_1 \in \mathrm{Pref}(u)$, which contradicts Theorem 4.10. Otherwise, i.e., $u_3 = \theta(u)$, we look at the overlap between $v_{m+1} = v$ and $\theta(v_{m+1}) = \theta(v)$. Note that this overlap is a

θ-palindrome and, moreover, since the border between u_1 and u_2 cuts this overlap exactly in half, see Figure 4.7, we can say it is of the form $z\theta(z)$ for some $z \in \Sigma^+$. Then $v = z\theta(z)y$ for some θ-palindrome word y. Note that, due to length constraints, $z\theta(z) \in \text{Pref}(v_{2m+1})$ and $\theta(v) = yz\theta(z)$. If $|z\theta(z)| \geq |y|$, then Lemma 4.15 implies that $\rho(z\theta(z)) = \rho(y)$, which contradicts the θ-primitivity of v. Otherwise, since $|z\theta(z)| < |y|$ we have $z \in \text{Pref}(y)$, and hence $\theta(z) \in \text{Suff}(y)$. So, if we look at the border between u_2 and u_3, then $yz\theta(z)^2 = z\theta(z)^2 y$. Thus $\rho(y) = \rho(z\theta(z)^2)$, and hence $y, z \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$, again a contradiction with the θ-primitivity of v.

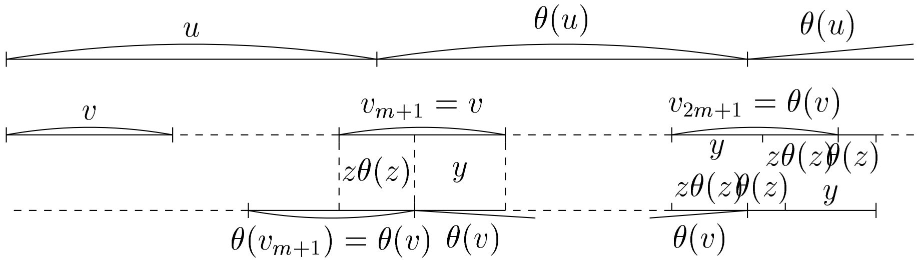

Figure 4.7: v_{m+1} overlaps with $\theta(v_{m+1})$ and the overlap is split exactly in half by the border between u_1 and u_2.

In conclusion, if $u_1 \neq u_2$, then $v_1 = \cdots = v_{2m+1}$ must hold. Theorem 4.10 gives the expressions of v and $u\theta(u)$ based on two θ-palindromes y and p.

Case 2: Now suppose that $u_2 = u_1 = u$, and we see that a contradiction occurs for all possible cases. If we look at the overlap between $v_1 \cdots v_m$ and $v_{m+1} \cdots v_{2m}$, then we see that all cases from Theorem 4.10 are possible.

Firstly we consider the subcase a) when $v_1 = \cdots = v_m = v$ and $v_{m+1} = \cdots = v_{2m} = \theta(v)$, which is illustrated in Figure 4.8. As mentioned before, the border between u_1 and u_2 falls inside v_{m+1}, and hence in this case $u_1 = v^m z$ for some $z \in \mathrm{Pref}(v_{m+1})$; moreover $|z| < \frac{1}{2}|v|$ since $2|u| < (2m+1)|v|$. Then, we can write $v_{m+1} = \theta(v) = zy$ for some $y \in \Sigma^+$ with $y = \theta(y)$, see Figure 4.8. Moreover, using length arguments, we have that the right end of u_2 falls inside v_{2m+1} after exactly $2|z|$ characters. Since $u = v^m z$ and $\theta(z) \in \mathrm{Suff}(v)$, we obtain $\theta(z)z \in \mathrm{Pref}(v_{2m+1})$. Also, since $\theta(v) = zy$ and $|z| < \frac{1}{2}|v|$, we have $|y| > |z|$.

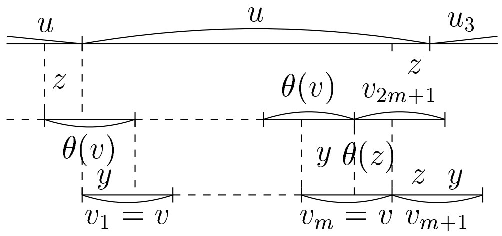

Figure 4.8: $v_1 \dots v_m$ and $v_{m+1} \dots v_{2m}$ overlap. Note that unless $u_3 = u$, we cannot assume that v_{m+1} overlaps with v_{2m+1}.

If $u_3 = u$, then $v_m v_{m+1} = v\theta(v)$ and $v_{2m}v_{2m+1} = \theta(v)v_{2m+1}$ overlap. So, due to Theorem 4.10, v_{2m+1} must be $\theta(v)$. So $z = \theta(z)$, and hence $\theta(v) = zy$ and $y \in \mathrm{Pref}(v_{m+1})$, i.e., $y \in \mathrm{Pref}(\theta(v))$. Then, since $|y| > |z|$, Lemma 4.15 implies $\rho(y) = \rho(z)$, which contradicts the θ-primitivity of v.

If $u_3 = \theta(u)$, then we consider two cases depending on the value of v_{2m+1}. First, suppose that $v_{2m+1} = v$. Since $\theta(z)z \in \mathrm{Suff}(u)$, we have $\theta(z)z \in \mathrm{Pref}(u_3)$ and we have two cases depending on $|v|$ and $2|\theta(z)z| = 4|z|$. If $|v| \le 4|z|$, then $v_{2m+1} = v = \theta(z)zx$ for some $x \in \mathrm{Pref}(\theta(z)z)$. Since $|y| = |x| + |z|$ and $y, \theta(z)z \in \mathrm{Pref}(v)$, we have $x \in \mathrm{Pref}(y)$ and $z \in \mathrm{Suff}(y)$, which means $y = xz$. Thus, we have $v = \theta(z)zx =$

$xz\theta(z)$. But we already know from [7] that this equation implies $x, z \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$, which contradicts the θ-primitivity of v. Otherwise, i.e., $4|z| < |v|$, since u_2u_3 is a θ-palindrome, $v_{2m}v_{2m+1}$ and $\theta(v_{2m+1})\theta(v_{2m})$ overlap with the overlap of length at least $|v|$.

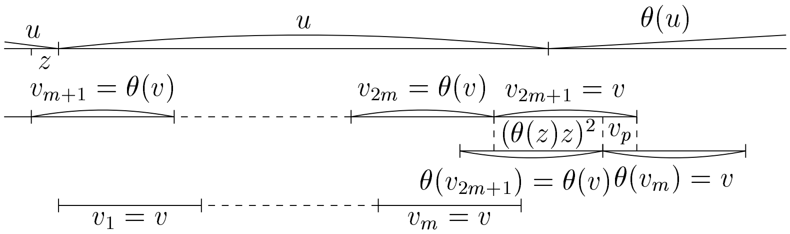

Figure 4.9: $v_{2m}v_{2m+1} = \theta(v)v$ overlaps with its image under θ. In addition, $v_{m+1} = \theta(v)$ overlaps with $v_1 = v$.

Since $|v| > 4|z|$, we can let $v = (\theta(z)z)^2v_p$ for some $v_p \in \text{Pref}(v) \cap \text{Suff}(v)$, see Figure 4.9. Then when we look at the overlap between $v_{m+1} = \theta(v)$ and $v_1 = v$, we can say that $\theta(v) = (z\theta(z))^2\theta(v_p)$. Hence $v = v_p(z\theta(z))^2 = (\theta(z)z)^2v_p$. Since $(z\theta(z))^2$ and $(\theta(z)z)^2$ are θ-palindromes, Lemma 4.6 leads to a contradiction with the θ-primitivity of v.

Next, suppose $v_{2m+1} = \theta(v)$. Then $z = \theta(z)$. If $2|z| < |v| \leq 4|z|$, then $v_{2m+1} = \theta(v) = z^kz_p$, where $k \in \{2, 3\}$ and $z_p \in \text{Pref}(z)$. This means that $z^3 \in \text{Suff}(u)$ since $z^2 \in \text{Suff}(v_m)$. It follows that $v_{2m} = \theta(v)$ has z as its suffix, which leads to a contradiction with the θ-primitivity of v since $\theta(v) = z^kz_p$. Otherwise, i.e., when $4|z| < |v|$, we can let $v_{2m+1} = \theta(v) = z^4v_p$ for some $v_p \in \text{Pref}(v)$ with $v_p = \theta(v_p)$ (refer to Figure 4.9, but keeping in mind that now $v_{2m+1} = \theta(v)$). Since

$v_1 = v$ overlaps with v_{m+1}, we have $z^3 \in \text{Pref}(v)$. Also the overlap between v_m and $v_{2m}v_{2m+1}$ implies that $v_p z \in \text{Suff}(v)$ (note that $v_p = \theta(v_p)$ in this case). Thus, $v = v_p z^4 = z^3 v_p z$, which contradicts the θ-primitivity of v since v_p is nonempty.

Secondly, we consider the subcase b). If $m \geq 2$, then $v_1 = \cdots = v_{m-1} = v$ and $v_m = \cdots = v_{2m} = \theta(v)$. This means that $v_{m-1}v_m = v\theta(v)$ and $v_{2m}v_{2m+1} = \theta(v)v_{2m+1}$ overlap, so Theorem 4.10 implies that v_{2m+1} cannot be either v or $\theta(v)$. For $m = 1$, we have $v_1 = v_2 = v$. If $v_3 = v$, the Fine and Wilf theorem implies that $\rho(u) = \rho(v)$. Then, however, the length conditions $|v| < |u| < 2|v|$ implies that v is not primitive, a contradiction. Thus, $v_3 = \theta(v)$. Since u starts with v, we can write $v = xy = yz$, for some $x, y, z \in \Sigma^+$ with $z = \theta(z)$ and $2|u| - 2|v| = 2|z|$, as illustrated in Figure 4.10. Thus, $x = pq$, $z = qp$, and $y = (pq)^i p$ for some $p, q \in \Sigma^*$ and $i \geq 0$. Moreover, since $2|u| < 3|v| < 3|u|$, we have $|v| > 2|z|$, which means that $z^2 \in \text{Suff}(v)$, i.e., $z^2 \in \text{Pref}(\theta(v))$. Hence $z \in \text{Suff}(u)$. But, we already had that $x \in \text{Suff}(u)$, which implies that $pq = qp$. Thus, $\rho(p) = \rho(q) = \rho(x) = \rho(y)$, which contradicts the θ-primitivity of v.

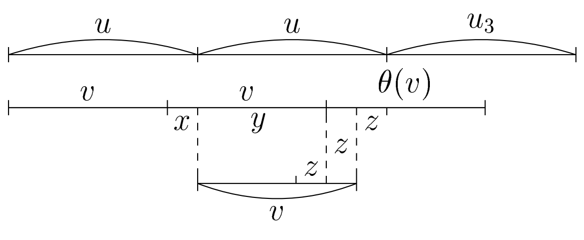

Figure 4.10: How $v_2v_3 = v\theta(v)$ and $v_1 = v$ overlap in the subcase b) for $m = 1$

Now we consider the other subcases. Note that in these subcases $m \geq 2$. The subcase c) when $v_1 = \cdots = v_m = v_{m+1} = v$ and $v_{m+2} = \cdots = v_{2m} = \theta(v)$ is illustrated in Figure 4.11. Then $v = yz$ for some $y, z \in \Sigma^+$ with $y = \theta(y)$, $z = \theta(z)$, and $y \in \mathrm{Suff}(v)$. Since $|y| \geq |z|$, Lemma 4.15 leads to a contradiction with the primitivity of v. The remaining subcases d) is when $v_1 = \cdots = v_{m-1} = v$, $v_m = \theta(v)$, $v_{m+1} = v$, and $v_{m+2} = \cdots = v_{2m} = \theta(v)$. In this subcase, $v_{m-1}v_m = v\theta(v)$ overlaps with $v_{2m}v_{2m+1} = \theta(v)v_{2m+1}$ with an overlapped part of length at least $|v|$. Theorem 4.10 implies that v_{2m+1} can be neither v nor $\theta(v)$.

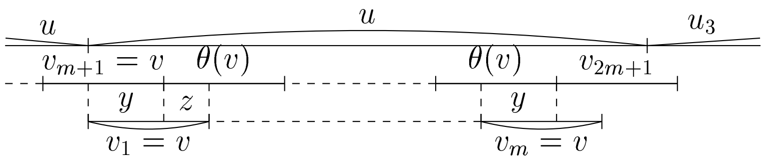

Figure 4.11: When $v_1 = \cdots = v_m = v_{m+1} = v$ and $v_{m+2} = \cdots = v_{2m} = \theta(v)$

To conclude, we showed that $u_2 \neq u_1$ and $v_1 = \cdots = v_{2m+1}$. $\square$

Proposition 4.17. *Let $u, v \in \Sigma^+$ such that v is θ-primitive, $u_1, u_2, u_3 \in \{u, \theta(u)\}$, and $v_1, \cdots, v_{2m} \in \{v, \theta(v)\}$ for some $m \geq 2$. If $v_1 \cdots v_{2m} \in \mathrm{Pref}(u_1u_2u_3)$ and $(2m-1)|v| < 2|u| < 2m|v|$, then either $u_1 \neq u_2$ and $v_1 = \cdots = v_{2m}$, with $v_1 = yp$ and $u_1u_2 = (yp)^{2m-1}y$ for some $y, p \in \Sigma^*$ such that $y = \theta(y)$ and $p = \theta(p)$, or $u_1 = u_2$, $v_1 = \cdots = v_m$, and $v_{m+1} = \cdots = v_{2m} = \theta(v_1)$, with $u_1 = [r(tr)^i(rt)^{i+j}r]^{m-1}r(tr)^i(rt)^j$ and $v_1 = r(tr)^i(rt)^{i+j}r$ for some $i \geq 0$, $j \geq 1$, and $r, t \in \Sigma^*$ such that $r = \theta(r)$, $t = \theta(t)$, and rt is primitive.*

Proof. Just as in the proof of Proposition 4.16, we can assume without loss of generality that $u_1 = u$ and $v_1 = v$. Then, we analyze two cases depending on whether $u_2 = u_1$.

Case 1: Let us look first at the case when $u_2 \neq u_1$, i.e., $u_2 = \theta(u)$, which differs only slightly from Case 1 from the proof of Proposition 4.16. Indeed, it is enough to consider only the case when $u_3 = \theta(u)$, $v_1 = \cdots = v_{2m-1} = v$ and prove that $v_{2m} = v$. Let us suppose for now that $v_{2m} = \theta(v)$ and let $v = yx$ and $x = z\theta(z)$ for some $x, y, z \in \Sigma^+$ with $x = \theta(x)$ and $y = \theta(y)$, as illustrated in Figure 4.12.

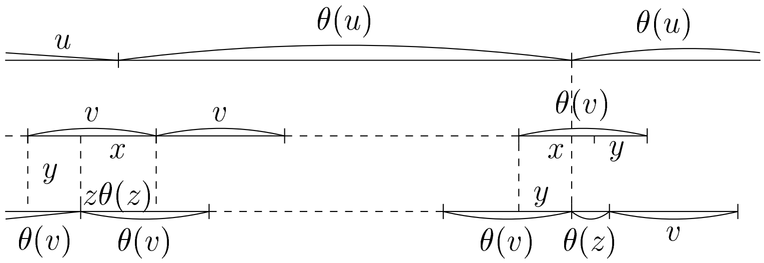

Figure 4.12: When $u_3 = \theta(u)$, $v_{2m} = \theta(v) = xy$ overlaps with $y\theta(z)v$ because $\theta(z)v \in \mathrm{Pref}(\theta(u))$.

Note that $\theta(v) = xy = z\theta(z)y$ and $y \in \mathrm{Pref}(\theta(v))$. If $|y| \geq |x|$, then Lemma 4.15 implies that $\rho(x) = \rho(y)$, which is a contradiction with θ-primitivity of v. If $|z| \leq |y| < |x|$, then $z \in \mathrm{Pref}(y)$ and $z\theta(z)y \in \mathrm{Pref}(y\theta(z)y)$, as illustrated in Figure 4.12. Thus, $z\theta(z)y = y\theta(z)z$, which implies that $y, z \in \{t, \theta(t)\}^+$, see [7], contradicting the θ-primitivity of v. If $\frac{1}{2}|z| \leq |y| < |z|$, then we have $y \in \mathrm{Pref}(z)$ and $y\theta(z)y \in \mathrm{Pref}(z\theta(z)y)$, see Figure 4.13 *i)*. Then, let $\theta(z) = z_1 y = y z_2$ for some $z_1, z_2 \in \Sigma^+$ with $z_1 = \theta(z_1)$ and $z_2 = \theta(z_2)$ since $y \in \mathrm{Pref}(z)$. Then, since $zy = y\theta(z)$ we have $z_2 y^2 = y^2 z_2$, and hence $\rho(y) = \rho(z_2)$, which contradicts the θ-primitivity of v as $v = yx = yz_2y^2z_2$. If $|y| < \frac{1}{2}|z|$, then we have $\theta(z) = z_3 y = y^2 z_4$ for some $z_3, z_4 \in \Sigma^+$

with $z_3 = \theta(z_3)$ and $z_4 = \theta(z_4)$, see Figure 4.13 *ii)*. Now, since $zy = y\theta(z)$, we have $z_4y^3 = y^3z_4$. This leads us to the same contradiction as above because $v = yz_4y^4z_4$.

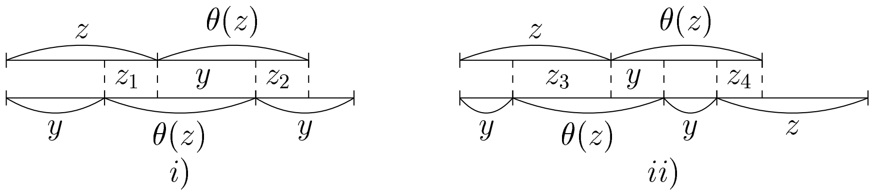

Figure 4.13: How $v_{2m} = z\theta(z)y$ overlaps with $y\theta(z)v$ when *i)* $\frac{1}{2}|z| \le |y| \le |z|$, or *ii)* $|y| \le \frac{1}{2}|z|$ in Case 1 of Proposition 4.17

Thus, if $u_1 \neq u_2$, then we must have $v_1 = \cdots = v_{2m} = v$. The representations of v_1 and u_1u_2 can be obtained using Theorem 4.10.

Case 2: Let us look next at the case when $u_2 = u_1 = u$, illustrated in Figure 4.14 and let $v = xy$ with $x \in \mathrm{Suff}(v_m)$ and $y \in \mathrm{Pref}(v_{m+1})$. Moreover, note that $|x| < |y|$ since $|x| = m|v| - |u|$ and $(2m-1)|v| < 2|u|$. Now, if we look at the overlap between $v_1 \cdots v_m$ and $v_m \cdots v_{2m-1}$, then, due to Theorem 4.10, we get the following subcases: a) $v_1 = \cdots = v_{m-1} = v$ and $v_m = v_{m+1} = \cdots = v_{2m-1} = \theta(v)$; b) $v_1 = \cdots = v_m = v$, $v_{m+1} = \cdots = v_{2m-1} = \theta(v)$.

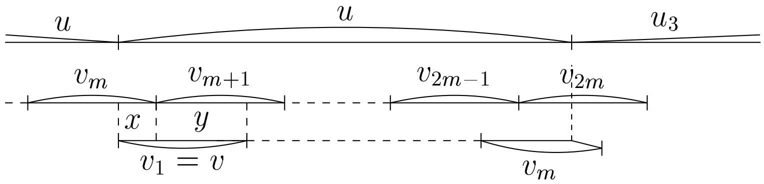

Figure 4.14: If $u_2 = u$, we can regard that $v_1 \ldots v_m$ overlaps with $v_m \ldots v_{2m-1}$ not depending on the value of u_3.

First, let us consider the subcase a). If $u_3 = u$, then $v_{m-1}v_m = v\theta(v)$ overlaps with $v_{2m-1}v_{2m} = \theta(v)v_{2m}$ and thus, due to Theorem 4.10, v_{2m} cannot be either

v or $\theta(v)$. Otherwise, $u_3 = \theta(u)$ and note that $x = \theta(x)$ and $y = \theta(y)$ since $v_m = v_{m+1} = \theta(v)$. Then, since the overlapped part between v_{2m-1} and v_m is x, we obtain $x \in \text{Pref}(\theta(v))$. Since $\theta(v) = yx$ and $|x| < |y|$, we have $x \in \text{Pref}(y)$, i.e., $x \in \text{Suff}(y)$. Thus $x \in \text{Suff}(u)$, that is, $x \in \text{Pref}(\theta(u))$. Since $u_3 = \theta(u)$ and $v_m = \theta(v) = yx$, we can say that $v_{m-1}v_m$ overlaps with $v_{2m-1}v_{2m}$, which results in the same conclusion as above. Thus, the subcase a) is not possible.

For the subcase b), we prove that $v_{2m} = \theta(v)$. Let us start our analysis by supposing that $v_{2m} = v$. First, since $v_m = v$ ends with x, let $v = zwx$ for some $z, w \in \Sigma^+$ with $|w| = |x|$. If $u_3 = \theta(u)$, since $v_{2m} = v = zwx$, we obtain that $w \in \text{Pref}(u_3)$, i.e., $\theta(w) \in \text{Suff}(u)$. But this means that $w = \theta(w)$, since the right end of the first u cuts $v_m = v = zwx$ after exactly $|zw|$ characters. Since the overlap between v_{2m-1} and v_m is x, we have $xz = zw$ with $x = \theta(x)$ and $w = \theta(w)$. Then Lemma 4.6 implies that $x, z, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$, a contradiction with the θ-primitivity of $v = zwx$. If $u_3 = u$, we immediately obtain $v = xy$ with $y \in \text{Pref}(v)$. Since $|x| < |y|$, the same contradiction derives from these relations due to Lemma 4.15.

In conclusion, for this case, i.e., when $u_2 = u_1$, we obtain that $v_1 = \cdots = v_m = v$ and $v_{m+1} = \cdots = v_{2m} = \theta(v)$. By applying Theorem 4.10 to the overlap between $v_1 \ldots v_m$ and $v_m \ldots v_{2m-1}$, we get the representations of u and v by two θ-palindromes r and t. $\square$

These propositions show that if we suppose v to be θ-primitive, then the values of u_1, u_2, u_4, and u_5 determine the values of $v_1, \ldots, v_n$ and $w_1, \ldots, w_m$ uniquely, modulo a substitution of v by $\theta(v)$, or of w by $\theta(w)$. Thus, they decrease significantly the number of cases to be considered. Furthermore, the value of u_3 may put an additional useful restriction on v or w.

Lemma 4.18. *Let $u, v \in \Sigma^+$ such that v is a θ-primitive word, $u_1, u_2, u_3 \in \{u, \theta(u)\}$, and $v_1, \cdots, v_n \in \{v, \theta(v)\}$ for some $n \geq 3$. If $v_1 \cdots v_n \in \mathrm{Pref}(u_1u_2u_3)$, $u_1 \neq u_2$, $u_1 = u_3$, and $(n-1)|v| < 2|u| < n|v|$, then $|v| < \frac{4}{2n-1}|u|$.*

Proof. Without loss of generality, we can assume that $u_1 = u$ and $v_1 = v$ because θ is an involution. Propositions 4.16 and 4.17 imply that $v_1 = \cdots = v_n = v$. Hence $u\theta(u) = v^{n-1}x$ for some $x \in \mathrm{Pref}(v)$. Since $u\theta(u)$ is a θ-palindrome, $v^{n-1}x = \theta(x)\theta(v)^{n-1}$ and this implies that $x = \theta(x)$ and $v = yx$ for some nonempty θ-palindrome y (see Figure 4.15).

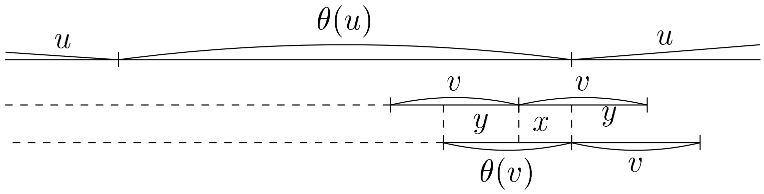

Figure 4.15: Since u begins with v, y is a prefix of v.

Since $v \in \mathrm{Pref}(u)$, we obtain that $y \in \mathrm{Pref}(v)$. If $|x| \leq |y|$, then Lemma 4.15 leads to a contradiction with the θ-primitivity of v. Thus $|y| < |x|$, which implies that $|y| < \frac{1}{2}|v|$. This means that $|v| < \frac{4}{2n-1}|u|$ because $|y| = n|v| - 2|u|$. $\square$

All we did so far in studying the extended Lyndon-Schützenberger equation $u_1 \dots u_5 = v_1 \dots v_n \; w_1 \dots w_m$ was to consider either the common prefix of $v_1 \dots v_n$ and $u_1 \dots u_5$, or the common suffix of $w_1 \dots w_m$ and $u_1 \dots u_5$. Next, we combine them together and consider the whole equation. The following lemma proves to be useful for our considerations.

Lemma 4.19. *Let $u, v \in \Sigma^+$ such that v is a θ-primitive word, $u_1, u_2, u_3 \in \{u, \theta(u)\}$ and $v_1, \cdots, v_n \in \{v, \theta(v)\}$ for some $n \geq 3$. If $v_1 \cdots v_n = u_1 u_2 z$ for some $z \in \mathrm{Pref}(u_3)$, $u_1 = u_2$, and $(n-1)|v| < 2|u|$, then $v_1 = xyx$ and $z = x^2$ for some $x, y \in \Sigma^+$ such that $x = \theta(x)$ and $yx = \theta(yx)$.*

Proof. Just as before, we assume that $u_1 = u$ and $v_1 = v$. Propositions 4.16 and 4.17 imply that $n = 2m$ for some $m \geq 2$, $u = \{r(tr)^i(rt)^{i+j}r\}^{m-1}r(tr)^i(rt)^{i+j}$, and $v = r(tr)^i(rt)^{i+j}r$ for some $r, t \in \Sigma^*$ such that $r = \theta(r)$, $t = \theta(t)$, $i \geq 0$, and $j \geq 1$. By taking $x = r(tr)^i$ and $y = (rt)^j$, we complete the proof. □

Next, we prove that the triple $(5, 4, m)$ imposes θ-periodicity for any $m \geq 4$.

Theorem 4.20. *Let $u, v, w \in \Sigma^+$, $u_1, u_2, u_3, u_4, u_5 \in \{u, \theta(u)\}$, $v_1, v_2, v_3, v_4 \in \{v, \theta(v)\}$, and $w_1, \cdots, w_m \in \{w, \theta(w)\}$ for some $m \geq 4$. If these words satisfy $u_1u_2u_3u_4u_5 = v_1v_2v_3v_4 \; w_1 \cdots w_m$, then u is not θ-primitive and $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. First note that we can assume that w is θ-primitive, since otherwise we would just increase the numerical parameter m. If u is not θ-primitive, that is,

$u \in \{p, \theta(p)\}^k$ for some θ-primitive word $p \in \Sigma^+$ and $k \geq 2$, then the equation can be rewritten as $p_1 p_2 \cdots p_{5k} = v_1 v_2 v_3 v_4 w_1 \ldots w_m$, where $p_i \in \{p, \theta(p)\}$ for $1 \leq i \leq 5k$. But then, due to Theorem 4.12, we obtain that $v, w \in \{p, \theta(p)\}^+$. Furthermore, we can assume that also v is θ-primitive. Indeed, if it is not, then $v \in \{q, \theta(q)\}^j$ for some θ-primitive word q and $j \geq 2$. Then, the equation becomes $u_1 \ldots u_5 = q_1 \ldots q_{4j} w_1 w_2 \ldots w_m$, where $q_i \in \{q, \theta(q)\}$ for $1 \leq i \leq 4j$. But this implies that $u, w \in \{q, \theta(q)\}^+$ due to Theorem 4.14. Since u and w are assumed to be θ-primitive, $u, w \in \{q, \theta(q)\}$ and we have $5|q| < 4j|q| + m|q|$, which contradicts the fact that u, v, and w satisfy the equation $u_1 \ldots u_5 = q_1 \ldots q_{4j} w_1 w_2 \ldots w_m$. Even when v is θ-primitive, if $m \geq 7$ then the same argument leads to the same contradiction.

Now we will show that if u, v, and w are θ-primitive, then the equation cannot hold for $m \leq 6$. Since θ is an involution, we can assume that $u_1 = u$, $v_1 = v$, and $w_1 = w$. Let us start by supposing that u, v, and w satisfy $u_1 u_2 u_3 u_4 u_5 = v_1 v_2 v_3 v_4 \ w_1 \cdots w_m$. Now, we have several cases depending on where the border between v_4 and w_1 is located. If it is left to or on the border between u_2 and u_3, then Theorem 4.2 implies that $u, w \in \{t, \theta(t)\}^+$ for some θ-primitive word $t \in \Sigma^+$, which further implies that also $v \in \{t, \theta(t)\}^+$. In fact, $u, v, w \in \{t, \theta(t)\}$ because they are θ-primitive. Then $|u_1 \ldots u_5| = 5|t|$, while $|v_1 v_2 v_3 v_4 w_1 \ldots w_m| = (4 + m)|t|$ with $m \geq 4$, which is a contradiction. The case when the border between v_4 and w_1 is right to or on the border between u_3 and u_4 will lead the contradiction along the same argument.

So let us suppose that $|u_1u_2| < |v_1v_2v_3v_4| < |u_1u_2u_3|$. Note that under this supposition, $|v|, |w| < |u|$. If $m|w| \geq 2|u| + |w| - 1$, then $u_3u_4u_5$ and $w_1 \dots w_m$ share a suffix long enough to impose the θ-periodicity onto u and w due to Theorem 4.2. However, as explained before, this leads to a contradiction. This argument also applies to $u_1u_2u_3$ and $v_1v_2v_3v_4$. As a result, it is enough to consider the case when $3|v| < 2|u| < 4|v|$ and $(m-1)|w| < 2|u| < m|w|$.

There are 16 cases to be considered depending on the values of u_2, u_3, u_4, and u_5. Note that once these values are determined, the values of v_1, v_2, v_3, v_4 and $w_1, \cdots, w_m$ are set uniquely due to Propositions 4.16 and 4.17. We number these cases from 0 to 15 by regarding $u_2u_3u_4u_5$ as the 4-bit number based on the conversion $u \to 0$ and $\theta(u) \to 1$. For example, case 5 is $u_2u_3u_4u_5 = u\theta(u)u\theta(u)$.

First, we consider the case 2, that is, $uuu\theta(u)u = v_1 \cdots v_4 \; w_1 \cdots w_m$. Since $3|v| < 2|u| < 4|v|$, $|v| < \frac{2}{3}|u|$. Moreover, Lemma 4.18 implies that $|w| < \frac{4}{2m-1}|u|$. Then $5|u| - (4|v| + m|w|) > 0$ which contradicts the fact that u, v, and w satisfy the given equation. The same arguments work for the cases when either $u_1u_2u_3 = u\theta(u)u$ (i.e., cases 8, 9, 10, 11), or $u_3u_4u_5 = u\theta(u)u$ (i.e., cases 2, 10), or $u_3u_4u_5 = \theta(u)u\theta(u)$ (i.e., cases 5, 13).

Secondly we consider the case 1, that is, $uuuu\theta(u) = v_1 \cdots v_4w_1 \cdots w_m$. Let $uux = v_1 \cdots v_4$, $yu\theta(u) = w_1 \cdots w_m$ for some $x, y \in \Sigma^+$ such that $u = xy$. We immediately obtain now, due to Lemma 4.19, that $x = \theta(x)$. Since $x \in \mathrm{Pref}(u_3)$, this means that $x \in \mathrm{Suff}(u_5)$, which implies that $w_m \in \mathrm{Suff}(x)$ or $x \in \mathrm{Suff}(w_m)$.

In both cases, we obtain that $u_3u_4u_5$ and $w_mw_1w_2\dots w_m$ share a common suffix of length at least $2|u|+|w|-1$. Then we employ Theorem 4.2 to lead a contradiction. Among the cases left to be investigated, the only one where we cannot apply this technique is case 0.

Now, case 0 is $u_1 = u_2 = u_3 = u_4 = u_5 = u$. Applying Propositions 4.16 and 4.17, we have that $m = 2k$ for some $k \geq 2$, $w_1 = \cdots = w_k = w$, $w_{k+1} = \cdots = w_{2k} = \theta(w)$, $v_1 = v_2 = v$, and $v_3 = v_4 = \theta(v)$. Note that $k \in \{2, 3\}$ since $4 \leq m \leq 6$. Then, Lemma 4.19 implies that $u = xyxxy = (y'x'x')^{k-1}y'x' = x^2x'^2$, $v = xyx$, and $\theta(w) = x'y'x'$ for some $x, y, x', y' \in \Sigma^+$ with $x = \theta(x)$, $yx = \theta(yx)$, $x' = \theta(x')$, and $x'y' = \theta(x'y')$.

When $k = 2$, i.e., $xyxxy = y'x'x'y'x'$, we have three subcases depending on the lengths of xy and $y'x'$. If $|xy| < |y'x'|$, then by looking at the two sides of the equality $xyxxy = y'x'x'y'x'$, we obtain $y'x' = xyz = \theta(z)xy$ and $x = zx'\theta(z)$ for some $z \in \Sigma^+$. Substituting $x = zx'\theta(z)$ into $xyz = \theta(z)xy$ we get $z = \theta(z)$, and hence $y'x' = xyz = zxy$. Thus, $y'x', xy, z \in \{p\}^+$ for some primitive word p. Let $z = p^i$ and $y'x' = p^j$ for some $i, j \geq 1$. Then $y'x' = zxy$ and $x = zx'z$ imply that $p^j = p^{2i}x'p^iy$. Since p is primitive, we obtain that $\rho(x') = p$, which contradicts the θ-primitivity of $\theta(w) = x'y'x'$. For the case when $|xy| > |y'x'|$ we can use similar arguments to reach a contradiction. Finally, if $|xy| = |y'x'|$, then $x = x'$, which is a contradiction with the θ-primitivity of u since $u = xxx'x'$.

When $k = 3$, i.e., $u = xyxxy = (y'x'x')^2y'x'$, we first note that $|xy| > |y'x'|$ and

$|xyx| > |y'x'x'|$. If $|xy| \geq |y'x'x'|$, then, by the Fine and Wilf theorem, $\rho(xyx) = \rho(y'x'x')$. Since xyx is strictly longer than $y'x'x'$, this means that $v = xyx$ is not primitive, which is a contradiction. Otherwise, i.e., $|y'x'| < |xy| < |y'x'x'|$, let $xy = y'x'z$ for some $z \in \mathrm{Pref}(x')$. Since $x' = \theta(x')$, the equation $xyxxy = (y'x'x')^2y'x'$ also implies that $xy = \theta(z)y'x'$. Moreover, since $xy = y'x'z = \theta(z)y'x'$ and $\theta(z) \in \mathrm{Suff}(x')$, we obtain $z = \theta(z)$. Thus $xy, y'x', z \in \{q\}^+$ for some primitive word $q \in \Sigma^+$, which, just as above, contradicts the θ-primitivity of $\theta(w)$. $\square$

The next result shows that the triple $(5, 3, m)$ also imposes θ-periodicity for any $m \geq 3$.

Theorem 4.21. *Let $u, v, w \in \Sigma^+$, $u_1, u_2, u_3, u_4, u_5 \in \{u, \theta(u)\}$, $v_1, v_2, v_3 \in \{v, \theta(v)\}$, and $w_1, \cdots, w_m \in \{w, \theta(w)\}$ with $m \geq 3$. If these words verify the equation $u_1u_2u_3u_4u_5 = v_1v_2v_3\ w_1 \cdots w_m$, then u is not θ-primitive and $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. As in the proof of Theorem 4.20, we can assume that w is θ-primitive. Also if u is not θ-primitive, then, just as before, Theorem 4.12 results in $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$. So let us assume that u is θ-primitive. Moreover, we can assume that v is θ-primitive. Indeed, if it is not, then $v \in \{p, \theta(p)\}^j$ for some θ-primitive word p and $j \geq 2$. Then the equation becomes $u_1u_2u_3u_4u_5 = p_1 \ldots p_{3j}w_1w_2 \ldots w_m$, where $p_i \in \{p, \theta(p)\}$ for $1 \leq i \leq 3j$. For the case $m \geq 5$ and the case $m = 4$, Theorems 4.13 and 4.20 lead us to the contradiction, respectively. If $m = 3$, we can change the roles of v and w, and reduce it to the case when v is θ-primitive. In the following, we assume that u, v, and w are θ-primitive and prove that the equation cannot hold.

Now, since θ is an involution, we can assume that $u_1 = u$, $v_1 = v$, and $w_1 = w$. As in the proof of Theorem 4.20, in all cases except when the border between v_3 and w_1 falls inside u_3, we get a contradiction. Furthermore, using the same arguments as in the previous proof, we can assume that $2|v| < 2|u| < 3|v|$ and $(m-1)|w| < 2|u| < m|w|$. Moreover, due to Proposition 4.16, $u_2 = \theta(u)$ and $v_1 = v_2 = v_3 = v$, see Figure 4.16. Then $u\theta(u)x = v^3$ for some $x \in \Sigma^+$, which satisfies $x = \theta(x)$ due to the same proposition. Since $x \in \text{Pref}(u_3)$, if $u_3 \neq u_5$, then $x \in \text{Suff}(u_5)$ which implies that either $w_m \in \text{Suff}(x)$ or $x \in \text{Suff}(w_m)$. In both cases, we obtain that $u_3u_4u_5$ and $w_m w_1 w_2 \cdots w_m$ share a common suffix of length at least $2|u| + |w| - 1$. Hence, Theorem 4.2 implies that $u, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$ and thus also $v \in \{t, \theta(t)\}^+$ which leads to the same contradiction as above. Otherwise, $u_3 = u_5$ and we have the following four cases left:

1. $u\theta(u)u\theta(u)u = vvvw_1 \cdots w_m$,

2. $u\theta(u)\theta(u)u\theta(u) = vvvw_1 \cdots w_m$,

3. $u\theta(u)uuu = vvvw_1 \cdots w_m$,

4. $u\theta(u)\theta(u)\theta(u)\theta(u) = vvvw_1 \cdots w_m$.

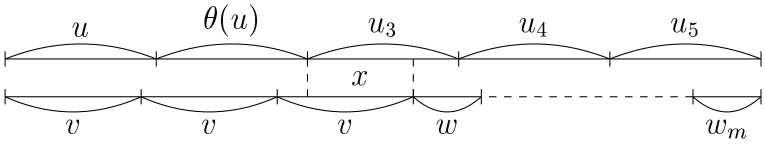

Figure 4.16: $u_1u_2u_3u_4u_5 = v_1v_2v_3w_1\cdots w_m$ for Theorem 4.21

Let us start by considering the first equation. Since v is θ-primitive, using Lemma 4.18, we have $|v| < \frac{4}{5}|u|$ and $|w| < \frac{4}{2m-1}|u|$. However, then $5|u| - (3|v| + m|w|) > 5|u| - \frac{12}{5}|u| - \frac{4m}{2m-1}|u| = \frac{6m-13}{5(2m-1)}|u| > 0$ because $m \geq 3$. Hence, $5|u| > 3|v| + m|w|$ contradicting our supposition that the words u, v, and w satisfy the equation $u\theta(u)u\theta(u)u = vvvw_1\cdots w_m$.

For the second equation, Propositions 4.16 and 4.17 imply that $w_1 = w_2 = \cdots = w_m = w$. Since $u\theta(u) = v^2v_p$ for some $v_p \in \text{Pref}(v)$ and $u\theta(u)$ is a θ-palindrome, we have $u\theta(u) = \theta(v_p)\theta(v)^2$. Note that $\theta(v_p) \in \text{Suff}(\theta(v))$. Also $u\theta(u) = w_s w^{m-1}$ for some $w_s \in \text{Suff}(w)$. Since $m \geq 3$, the Fine and Wilf theorem implies that $\rho(\theta(v)) = \rho(w)$ and thus we obtain again the same contradiction as above.

Next we consider the third equation. Since $u_4 = u_5$, Propositions 4.16 and 4.17 imply that $m = 2k$ for some $k \geq 2$ and $w_1 = \cdots = w_k = w$ and $w_{k+1} = \cdots = w_{2k} = \theta(w)$. Let $w^k\theta(w)^k = z_1z_2u^2$ for some $z_1, z_2 \in \Sigma^+$ with $|z_1| = |z_2| = k|w| - |u|$, as illustrated in Figure 4.17.

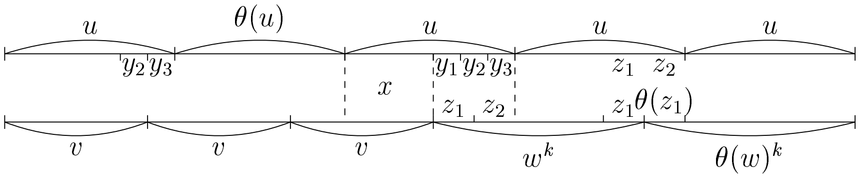

Figure 4.17: The suffix of u_3 can be written in two ways as $y_1y_2y_3$ and z_1z_2.

Then, $z_1z_2 \in \mathrm{Suff}(u)$, which due to length conditions means that $z_1 \in \mathrm{Suff}(w^k)$. Thus, $\theta(z_1) \in \mathrm{Pref}(\theta(w)^k)$ which implies immediately that $z_2 = \theta(z_1)$. Similarly, we can let $u\theta(u)u = v^3y_1y_2y_3$ for some $y_1, y_2, y_3 \in \Sigma^+$ with $|y_1| = |y_2| = |y_3| = |u| - |v|$. Then $y_1y_2y_3 = z_1\theta(z_1)$, which implies $y_3 = \theta(y_1)$ and $y_2 = \theta(y_2)$. Recall that $(2k-1)|w| < 2|u| < 2k|w|$ was assumed. So we have $|y_1y_2y_3| < |w|$ and $|w| < \frac{2}{2k-1}|u| \leq \frac{2}{3}|u|$. Thus, $|y_1y_2y_3| < \frac{2}{3}|u|$. This further implies that $|x| = |u| - |y_1y_2y_3| > |y_1|$. If we look at the second v, since $y_3 \in \mathrm{Suff}(u)$, using length arguments, we obtain that $y_3 \in \mathrm{Pref}(v)$, and hence $y_3 \in \mathrm{Pref}(u)$. Since $|y_3| < |x|$, this means that $y_3 \in \mathrm{Pref}(x)$ and hence $\theta(y_3) \in \mathrm{Suff}(x)$, which further implies $\theta(y_3) \in \mathrm{Suff}(v)$. Thus $y_2 = \theta(y_3)$ because $y_2 \in \mathrm{Suff}(v)$, which results in $y_1 = y_2 = y_3$ and, moreover, they are all θ-palindromes. Hence $y_1y_2 = \theta(y_2)\theta(y_1) = \theta(y_1y_2)$, which is a prefix of $\theta(v)$. This means that $u\theta(u)u$ and $v^3\theta(v)$ share a prefix of length at least $2|u|+|v|-1$. Consequently $\rho_\theta(u) = \rho_\theta(v)$ which leads to the same contradiction as before.

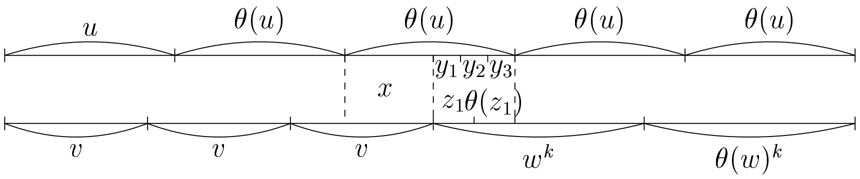

Figure 4.18: The suffix of $u_3 = \theta(u)$ can be written in two ways as $y_1y_2y_3$ and $z_1\theta(z_1)$.

Lastly, we consider the fourth equation, illustrated in Figure 4.18. Just as in the case of the third equation, $y_3 = \theta(y_1)$ and $y_2 = \theta(y_2)$. Since $y_2y_3 \in \mathrm{Suff}(\theta(u))$, these equalities give $\theta(y_3)\theta(y_2) = y_1y_2 \in \mathrm{Pref}(u) \subseteq \mathrm{Pref}(v^2)$. Thus, we can see that

$u\theta(u)^2$ and v^5 share their prefix of length at least $2|u| + |v|$. The rest is as same as for the third equation.

In conclusion, if u is θ-primitive, then, using length arguments, we always reach a contradiction. On the other hand, if u is not θ-primitive, then we proved that there exists a word $t \in \Sigma^+$ such that $u, v, w \in \{t, \theta(t)\}^+$. $\square$

Combining Theorems 4.12, 4.13, 4.20, and 4.21 and Examples 10 and 11 all together, now we conclude our analysis on the extended Lyndon-Schützenberger equation with the summarizing theorem.

Theorem 4.22. *For $u, v, w \in \Sigma^+$, let $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. If $u_1 \ldots u_\ell = v_1 \ldots v_n \; w_1 \ldots w_m$ and $\ell \geq 5$, $n, m \geq 3$, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$. Furthremore, $n = 3$ and $m = 3$ are optimal.*

4.5 Conclusion

This paper continues the investigation of an extended notion of primitiveness of words, based on replacing the identity between words by a weaker notion of "equivalence" between a word u and $\theta(u)$, where θ is a given antimorphic involution. Firstly, we completely characterize all non-trivial overlaps between two words in $\{v, \theta(v)\}^+$ of the form $\alpha(v, \theta(v)) \cdot x = y \cdot \beta(v, \theta(v))$. As an application of this characterization, we extend the Lyndon-Schützenberger equation to the equation

$u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m$, where $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. The strongest result obtained states that for $\ell \geq 5$ and $n, m \geq 3$, $u, v, w \in \{t, \theta(t)\}^+$ for some word t, while once n or m become 2, the existence of such t is not guaranteed any more.

Bibliography

[1] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 329–438. Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[2] S. Constantinescu and L. Ilie. Fine and Wilf's theorem for Abelian periods. *Bulletin of the EATCS*, 89:167–170, June 2006.

[3] M. Crochemore, C. Hancart, and T. Lecroq. *Algorithms on Strings*. Cambridge University Press, 2007.

[4] M. Crochemore and W. Rytter. *Jewels of Stringology*. World Scientific, 2002.

[5] L. J. Cummings and W. F. Smyth. Weak repetitions in strings. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 24:33–48, 1997.

[6] E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schützenberger result to pseudoperiodic words. In V. Diekert and D. Nowotka, editors, *Proc. DLT09*, volume 5583 of *Lecture Notes in Computer Science*, pages 183–194, Berlin, 2009. Springer-Verlag.

[7] E. Czeizler, L. Kari, and S. Seki. On a special class of primitive words. *Theoretical Computer Science*, 411(3):617–630, 2010.

[8] A. de Luca and A. De Luca. Pseudopalindrome closure operators in free monoids. *Theoretical Computer Science*, 362:282–300, 2006.

[9] N. J. Fine and H. S. Wilf. Uniqueness theorem for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, February 1965.

[10] T. Harju and D. Nowotka. The equation $x^i = y^j z^k$ in a free semigroup. *Semigroup Forum*, 68:488–490, 2004.

[11] T. Harju and D. Nowotka. On the equation $x^k = z_1^{k_1} z_2^{k_2} \dots z_n^{k_n}$ in a free semigroup. *Theoretical Computer Science*, 330(1):117–121, 2005.

[12] L. Kari, S. Konstantinidis, E. Losseva, P. Sosík, and G. Thierrin. A formal language analysis of DNA hairpin structures. *Fundamenta Informaticae*, 71(4):453–475, 2006.

[13] L. Kari and K. Mahalingam. Watson-Crick conjugate and commutative words. In *Proc. of DNA 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 273–283, 2008.

[14] L. Kari and K. Mahalingam. Watson-Crick palindromes in DNA computing. *Natural Computing*, 2009. DOI: 10.1007/s11047-009-9131-2.

[15] L. Kari and S. Seki. On pseudoknot-bordered words and their properties. *Journal of Computer and System Sciences*, 75:113–121, 2009.

[16] A. Lentin. Sur l'équation $a^m = b^n c^p d^q$ dans un mono#ide libre. *Comptes Rendus de l'Académie des Sciences Paris*, 260:3242–3244, 1965.

[17] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications.* Addison-Wesley, 1983.

[18] R. C. Lyndon and M. P. Schützenberger. The equation $a^m = b^n c^p$ in a free group. *Michigan Mathematical Journal*, 9:289–298, 1962.

[19] G. Păun, G. Rozenberg, and T. Yokomori. Hairpin languages. *International Journal of Foundations of Computer Science*, 12(6):837–847, 2001.

[20] J. Ziv and A. Lempel. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, IT-23(3):337–343, May 1977.

Chapter 5

An improved bound for the extended Fine and Wilf's theorem

This chapter consists of the latest updates on the extended Fine and Wilf's theorem discussed in Chapter 3. A paper[1] on these results has just been accepted for the publication in Fundamenta Informaticae (as of August 13, 2010) as follows:

L. Kari and S. Seki.

An improved bound for an extension of Fine and Wilf's theorem and its optimality. *Fundamenta Informaticae* 101(3) (2010) 215-236.

Summary: Considering two DNA molecules which are Watson-Crick (WK) complementary to each other "equivalent" with respect to the information they encode enables us to extend the classical notions of repetition, period, and power. WK-complementarity has been modelled mathematically by an antimorphic involution θ, i.e., a function θ such that $\theta(xy) = \theta(y)\theta(x)$ for any $x, y \in \Sigma^*$, and θ^2 is the identity.

[1]A version of this chapter has been accepted for publication.

The WK-complementarity being thus modelled, any word which is a repetition of u and $\theta(u)$ such as uu, $u\theta(u)u$, and $u\theta(u)\theta(u)\theta(u)$ can be regarded repetitive in this sense, and hence, called a θ-power of u. Taking the notion of θ-power into account, the Fine and Wilf's theorem was extended as "given an antimorphic involution θ and words u, v, if a θ-power of u and a θ-power of v have a common prefix of length at least $b(|u|, |v|) = 2|u| + |v| - \gcd(|u|, |v|)$, then u and v are θ-powers of a same word." In this paper, we obtain an improved bound $b'(|u|, |v|) = b(|u|, |v|) - \lfloor \gcd(|u|, |v|)/2 \rfloor$. Then we show all the cases when this bound is optimal by providing all the pairs of words (u, v) such that they are not θ-powers of a same word, but one can construct a θ-power of u and a θ-power of v whose maximal common prefix is of length equal to $b'(|u|, |v|) - 1$. Furthermore, we characterize such words in terms of Sturmian words.

An improved bound for an extension of Fine and Wilf's theorem and its optimality

Lila Kari and Shinnosuke Seki

Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

5.1 Introduction

This paper investigates an extension of Fine and Wilf's theorem in combinatorics of words. Recall that a positive integer p is called a *period* of a word w if the i-th and the $(i+p)$-th letters of w are the same for any $1 \leq i \leq |w| - p$. Fine and Wilf's theorem [12] states that if a word has two periods p, q and is of length at least $p + q - \gcd(p, q)$, then $\gcd(p, q)$ is also its period, where gcd denotes the greatest common divisor. A concise method to prove this result, [5], also proves that the lower bound is "strongly optimal" in the following sense, which was defined in [6], that for *any* pair (p, q) of integers with $p > q > \gcd(p, q)$, one can construct a word of length $p + q - \gcd(p, q) - 1$, with p and q as periods, but without $\gcd(p, q)$ as period (the set of all such words with p and q being coprime is denoted by PER). This theorem has several extensions: e.g., considering more than two periods [3], [4], [6], [13], based on abelian periods [7], for partial or bidimensional words [1], [2], [15].

Changing the focus from integers to words, this theorem can be reformulated as follows: "Given words u, v, if a power of u and a power of v have a common prefix of length at least $|u| + |v| - \gcd(|u|, |v|)$, then u and v are powers of a common word, i.e., they share their primitive root." This result was recently extended in [9], by generalizing the notion of power of a word as inspired by the characteristics of DNA-encoded information. Briefly, a DNA strand can be abstracted as a word over the four-letter alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. Due to the so-called Watson-Crick (WK) complementarity A-T and C-G, two complementary DNA single strands with *opposite* orientations bind to each other to form the structure known as a DNA double strand. WK-complementarity has been modelled mathematically by an antimorphic involution θ, i.e., a function θ such that $\theta(xy) = \theta(y)\theta(x)$ for any $x, y \in \Sigma^*$ (antimorphism), and θ^2 is the identity (involution). An antimorphic involution captures the main features of WK-complementarity, namely that the WK-complement of a DNA single strand is the reverse (antimorphic property) complement (involution property) of the given strand. If we set the antimorphic involution on the four-letter DNA alphabet defined by $\theta(\mathtt{A}) = \mathtt{T}$ and $\theta(\mathtt{C}) = \mathtt{G}$, then for any word $w \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^*$ representing a DNA single strand, the word $\theta(w)$ will represent its WK-complement. For example, using θ, we can calculate the WK-complement of AAC as $\theta(\mathtt{AAC}) = \theta(\mathtt{C})\theta(\mathtt{AA}) = \theta(\mathtt{C})\theta(\mathtt{A})\theta(\mathtt{A}) = \mathtt{GTT}$. We can say that two complementary DNA single strands are equivalent because one can be obtained from the other by θ. Based on this idea, for instance, the strand AACGTTGTT becomes a "power" of AAC because it consists of AAC followed by its WK-complement GTT twice. By using an antimorphic involution θ as a model of the WK-complementarity, a word in $u\{u, \theta(u)\}^*$ is called a *θ-power* of u [9]. With this extended notion of power, the Fine and Wilf's theorem was extended in [9] in the following way: "Given an antimorphic involution θ over an alphabet Σ, and given non-empty words u, v over Σ of lengths p, q with $p > q$, if a θ-power of u and a θ-power of v share a prefix of length at least $b(p,q) = 2p + q - \gcd(p,q)$, then u and v are θ-powers of a common word (in such case, we say that u and v share their *θ-primitive root*)." In [9] some examples of words u, v were provided with the property that such a common prefix of length $b(p,q) - 1$ is too short to force u and v to have the same θ-primitive root. However, these examples do not answer the question of whether $b(p,q)$ is strongly optimal or not, i.e., whether for *any* (p,q), we can find two words u, v of length p, q with different θ-primitive roots such that a θ-power of u and a θ-power of v share a prefix of length $b(p,q) - 1$.

The first contribution of this paper is to give the extended Fine and Wilf's theorem an improved bound $b'(p,q) = b(p,q) - \lfloor \gcd(p,q)/2 \rfloor$ in a constructive manner, which amounts to the negative answer to the above question. Specifically speaking, we design a pair (u, v) of words of lengths p, q with distinct θ-primitive roots in such a manner that one can construct a θ-power of u and a θ-power of v such that their common prefix is as long as possible relative to p and q. We prove that such a common prefix is of length at most $b'(p,q) - 1$, and hence, $b'(p,q)$ becomes the improved bound (Theorem 5.22). We call such a common prefix of length exactly $b'(p,q) - 1$ a *boundary common prefix based on u and v*. Being constructive, our proof simultaneously characterizes the set of all pairs of words with distinct θ-primitive roots based on which one can construct a boundary common prefix. This characterization is the main contribution of this paper. Two corollaries of interest follow: First, there are (infinitely many) pairs of integers (p, q) such that there does not exist any boundary common prefix based on words of respective lengths p, q (Corollary 5.23), and hence, $b'(p, q)$ is not strongly optimal. Second, all the boundary common prefixes are homomorphic images of boundary common prefixes based on some binary words of coprime lengths. This is very similar to the fact that the words which verify the strong optimality of the bound for the Fine and Wilf's theorem are homomorphic images of a (binary) word in PER. de Luca and Mignosi in [11] proved that a word in PER is a finite Sturmian word, or more strongly, the set of all factors of words in PER is equal to the set of all finite Sturmian words. We will show that boundary common prefixes based on words of coprime lengths are also finite Sturmian words, but there exists a finite Sturmian word which never appears as a factor of such boundary common prefixes.

This paper is organized as follows: Section 5.2 introduces basic notions and notation as well as some known results used for our discussion. That is followed by the constructive proof of the improved bound $b'(p, q)$ in Section 5.3 with a few results stating that this bound is not strongly optimal. In Section 5.4, the relationship between boundary common prefixes and finite Sturmian words is discussed. Section 5.5 concludes this paper with some future directions of research.

5.2 Preliminaries

Let Σ be a finite alphabet containing at least two letters. Throughout this paper, elements of Σ (letters) will be denoted by a, b. By Σ^* we denote the set of all finite words over Σ. The empty word is denoted by λ and let $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$. The *length* of a word $w \in \Sigma^*$ is denoted by $|w|$. For a set $X \subseteq \Sigma^+$, $X^* = \{x_1 x_2 \cdots x_n \mid x_i \in X \text{ for all } 1 \leq i \leq n\}$, and $X^+ = X^* \setminus \{\lambda\}$. For a word $w \in \Sigma^*$, a word $x \in \Sigma^*$ is called a *prefix* (*suffix*) of w if $w = xr$ (resp. $w = rx$) for some $r \in \Sigma^*$. Let $\mathrm{Pref}(w)$ and $\mathrm{Suff}(w)$ be the sets of all prefixes of w and of all suffixes of w, respectively. Also let $\mathrm{pref}_n(w)$ denote the prefix of w of length n. If $w = rxt$ for some $r, t \in \Sigma^*$, then x is called an *infix* of w, and if furthermore $r, t \neq \lambda$, x is called a *proper infix* of w. For $x, y \in \Sigma^*$, we denote by $x \wedge y$ the *maximal common prefix* of x and y.

A non-empty word $w \in \Sigma^+$ is said to be *primitive* if it cannot be written as a power of another word; that is, if $w = t^n$, then $n = 1$ and $w = t$. For a non-empty word $w \in \Sigma^+$, the shortest word $t \in \Sigma^+$ such that $w = t^n$ for some $n \geq 1$ is called the *primitive root* of w and is denoted by $\rho(w)$. With respect to the primitive root and the maximal common prefix, there is a result from [5] shown in a form that will be utilized in this paper.

Proposition 5.1 ([5]). *Let $X = \{r, t\} \subseteq \Sigma^+$, $x \in rX^*$, and $y \in tX^*$. If $|x \wedge y| \geq |rt|$, then $\rho(r) = \rho(t)$.*

A mapping $\theta : \Sigma^* \to \Sigma^*$ is called an *antimorphism* if for any words $x, y \in \Sigma^*$,

$\theta(xy) = \theta(y)\theta(x)$; an *involution* if θ^2 is the identity function. Throughout this paper, θ is assumed to be an antimorphic involution on Σ unless otherwise noted explicitly. The mirror image (or *mirror involution*), which maps a word to its reverse, is a typical antimorphic involution. A word $w \in \Sigma^*$ is called a *θ-palindrome* if $w = \theta(w)$, see [10]. The next two lemmas on θ-palindromes play significant roles in this paper.

Lemma 5.2. *For θ-palindromes $x, y \in \Sigma^*$ of the same length d, if $\mathrm{pref}_{\lceil d/2 \rceil}(x) = \mathrm{pref}_{\lceil d/2 \rceil}(y)$, then $x = y$.*

Lemma 5.3. *Let $x, y \in \Sigma^+$ be two θ-palindromes with $d = \gcd(|x|, |y|)$ and $|x|+|y| \geq 3d$. For any integer $i \geq 1$, if $|xy \wedge y^i x| \geq |xy| - 2d$, then $\rho(x) = \rho(y)$.*

Proof. The first case is when $|x| = d$. Due to the hypothesis on $|x| + |y|$, in this case we have $|y| \geq 2d$. Then the overlap between xy and $y^i x$ implies that y begins with x. If $|y| = 2d$, then $y = x^2$ due to $x = \theta(x)$ and $y = \theta(y)$; otherwise ($|y| \geq 3d$), the overlap implies $|xy \wedge y| \geq 2d$, and hence, $x^2 \in \mathrm{Pref}(y)$. Because of $x = \theta(x)$ and $y = \theta(y)$, y has x^2 also as its suffix. Combining these together yields $xy = yx$. Using Proposition 5.1, we get $\rho(x) = \rho(y)$.

The second case is when $|x| \geq 2d$ and $|y| = d$. Let $x_p = \mathrm{pref}_{|x|-d}(x)$. Under this length condition, the overlap between xy and $y^i x$ implies that $x_p \in \mathrm{Pref}(y^i x_p)$. Since the length of x_p is a multiple of d, this means that x_p is a power of y and y is a prefix of x_p, i.e., $y \in \mathrm{Pref}(x)$. This is equal to $y \in \mathrm{Suff}(x)$ and actually now we have that x is a power of y.

The last case is when $|x|, |y| \geq 2d$. In this case, the overlap gives $x \in \mathrm{Pref}(y^i x)$ and $y \in \mathrm{Pref}(xy)$. So, if $|y| \geq |x|$, then the latter prefix relation implies that $x \in \mathrm{Pref}(y)$, which is equivalent to $x \in \mathrm{Suff}(y)$. With $y \in \mathrm{Pref}(xy)$, this implies that $xy = yx$ so that $\rho(x) = \rho(y)$. Conversely, if $|y| < |x|$, then according to $x \in \mathrm{Pref}(y^i x)$, we can let $x = y^j y_p$ for some $j \geq 1$ and $y_p \in \mathrm{Pref}(y)$. Since x and y are θ-palindromes, $x = y^j y_p = \theta(y_p) y^j$ holds. This equality gives $y_p = \theta(y_p)$, and hence, imposes $\rho(y_p) = \rho(y) = \rho(x)$ due to Proposition 5.1. □

In [9], a special class of primitive words was proposed that takes into account the notion of antimorphic involution. For a non-empty word $t \in \Sigma^+$, we call a word in $t\{t, \theta(t)\}^*$ a *θ-power of t*. A non-empty word $w \in \Sigma^+$ is said to be *θ-primitive* if it cannot be written as a θ-power of another word, that is, for $t \in \Sigma^+$, $w \in t\{t, \theta(t)\}^*$ implies $w = t$. The *θ-primitive root* of w, denoted by $\rho_\theta(w)$, is the θ-primitive word t such that $w \in t\{t, \theta(t)\}^*$. The uniqueness of θ-primitive root was proved in [9] using Theorem 5.7 in Section 5.3.

Lemma 5.4 ([9]). *Let $w \in \Sigma^+$ be a θ-primitive word and $w_1, w_2, w_3, w_4 \in \{w, \theta(w)\}$. If $w_1 w_2 x = y w_3 w_4$ holds for some non-empty words $x, y \in \Sigma^+$ with $|x|, |y| < |w|$, then $w_2 \neq w_3$.*

From this lemma, the next theorem easily follows. This is an analogous result to the one stating that a primitive word cannot be a proper infix of its square.

Theorem 5.5 ([14]). *For a θ-primitive word $w \in \Sigma^+$, neither $w\theta(w)$ nor $\theta(w)w$ can be a proper infix of a word in $\{w, \theta(w)\}^3$.*

5.3 An Improved Bound for the Extension of Fine and Wilf's Theorem

Taking the θ-primitivity into account, an extension of the Fine and Wilf's theorem was proposed in [9], of the following two forms:

Theorem 5.6 ([9]). *For $u, v \in \Sigma^+$ with $|u| \geq |v|$, if a θ-power of u and a θ-power of v share a common prefix of length at least $2|u|+|v|-\gcd(|u|,|v|)$, then $\rho_\theta(u) = \rho_\theta(v)$, i.e., there exists a θ-primitive word $t \in \Sigma^+$ such that $u, v \in t\{t, \theta(t)\}^*$.*

Theorem 5.7 ([9]). *For $u, v \in \Sigma^+$, if a θ-power of u and a θ-power of v share a common prefix of length at least $\mathrm{lcm}(|u|,|v|)$, then $\rho_\theta(u) = \rho_\theta(v)$, where $\mathrm{lcm}(|u|,|v|)$ denotes the least common multiple of $|u|$ and $|v|$.*

These theorems give two bounds, and one can be larger than the other depending on the value of $\gcd(|u|,|v|)$ as $\mathrm{lcm}(|u|,|v|) < 2|u| + |v| - \gcd(|u|,|v|)$ if and only if $|v| \leq 2\gcd(|u|,|v|)$. Thus, for integers p, q with $p \geq q$, by letting

$$b(p,q) = \begin{cases} \mathrm{lcm}(p,q) & \text{if } q \leq 2\gcd(p,q); \\ \\ 2p + q - \gcd(p,q) & \text{if } q \geq 3\gcd(p,q), \end{cases} \tag{5.1}$$

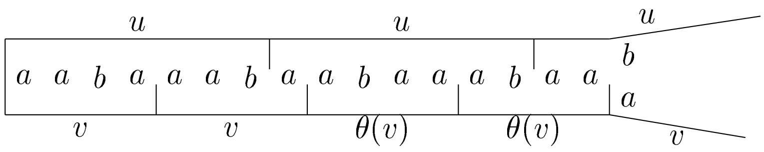

Figure 5.1: Even from words with distinct θ-primitive roots, it is possible to construct θ-powers whose maximal common prefix is shorter by 1 than the bound given in Theorem 5.8.

one can merge Theorems 5.6 and 5.7 into one theorem as follows.

Theorem 5.8. *For $u, v \in \Sigma^+$ with $|u| \geq |v|$, if a θ-power of u and a θ-power of v share a common prefix of length at least $b(|u|, |v|)$, then $\rho_\theta(u) = \rho_\theta(v)$.*

This theorem indicates the possibility of constructing two words u, v with $|u| > |v|$ such that a θ-power of u and a θ-power of v have a common prefix of length $b(|u|, |v|) - 1$, while at the same time $\rho_\theta(u) \neq \rho_\theta(v)$. Here we provide two of such examples, which were introduced in [9].

Example 12. Let $\theta : \{a, b\}^* \to \{a, b\}^*$ be the mirror involution, $u = a^2ba^3b$, and $v = a^2ba$. Then, u^3 and $v^2\theta(v)^2v$ have a common prefix of length $2|u| + |v| - \gcd(|u|, |v|) - 1$, but $\rho_\theta(u) \neq \rho_\theta(v)$. Figure 5.1 is a visualization of this example.

Example 13. Let $\theta : \{a, b\}^* \to \{a, b\}^*$ be the mirror involution, $u = ba^2baba$, and $v = ba^2ba$. Then $u\theta(u)^2$ and v^4 have a common prefix of length $2|u|+|v|-\gcd(|u|, |v|)-1$, but $\rho_\theta(u) \neq \rho_\theta(v)$.

In [6], a sharp distinction was made between a "good" bound and an "optimal" bound. Following this distinction, we define the optimality of a bound in the context of the extended Fine and Wilf's theorem. For a pair of integers (p, q) with $p > q \geq 2\gcd(p,q)$[2], an integer k is called a *good bound for* (p, q) if for any antimorphic involution θ and for any words $u, v \in \Sigma^+$ with $|u| = p$ and $|v| = q$, once there exist a θ-power of u and a θ-power of v which share a prefix of length at least k, one has $\rho_\theta(u) = \rho_\theta(v)$. Based on this, k is an *optimal bound for* (p, q) if it is a good bound for (p, q) whereas $k-1$ is not; i.e., there exist an antimorphic involution θ and words u, v of length p and q with $\rho_\theta(u) \neq \rho_\theta(v)$ from which one can construct a θ-power of u and θ-power of v whose maximal common prefix is of length $k-1$. A bound $b(\cdot,\cdot)$ of two variables is said to be *strongly optimal* if for any (p, q) satisfying the inequality mentioned previously, $b(p, q)$ is optimal. Although the goodness, optimality, and strong optimality are defined here for the extended Fine and Wilf's theorem, these notions can be defined for any variant of this theorem.

Examples 12 and 13 prove the optimality of $b(p, q)$ for (p, q) equal to $(7, 4)$ and $(7, 5)$, respectively. The bound given by the Fine and Wilf's theorem is known to be strongly optimal (see [5]). A question, therefore, arises of whether $b(p, q)$ is *strongly optimal* or not. We will show that $b(p, q)$ is not strongly optimal by proving that $b'(p, q) = b(p, q) - \lfloor \gcd(p, q)/2 \rfloor$ is still a good bound, strictly smaller than $b(p, q)$ unless p and q are coprime.

---

[2]The first inequality can be assumed without loss of generality. The second one is reasonable in the context of Fine and Wilf's theorem because $q = \gcd(p, q)$ means that p is a multiple of q, and hence, the period p is not essential. Whenever we refer to p, q from now on, we implicitly assume that the inequality $p > q \geq 2\gcd(p, q)$ holds.

Unlike the proof of Theorem 5.8 in [9], our proof in the following is constructive. More concretely speaking, we will search for words u and v based on which one can build a *boundary common prefix*. For words $u, v \in \Sigma^+$ with $|u| > |v|$ and $\rho_\theta(u) \neq \rho_\theta(v)$, we call a word $w \in \Sigma^+$ a *boundary common prefix based on u and v* if there exist a θ-power of u and a θ-power of v whose maximal common prefix is w and of length *at least* $b'(|u|, |v|) - 1$. By $\mathrm{BCP}_\theta(u, v)$, we denote the set of all boundary common prefixes based on u and v. Figure 5.1 illustrates a boundary common prefix $a^2ba^3ba^2ba^3ba^2$ based on the specific u and v given in Example 12. What we actually prove in the following is that the length of boundary common prefixes based on u and v is *exactly* $b'(|u|, |v|) - 1$.

As shown in Eq.(5.1), $b(p, q)$ displays different behaviours depending on whether $q \leq 2\gcd(p, q)$ or not, and hence, so does $b'(p, q)$. As such, we will prove that $b'(p, q)$ is good for (p, q) with $p > q = 2\gcd(p, q)$ in Section 5.3.1, and for (p, q) with $p > q \geq 3\gcd(p, q)$ in Section 5.3.2. Note that we do not have to consider any (p, q) with $p > q = \gcd(p, q)$ as mentioned previously. In Section 5.3.3, we will combine these two results together to conclude that $b'(p, q)$ is good for any (p, q).

5.3.1 The case when $q = 2\gcd(p, q)$

Firstly we handle the case $q = 2\gcd(p, q)$ in Proposition 5.9. Its proof will suggest a construction of examples which verify the optimality of the new bound $b'(p, q)$ for any pair of integers (p, q) satisfying this condition.

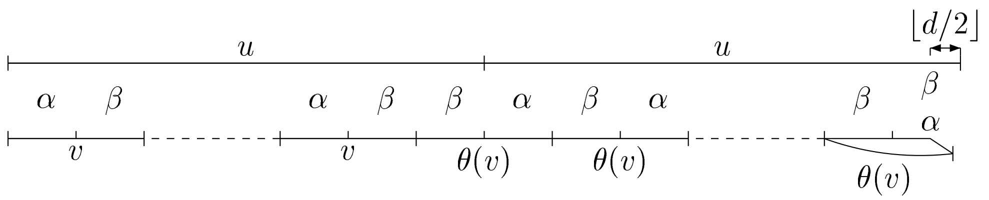

Figure 5.2: Two words u^2 and $v^{(n-1)/2}\theta(v)^{(n-1)/2+1}$ share a prefix of length $2|u| - \lfloor d/2 \rfloor$.

Proposition 5.9. *Let $u, v \in \Sigma^+$ with $|u| > |v|$ and $|v| = 2\gcd(|u|, |v|)$. If a θ-power of u and a θ-power of v share a prefix of length $2|u| - \lfloor \gcd(|u|, |v|)/2 \rfloor$, then $\rho_\theta(u) = \rho_\theta(v)$.*

Proof. Let $d = \gcd(|u|, |v|)$. The length condition on $|u|$ and $|v|$ is equivalent to that $2|u| = n|v|$ holds for some odd integer $n \geq 3$. Let us translate the problem setting as: u_1u_2 and $v_1v_2 \cdots v_n$ agree with each other up to their first $2|u| - \lfloor d/2 \rfloor$ letters, where $u_1 = u$, $u_2 \in \{u, \theta(u)\}$, $v_1 = v$, and $v_2, \ldots, v_n \in \{v, \theta(v)\}$ (see Figure 5.2). One can regard u_1 as a catenation of n words or '*blocks*' $w_1, w_2, \ldots, w_n$ of length d. In the similar fashion, one can let $u_2 = w_{n+1} \cdots w_{2n}$ for some words $w_{n+1}, \ldots, w_{2n}$ of length d. Then $v_i = w_{2i-1}w_{2i}$ holds for any i up to $n-1$. As for v_n, we can let $v_n = w_{2n-1}\mathrm{pref}_{\lceil d/2 \rceil}(w_{2n})x$ for some word x of length $\lfloor d/2 \rfloor$.

It is clear that when u_2 is $\theta(u)$, $v_{(n+1)/2} = w_nw_{n+1}$ becomes a θ-palindrome ($v = \theta(v)$) because it is located at the center of u_1u_2. Hence, $w_{n+1} = \theta(w_n)$, i.e., $v = \theta(v) = w_n\theta(w_n)$, and $u = v^{(n-1)/2}w_n$. These mean that $u, v \in w_n\{w_n, \theta(w_n)\}^*$ so that $\rho_\theta(u) = \rho_\theta(v)$.

Let us consider the other case when u_2 is u. Let $w_1 = \alpha$ and $w_2 = \beta$. Since

$u_2 = u$ begins with $\alpha\beta$, $w_{n+1} = \alpha$ and $w_{n+2} = \beta$. If either $v_{(n+1)/2} = w_n w_{n+1}$ or $v_{(n+1)/2+1} = w_{n+2}w_{n+3}$ is v, then α overlaps with β and results in that $\alpha = \beta$. As a result, $u, v \in \alpha\{\alpha, \theta(\alpha)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. If neither holds, then we obtain $\alpha = \theta(\alpha)$, $\beta = \theta(\beta)$, and $w_n = \beta$; furthermore if $n + 3 \neq 2n$, then $w_{n+3} = \alpha$. According to the same argument, we can figure out that unless $v_2 = \cdots = v_{(n+1)/2-1} = v$ and $v_{(n+1)/2} = \cdots = v_{n-1} = v_n = \theta(v)$, one has $\rho_\theta(u) = \rho_\theta(v)$. In this only one remaining case (illustrated in Figure 5.2), $w_{2n} = \beta$ and $\mathrm{pref}_{\lceil d/2 \rceil}(w_{2n})x = \alpha$. Thus, $\alpha = \beta$ due to Lemma 5.2, and hence, $\rho_\theta(u) = \rho_\theta(v)$. $\square$

This proof clarifies that the only pair of a θ-power of u and a θ-power of v which can share a prefix of length $2|u| - d$ without imposing $\rho_\theta(u) = \rho_\theta(v)$ is $(uu, v^{(n-1)/2}\theta(v)^{(n-1)/2+1})$, where n satisfies $2|u| = n|v|$. Since $2|u| - d \leq 2|u| - \lfloor d/2 \rfloor - 1$, the next result follows from this proof.

Corollary 5.10. $|\mathrm{BCP}_\theta(u, v)| \leq 1$ *for any* $u, v \in \Sigma^+$ *with* $\rho_\theta(u) \neq \rho_\theta(v)$ *and* $|u| > |v| = 2\gcd(|u|, |v|)$.

The proof of Proposition 5.9 and Figure 5.2 hint the possibility that if α and β are θ-palindromes of the same length d which disagree with each other for the first time at their center, i.e., their $\lceil d/2 \rceil$-th letter, then we can reach the new bound minus one while keeping $\rho_\theta(u) \neq \rho_\theta(v)$. For instance, let θ be the mirror involution on $\{a, b\}$, $\alpha = a^d$, and

$$\beta = \begin{cases} a^{\lceil d/2 \rceil - 1} b a^{\lceil d/2 \rceil - 1} & \text{if } d \text{ is odd} \\ \\ a^{\lceil d/2 \rceil - 1} bb a^{\lceil d/2 \rceil - 1} & \text{if } d \text{ is even.} \end{cases} \tag{5.2}$$

For $u = (\alpha\beta)^{(n-1)/2}\beta$ and $v = \alpha\beta$, we have $|u^2 \wedge v^{(n-1)/2}\theta(v)^{(n-1)/2+1}| = 2|u| - \lfloor d/2 \rfloor - 1$. Since v contains at most two occurrences of b and they occur only in the latter half of it, v is θ-primitive. Hence, $\rho_\theta(u) \neq \rho_\theta(v)$.

Theorem 5.11. *$b'(p, q)$ is optimal for any pair (p, q) with $p > q = 2\gcd(p, q)$.*

Besides giving the verification of optimality to $b'(p, q)$, the proof enables us to enumerate all pairs of (u, v) with distinct θ-primitive roots, $|u| > |v| = 2\gcd(|u|, |v|)$, and $\mathrm{BCP}_\theta(u, v)$ is non-empty, i.e., $|\mathrm{BCP}_\theta(u, v)| = 1$ (Corollary 5.10). The way to construct (u, v) from (α, β) being known (see Figure 5.2), it suffices to provide the set of all possible values of (α, β). Note that it is insufficient for (α, β) to be a pair of two distinct θ-palindromes of the same length d and with the same prefix of length $\lceil d/2 \rceil - 1$. For instance, although $\alpha = a\theta(a)$ and $\beta = \theta(a)a$ satisfy these conditions, $u, v \in a\{a, \theta(a)\}^*$, i.e., $\rho_\theta(u) = \rho_\theta(v)$. Excluding these instances leaves the following three candidate sets:

$$\begin{aligned}
T_1 &= \big\{(xa\theta(x), xb\theta(x)) \bigm| x \in \Sigma^*,\ a, b \in \Sigma \text{ such that } a \neq b,\ a = \theta(a),\ b = \theta(b)\big\};\\
T_2 &= \big\{(xa\theta(a)\theta(x), xb\theta(b)\theta(x)) \bigm| x \in \Sigma^*,\ a, b \in \Sigma \text{ such that } a \neq b,\ a \neq \theta(b)\big\};\\
T_3 &= \big\{(xa\theta(a)\theta(x), x\theta(a)a\theta(x)) \bigm| x \in \Sigma^+ \text{ such that } a \neq \theta(a),\ x \notin \{a, \theta(a)\}^+\big\}.
\end{aligned}$$

Actually all of these sets serve our purpose, and hence, in the rest of this paper, we will use α, β only to denote a pair of words in $T_1 \cup T_2 \cup T_3$. In order to see that (α, β) makes the words $u = (\alpha\beta)^{(n-1)/2}\beta$ and $v = \alpha\beta$ have distinct θ-primitive roots, we just have to prove that there does not exist a word t such that $\alpha, \beta \in \{t, \theta(t)\}^+$. This is because $u, v \in \{\alpha, \beta\}^+$ and if $\rho_\theta(u) = \rho_\theta(v)$, then due to $d = \gcd(|u|, |v|)$, $t = \rho_\theta(u)$ is of length at most d, and hence, $\alpha, \beta \in \{t, \theta(t)\}^+$.

Proposition 5.12. *If $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$, then there does not exist $t \in \Sigma^+$ such that $\alpha, \beta \in \{t, \theta(t)\}^+$.*

Proof. Note that $\alpha \neq \beta$. Suppose the existence of such t and let $\alpha = t_1 \cdots t_k$ and $\beta = t'_1 \cdots t'_k$ for some $k \geq 1$ and $t_1, \ldots, t_k, t'_1, \ldots, t'_k \in \{t, \theta(t)\}$. If $(\alpha, \beta) \in T_1$, then the length of α (and β) is odd so that k is odd. Since $\alpha = \theta(\alpha)$, this means that $t = \theta(t)$, and hence, $\alpha = \beta$, which is a contradiction. Even if $(\alpha, \beta) \in T_2 \cup T_3$, an odd k causes the same problem.

Let us consider the case $(\alpha, \beta) \in T_2$ and k is even. Then $t_1 \cdots t_{k/2} = xa$, $t_{k/2+1} \cdots t_k = \theta(a)\theta(x)$, $t'_1 \cdots t'_{k/2} = xb$, and $t'_{k/2+1} \cdots t'_k = \theta(b)\theta(x)$. Hence, for some

$y \in \mathrm{Suff}(x)$, $t_{k/2} = ya$, $t_{k/2+1} = \theta(a)\theta(y)$, $t'_{k/2} = yb$, and $t'_{k/2+1} = \theta(b)\theta(y)$. Since $a \neq b$, both $t_{k/2} \neq t'_{k/2}$ and $t_{k/2+1} \neq t'_{k/2+1}$ must hold. These four words are either t or $\theta(t)$ so that we have either $ya = \theta(a)\theta(y)$ and $yb = \theta(b)\theta(y)$ or $ya = \theta(b)\theta(y)$ and $yb = \theta(a)\theta(y)$. In the latter case, if y is empty, then $a = \theta(b)$; otherwise, these two equations imply that y begins with $\theta(b)$ and with $\theta(a)$ so that $\theta(b) = \theta(a)$; both contradict the condition on a, b in T_2. Even in the former case, unless y is empty, we reach this contradiction along the same argument. If y is empty, then $a = \theta(a)$, $b = \theta(b)$, and one of these has to be t and the other has to be $\theta(t)$. This is, however, impossible because a is assumed to be neither b nor $\theta(b)$.

The same but simpler argument works for $(\alpha, \beta) \in T_3$. Note that along this argument y should be non-empty because otherwise $t_{k/2} = a$, and hence, $x = t_1 \cdots t_{k/2-1} \in \{a, \theta(a)\}^+$, which is against the definition of T_3. □

Theorem 5.13. *Let $u, v \in \Sigma^+$ with $\rho_\theta(u) \neq \rho_\theta(v)$ and $|u| > |v| = 2\gcd(|u|, |v|)$. Then $\mathrm{BCP}_\theta(u, v) \neq \emptyset$ if and only if $u = (\alpha\beta)^{(n-1)/2}\beta$ and $v = \alpha\beta$ for some odd integer $n \geq 3$ and $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$.*

In the next subsection, we will prove that even when this length condition $|u| > |v| = 2\gcd(|u|, |v|)$ does not hold, the existence of boundary common prefix requires u and v to be described by two distinct θ-palindromes α, β of length d taken from T_1, T_2, or T_3, and hence, these three sets will completely characterize the boundary common prefixes.

5.3.2 The case when $q \geq 3\gcd(p, q)$

The proof of our improved bound $b'(p, q)$ continues here for (p, q) with $p > q \geq 3\gcd(p, q)$. Under this length condition, by definition, $b'(p, q) = 2p + q - \gcd(p, q) - \lfloor \gcd(p, q)/2 \rfloor$. Unlike the case considered in the previous subsection, this bound shall turn out not to be optimal for some such (p, q). A constructive way to find the optimal bound is to build an antimorphic involution θ and words u and v with distinct θ-primitive roots and $|u| > |v| \geq 3\gcd(|u|, |v|)$ such that the maximal common prefix between a θ-power of u and a θ-power of v gets as long as possible relative to $|u|$ and $|v|$. This informal description allows us to assume that u and v are θ-primitive, though in formal problem settings the validity of this assumption has to be verified (see Lemma 5.15.)

First of all, we briefly mention how small the optimal bound for (p, q) can be relative to p and q. The following parameterized example proves that $2p$ is not a good bound for any such pair (p, q), that is, the optimal bound has to be bigger than $2p$.

Example 14. Let θ be the mirror involution on $\{a, b\}$. For a given (p, q) with $p > q \geq 3\gcd(p, q)$, let $v = a^{2p \pmod q} b^{-2p \pmod q}$ and $u\theta(u) = v^{\lfloor 2p/q \rfloor} a^{2p \pmod q}$. Then $|u\theta(u)u_3 \wedge v^{\lceil 2p/q \rceil}| = 2p$ regardless of the value of $u_3 \in \{u, \theta(u)\}$ because both u and $\theta(u)$ begin with a. In addition, $\rho_\theta(u) \neq \rho_\theta(v)$.

As a digression, this example can be easily modified to show that $2p + \lceil \gcd(p, q)/2 \rceil -$

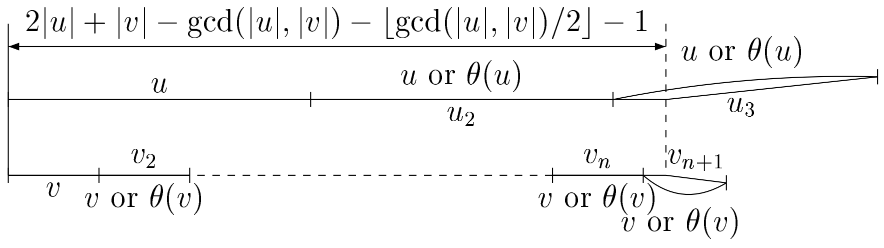

Figure 5.3: A boundary common prefix based on u and v. This shows how uu_2u_3 and $vv_2 \cdots v_n v_{n+1}$ overlap with each other when Condition (5.3) is satisfied.

1 is not a good bound for any such (p, q), either. Since $-2p \pmod q$ is a multiple of $\gcd(p, q)$, we can say that the suffix $b^{-2p \pmod q}$ of v consists of $\frac{-2p \pmod q}{\gcd(p,q)}$ blocks b^d. Replacing each of these blocks with β given in Eq. (5.2) verifies this point.

To return to our point, u and v are to be constructed so as for such a maximal common prefix to be of length at least $2|u|$ in light of Example 14. Hence, the common prefix is formalized with an integer n satisfying $(n-1)|v| < 2|u| < n|v|$ and words $u_1, u_2, u_3 \in \{u, \theta(u)\}$ and $v_1, \ldots, v_n, v_{n+1} \in \{v, \theta(v)\}$ as $u_1u_2u_3 \wedge v_1 \cdots v_n v_{n+1}$ with the following condition:

$$|u_1u_2u_3 \wedge v_1 \cdots v_n v_{n+1}| \geq 2|u| + k \text{ for some } k \geq 0 \tag{5.3}$$

Note that u_1 and v_1 are to be fixed to u and v without loss of generality. Figure 5.3 illustrates the maximal common prefix between a θ-power of u and a θ-power of v satisfying Condition (5.3) with $k = |v| - \gcd(|u|, |v|) - \lfloor \gcd(|u|, |v|)/2 \rfloor - 1$.

Lemma 5.14. *Let u, v be distinct θ-primitive words with $|u| > |v| \geq 3 \gcd(|u|, |v|)$.*

If there exist an integer n satisfying $(n-1)|v| < 2|u| < n|v|$, and words $u_1 = u$, $u_2, u_3 \in \{u, \theta(u)\}$, $v_1 = v$, $v_2, \ldots, v_n, v_{n+1} \in \{v, \theta(v)\}$ satisfying Condition (5.3), then one of the following two cases holds:

1. *$u_2 = \theta(u)$ and $v_1 = \cdots = v_{n-1} = v$;*

2. *$u_2 = u$, $v_1 = \cdots = v_{\lceil n/2 \rceil - 1} = v$, and $v_{\lceil n/2 \rceil + 1} = \cdots = v_{n-1} = \theta(v)$.*

Proof. Let us consider the case when $u_2 = \theta(u)$ first. In this case, we have $u\theta(u) = v_1 \cdots v_{n-1}w$, where w is a non-empty prefix of v_n. Since $u\theta(u)$ is a θ-palindrome, $v_1 \cdots v_{n-1}w = \theta(w)\theta(v_{n-1}) \cdots \theta(v_1)$ holds. This means that $\theta(v_{n-1}) \cdots \theta(v_1)$ is a proper infix of $v_1 \cdots v_n$. Then we can apply Theorem 5.5 to conclude $\theta(v_{n-1}) = \cdots = \theta(v_1)$ because v is assumed to be θ-primitive.

Even for the second case when $u_2 = u$, the basic strategy is the same. Since the border between u_1 and u_2 is located on $v_{\lceil n/2 \rceil}$, one can let $v_{\lceil n/2 \rceil} = xy$ for some non-empty words x, y such that $u_1 = v_1 \cdots v_{\lceil n/2 \rceil - 1}x$ and $u_2 = yv_{\lceil n/2 \rceil + 1} \cdots v_{n-1}z$, where z is a non-empty prefix of v_n. Then we have $v_1 \cdots v_{\lceil n/2 \rceil - 1}x = yv_{\lceil n/2 \rceil + 1} \cdots v_{n-1}z$ because $u_1 = u_2$. This equation implies that $v_1 \cdots v_{\lceil n/2 \rceil - 1}$ is a proper infix of $v_{\lceil n/2 \rceil} \cdots v_n$ so that $v_1 = \cdots = v_{\lceil n/2 \rceil - 1} = v$. If $n \geq 4$, we can also determine the values of $v_{\lceil n/2 \rceil + 1}, \ldots, v_{n-1}$. Firstly, the value of $v_{\lceil n/2 \rceil + 1}$ is determined to be $\theta(v)$ by applying Lemma 5.4 to the overlap between v_1v_2 and $v_{\lceil n/2 \rceil}v_{\lceil n/2 \rceil + 1}$. When $n \geq 6$, Theorem 5.5 is applied to that $v_{\lceil n/2 \rceil + 1} \cdots v_{n-1}$ being a proper infix of $v_1 \cdots v_{\lceil n/2 \rceil}$ to fix $v_{\lceil n/2 \rceil + 1} = \cdots = v_{n-1} = \theta(v)$. $\square$

As suggested previously, an element of $\mathrm{BCP}_\theta(u,v)$ is characterized by Condition (5.3) with $k = |v| - \gcd(|u|,|v|) - \lfloor \gcd(|u|,|v|)/2 \rfloor - 1$. Thus, this condition is replaced by the next condition:

$$|u_1u_2u_3 \wedge v_1 \cdots v_n v_{n+1}| \geq 2|u| + |v| - \gcd(|u|,|v|) - \left\lfloor \frac{\gcd(|u|,|v|)}{2} \right\rfloor - 1. \qquad (5.4)$$

Once this inequality proves not to hold strictly, $b'(p,q)$ becomes a good bound for an arbitrary pair (p,q). The next lemma verifies that the assumption of u,v being θ-primitive is valid when we consider $\mathrm{BCP}_\theta(u,v)$.

Lemma 5.15. *Let $u,v \in \Sigma^+$ such that $\rho_\theta(u) \neq \rho_\theta(v)$ and $|u| > |v| \geq 3\gcd(|u|,|v|)$. Unless both u and v are θ-primitive,* $\mathrm{BCP}_\theta(u,v) = \emptyset$.

Proof. Here we prove its contrapositive: if $\mathrm{BCP}_\theta(u,v) \neq \emptyset$, then both u and v are θ-primitive. For this purpose, suppose the non-emptiness and that u and v were not θ-primitive at the same time, and see that a contradiction is unavoidable. Let $r = \rho_\theta(u)$, $t = \rho_\theta(v)$, $d = \gcd(|u|,|v|)$, and $d' = \gcd(|r|,|t|)$. It suffices to show that $b(\max(|r|,|t|), \min(|r|,|t|)) \leq b'(|u|,|v|) - 1$ under this supposition, which would lead us to the contradictive conclusion $\rho_\theta(u) = \rho_\theta(v)$ due to Theorem 5.8.

If $\min(|r|,|t|) \leq 2d'$, then by definition $b(\max(|r|,|t|), \min(|r|,|t|)) = \mathrm{lcm}(|r|,|t|)$, and we have $\mathrm{lcm}(|r|,|t|) \leq 2\max(|r|,|t|) \leq 2|u| \leq b'(|u|,|v|) - 1$.

In the case $\min(|r|, |t|) \geq 3d'$, we claim that

$$2\max(|r|, |t|) + \min(|r|, |t|) - d' \leq b'(|u|, |v|) - 1 \tag{5.5}$$

holds if $r \neq u$ or $t \neq v$. To prove this claim, it is worth noting that $|t| \leq |u| - d$ holds because $|t| \leq |v|$ and $|v| \leq |u| - d$. Firstly, let us consider the case $|r| > |t|$. If $r \neq u$, then one easily obtains $2|r| + |t| - d' < 2|u| < b'(|u|, |v|) - 1$ because $r \neq u$ means $2|r| \leq |u|$. This inequality is exactly same as (5.5) when $|r| > |t|$. If $r = u$, then $t \neq v$ so that $|t| \leq |v|/2 \leq |v| - d - \lfloor d/2 \rfloor$ holds; the latter inequality follows from $|v| \geq 3d$. Thus, $|t| - d' \leq |t| - 1 \leq |v| - d - \lfloor d/2 \rfloor - 1$, and hence, we have $2|r| + |t| - d' \leq b'(|u|, |v|) - 1$. Conversely if $|r| < |t|$, then $|r| < |v|$ holds. Due to the inequality:

$$2|u| + |v| - 2d \leq 2|u| + |v| - d - \lfloor d/2 \rfloor - 1, \tag{5.6}$$

we have $2|t| + |r| - d' \leq 2(|u| - d) + |v| - d' \leq 2|u| + |v| - 2d \leq b'(|u|, |v|) - 1$. This is the same as (5.5) when $|r| < |t|$. Having proved the claim, now it suffices to note that the left-hand side of (5.5) is equivalent to $b(\max(|r|, |t|), \min(|r|, |t|))$. $\square$

Note that the inequality given in Eq. (5.6) will play a significant role throughout this paper.

Up to now, we have seen that the combinations of the values of $u_2, u_3, v_2, \ldots, v_{n+1}$ have been already severely-limited under the condition (5.3) due to Lemma 5.14.

We will see in the following that some specific value of k in this condition further restricts the number of possible combinations.

Proposition 5.16 ([8]). *Let $u, v \in \Sigma^+$ such that v is θ-primitive, $u_2, u_3 \in \{u, \theta(u)\}$, and $v_2, \ldots, v_n \in \{v, \theta(v)\}$ for some integer $n \geq 3$. If $vv_2 \cdots v_n$ is a prefix of uu_2u_3 and $(n-1)|v| < 2|u| < n|v|$, then there are only two cases possible:*

1. *$u_2 = \theta(u)$ and $v_2 = \cdots = v_n = v$ with $u\theta(u) = (yx)^{n-1}y$ and $v = yx$ for some non-empty θ-palindromes x, y; and*

2. *$u_2 = u$, n is even, $v_2 = \cdots = v_{n/2} = v$, and $v_{n/2+1} = \cdots = v_n = \theta(v)$ with $v = r(tr)^i(rt)^{i+j}r$ and $u = v^{n/2-1}r(tr)^i(rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t.*

This proposition is applicable to our problem when $v_1 \cdots v_n$ is a prefix of $u_1u_2u_3$, that is, when the border between v_n and v_{n+1} is at the left of the vertical dashed line in Figure 5.3. Since $2|u| - (n-1)|v|$ is a multiple of $\gcd(|u|, |v|)$, this condition is formalized as $2|u| - (n-1)|v| \geq 2\gcd(|u|, |v|)$. This always holds when n is odd because $|u| - \frac{n-1}{2}|v|$ is a multiple of $\gcd(|u|, |v|)$. On the contrary, $2|u| - (n-1)|v| = \gcd(|u|, |v|)$ may hold when n is even. Then $v_1 \cdots v_n$ disagrees with $u_1u_2u_3$ somewhere within the $(\lfloor \gcd(|u|, |v|)/2 \rfloor + 1)$ rightmost letters of v_n, as shown in the next example.

Example 15. Let $u = abbab$, $v = abb$, and θ be the mirror image on $\{a, b\}$. Then $u\theta(u)^2$ and v^4 satisfy Condition (5.4), with $n = 4$, and $2|u| - (n-1)|v| = \gcd(|u|, |v|)$.

Lemma 5.17 ([8]). *Let $v \in \Sigma^+$ be a θ-primitive word, and $x, y \in \Sigma^+$ be words strictly shorter than v. For an integer $k \geq 1$, the solution to $v\theta(v)^k x = yv^{k+1}$ is characterized as $v = r(tr)^i(rt)^{i+j}r$, $y = r(tr)^i(rt)^j$, and $x = (rt)^{i+j}r$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t.*

Lemma 5.18. *Let u, v be distinct θ-primitive words with $|u| > |v| \geq 3\gcd(|u|, |v|)$. If there exist an integer n, and words $u_1 = u$, $u_2, u_3 \in \{u, \theta(u)\}$, $v_1 = v$, $v_2, \ldots, v_n, v_{n+1} \in \{v, \theta(v)\}$ satisfying Condition (5.4) and $2|u| - (n-1)|v| = \gcd(|u|, |v|)$, then $u_2 = \theta(u)$ and $v_2 = \cdots = v_n = v$. Moreover, $v = yx$ and $u\theta(u) = (yx)^{n-1}y$ for some θ-palindromes $y, x \in \Sigma^+$.*

Proof. Let $d = \gcd(|u|, |v|)$. As mentioned previously, in order for $2|u| - (n-1)|v| = d$ to hold, n has to be even. So, let $n = 2(m+1)$ for some $m \geq 1$.

Firstly, we investigate the case when $u_2 = \theta(u)$. In this case, Lemma 5.14 fixes all of $v_2, \ldots, v_{2m-1}$ to be equal to v_1, i.e., v. Then we can let

$$u_1u_2 = u\theta(u) = v^{2m+1}y \tag{5.7}$$

for some $y \in \mathrm{Pref}(v_{2m+2})$. From Eq. (5.7) and the hypothesis of this lemma, $|y| = 2|u| - (2m+1)|v| = d$. Combining this relation and Condition (5.4) implies that yu_3 and v_{2m+2} share their prefix of length at least $|v| - d$. At any rate, since $u\theta(u)$ is a θ-palindrome, Eq. (5.7) gives $v^{2m+1}y = \theta(y)\theta(v)^{2m+1}$. This means that $vy \in \mathrm{Suff}(\theta(v)^2)$ because $m \geq 1$, and this suffix condition allows us to let $\theta(v) = xy$

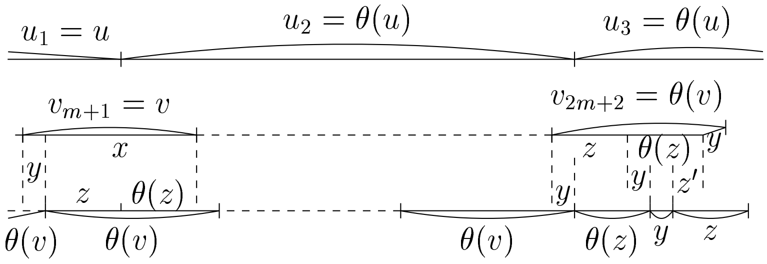

Figure 5.4: When $u_3 = \theta(u)$ and $|y| = d$, $v_{2m+2} = \theta(v)$ and the prefix $\theta(z)yz$ of $\theta(u)$ partially overlap as shown here.

for some $x \in \Sigma^+$. Substituting this back to the suffix condition results in $\theta(y)\theta(x)y \in \mathrm{Suff}(xyxy)$. From this, we can easily observe that $y = \theta(y)$ and $x = \theta(x)$. Now we have $v = yx$. Note that the relation $(2m+2)|v| - 2|u| = |x|$ results from this equation and Eq. (5.7) so that x is a θ-palindrome of even length; that is, we can let $x = z\theta(z)$ for some $z \in \Sigma^+$. Hence, $v = yz\theta(z)$, and by substituting this into Eq. (5.7), we can obtain that

$$u = v^m yz = (yx)^m yz = (yz\theta(z))^m yz. \tag{5.8}$$

According to this equation, the Euclidean algorithm gives $d = \gcd(|u|, |v|) = \gcd(|v|, |y| + |z|) = \gcd(|y| + |z|, |z|) = \gcd(|z|, |y|)$. This equation further implies $\gcd(|x|, |y|) = \gcd(2|z|, |y|) = d$ because $|y| = d$.

From now on, we will prove that v_{2m+2} has to be v. Suppose not, that is, $v_{2m+2} = \theta(v) = xy$. Recall that yu_3 and v_{2m+2} share their prefix of length at least $|x|+|y|-d$. In addition, $|x|+|y|-d \geq 2d$ according to the hypothesis $|v| \geq 3d$. Thus, if $|z| = d$, then this common prefix immediately gives $z = y$, and hence, v would be

y^3. This contradicts the θ-primitivity of v so that z has to be of length at least $2d$. If $u_3 = u$, then $yu_3 = y(yx)^m yz$ holds due to Eq. (5.8), and hence, this common prefix implies that yyx and xy share their prefix of length $|x| + |y| - d$. As seen above, $\gcd(|x|, |y|) = d$ and $|v| = |yx| \geq 3d$. Thus, Lemma 5.3 is applicable to this common prefix, and results in $\rho(x) = \rho(y)$. This, however, contradicts $\rho_\theta(u) \neq \rho_\theta(v)$, and hence, u_3 has to be $\theta(u) = \theta(z)y(z\theta(z)y)^m$. See Figure 5.4. In this case, the common prefix between v_{2m+2} and yu_3 gives

$$z\theta(z) = y\theta(z)yz' \tag{5.9}$$

for some $z' \in \mathrm{Pref}(z)$. Eq. (5.9) implies that $z' \in \mathrm{Suff}(\theta(z))$, i.e., $\theta(z') \in \mathrm{Pref}(z)$, so that $z' = \theta(z')$. Eq. (5.9) also enables us to let $z = yz''$ for some $z'' \in \Sigma^*$. Substituting $\theta(z) = \theta(z'')y$ back into Eq. (5.9) yields $z\theta(z) = y\theta(z'')y^2z'$, and hence, $\theta(z) = y^2z'$, i.e., $z = z'y^2$. If $|z| = 2d$, then $z' = \lambda$; otherwise, by replacing Eq. (5.9) by these, we can obtain $z'y^2y^2z' = y^3z'yz'$, and hence, $z'y^3 = y^3z'$, which implies $\rho(z') = \rho(y)$. However, in both cases, we reach the contradiction with the θ-primitivity of $v = yz'y^2y^2z'$.

Now we have to prove that u_2 cannot be u. Suppose for the sake of contradiction that $u_2 = u$. Lemma 5.14 gives $v_1 = \cdots = v_m = v$ and $v_{m+2} = \cdots = v_{2m+1} = \theta(v)$. So,

$$u = v^m z = x\theta(v)^m z' \tag{5.10}$$

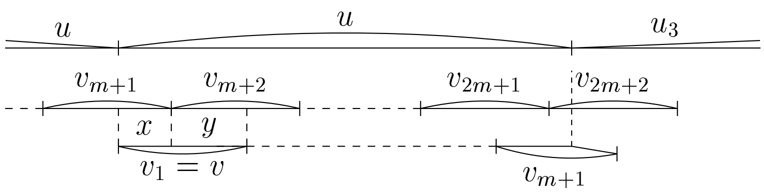

Figure 5.5: If $u_2 = u$, then $v_1 v_2 \dots v_{m+1}$ overlaps with $v_{m+1} \dots v_{2m+1}$ not depending on the value of u_3.

for some $z, x, z' \in \Sigma^+$ such that $v_{m+1} = zx$. Since $m \geq 1$, this allows us to let $v = xy$ for some $y \in \mathrm{Pref}(\theta(v))$ (see Figure 5.5). Substituting this into Eq. (5.10) gives $y = \theta(y)$. Eq. (5.10) also gives $|x| = (m+1)|v| - |u|$, and by combining this with the hypothesis $(2m+1)|v| < 2|u| < (2m+2)|v|$, we obtain $2|x| = (2m+2)|v| - 2|u| < |v| = |x| + |y|$, and hence, $|x| < |y|$. As done before, based on $u = v^m z$ and $|z| = |y|$, the Euclidean algorithm gives $d = \gcd(|u|, |v|) = \gcd(|x|, |y|)$. Thus, $|x| < |y|$ implies that $|y| \geq 2d$. With Eq. (5.10), this length condition results in

$$(m+1)|v| = |u| + |x| \leq |u| + |x| + |y| - 2d. \tag{5.11}$$

For our purpose, it suffices to prove that v_{m+1} can be neither v or $\theta(v)$. Suppose first that $v_{m+1} = \theta(v) = yx$. First of all, we can easily see that $z = y$ because v_{m+1} was let to be zx. Thus, Eq. (5.10) can be rewritten as $u = v^m y$. As illustrated in Figure 5.5, the overlap between $v_{2m+1} = \theta(v)$ and $v_{m+1} = \theta(v)$ implies that $x \in \mathrm{Pref}(\theta(v))$. This implies $x \in \mathrm{Pref}(y)$, i.e., $\theta(x) \in \mathrm{Suff}(y)$, because $\theta(v) = yx$ and $|x| < |y|$. As a result, $\theta(x) \in \mathrm{Suff}(u)$, and hence, x is a prefix of both u and $\theta(u)$.

This means that $v^m\theta(v) \in \mathrm{Pref}(uu_3)$ regardless of whether u_3 is u or $\theta(u)$. Note that, by the hypothesis, $u_2u_3 = uu_3$ and $xv_{m+2}\cdots v_{2m+1}v_{2m+2}$ share their prefix of length at least $|u|+|v|-2d$. Due to Condition (5.11), $v^m\theta(v) \in \mathrm{Pref}(xv_{m+2}\cdots v_{2m+2})$, that is, $v^m\theta(v)$ is an infix of $v_{m+1}\cdots v_{2m+2}$. However, since $m \geq 1$ and v is primitive, this contradicts Theorem 5.5. Thus, v_{m+1} cannot be $\theta(v)$ so that has to be v. If so, applying Lemma 5.17 to the overlap between $v_1\cdots v_{m+1}$ and $v_{m+1}\cdots v_{2m+1}$ yields $v = r(tr)^i(rt)^{i+j}r$ and $u = v^m r(tr)^i(rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t. One can easily check that $u^2 = v^{m+1}\theta(v)^m(rt)^j$ holds, and hence, $|(rt)^j| = d$ due to $2|u| - (2m+1)|v| = d$. On the contrary, from $\gcd(|u|,|v|) = d$, the Euclidean algorithm derives $\gcd(|r(tr)^i|, |(rt)^j|) = d$. However, $|r(tr)^i|$ cannot be a multiple of $|(rt)^j| = d$ because $r, t \neq \lambda$. $\square$

Lemma 5.19. *Let u, v be distinct θ-primitive words with $|u| > |v| \geq 3\gcd(|u|,|v|)$. If there exist an integer n, words $u_1, u_2, u_3 \in \{u, \theta(u)\}$, and $v_1, \ldots, v_n, v_{n+1} \in \{v, \theta(v)\}$ satisfying Condition* (5.4)*, then $u_2 = u_3$.*

Proof. Let $d = \gcd(|u|,|v|)$. We will consider two cases depending on whether u_2 is $\theta(u)$ or u, and will prove that u_3 has to be equal to u_2.

The first case is when $u_2 = \theta(u)$. In this case, Proposition 5.16 and Lemma 5.18 imply that $v_2 = \cdots = v_n = v$, $v = yx$ and $u\theta(u) = (yx)^{n-1}y$ for some non-empty θ-palindromes x, y. The Euclidean algorithm yields $\gcd(2|u|,|v|) = \gcd(|y|,|v|) = \gcd(|x|,|y|)$, and hence, $\gcd(|x|,|y|)$ is either d or $2d$ because $d = \gcd(|u|,|v|)$. Suppose $u_3 = u$. This means that u_3 starts with yx, and hence, yx and xv_{n+1} share their prefix of length at least $|x| + |y| - d - \lfloor d/2 \rfloor - 1$. If $|x| + |y| = 2\gcd(|x|, |y|)$, then $|x| = |y| = d$ or $|x| = |y| = 2d$, but indeed only the latter is valid under the assumption $|v| = |x| + |y| \geq 3d$. This means that the common prefix is of length at least $|x|$ so that it implies $x = y$, which however contradicts the θ-primitivity of v. Conversely, if $|x|+|y| \geq 3\gcd(|x|, |y|)$, then $|x|+|y|-d-\lfloor d/2 \rfloor -1 \geq |x|+|y|-2d \geq |x| + |y| - 2\gcd(|x|, |y|)$ (here $d \leq \gcd(|x|, |y|)$ is used). Since v_{n+1} is either $v = yx$ or $\theta(v) = xy$, Lemma 5.3 is applicable to the common prefix to obtain $\rho(x) = \rho(y)$. Now that we have reached the same contradiction, we can conclude that the only possible choice of u_3 is $\theta(u)$.

The next case is when $u_2 = u$. Due to Proposition 5.16 and Lemma 5.18, n is even ($n = 2m + 2$ for some $m \geq 1$), $v_2 = \cdots = v_{m+1} = v$ and $v_{m+2} = \cdots = v_{2m+2} = \theta(v)$, with $v = r(tr)^i(rt)^{i+j}r$ and $u = v^m r(tr)^i(rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t. Then we have $v^{m+1}\theta(v)^{m+1} = uu(rt)^i r(rt)^i r$. The Euclidean algorithm derives $\gcd(|(rt)^i r|, |(tr)^j|) = d$ from $\gcd(|u|, |v|) = d$. Note that $|u_3 \wedge (rt)^i r(rt)^i r v_{2m+3}| = |v| - d - \lfloor d/2 \rfloor - 1 \geq |v| - 2d \geq |v| - 2|r(tr)^i| = |(tr)^j| \geq |rt|$. Consequently u_3 must not begin with t in light of Proposition 5.1, and hence, u_3 cannot be $\theta(u)$. □

Now we are ready to prove that $b'(p, q)$ is an improved bound for the extended Fine and Wilf's theorem. Recall Eq. (5.6), which makes it possible to distinguish the cases in which boundary common prefixes are constructable.

Theorem 5.20. *Let $u, v \in \Sigma^+$ with $\rho_\theta(u) \neq \rho_\theta(v)$ and $|u| > |v| \geq 3\gcd(|u|,|v|)$. Then the length of a word in* $\mathrm{BCP}_\theta(u,v)$ *is* $2|u|+|v|-\gcd(|u|,|v|)-\lfloor \gcd(|u|,|v|)/2 \rfloor - 1$. *Moreover,* $\mathrm{BCP}_\theta(u,v) \neq \emptyset$ *if and only if one of the following two cases holds: for some $m \geq 1$, $i \geq 0$, and $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$ and*

1. $u = (\alpha\beta(\beta\alpha)^i\beta)^m\alpha\beta$, $v = \alpha\beta(\beta\alpha)^i\beta$;

2. $u = [\alpha(\beta\alpha)^i(\alpha\beta)^{i+1}\alpha]^m\alpha(\beta\alpha)^i\alpha\beta$, $v = \alpha(\beta\alpha)^i(\alpha\beta)^{i+1}\alpha$.

Proof. Let $d = \gcd(|u|,|v|)$ and assume that $\mathrm{BCP}_\theta(u,v)$ is not empty. Then Lemma 5.15 implies that both u and v are θ-primitive. Since an element of $\mathrm{BCP}_\theta(u,v)$ is characterized by Condition (5.4), Proposition 5.16, Lemmas 5.18 and 5.19 leave only two cases to be investigated:

1. $u_2 = u_3 = \theta(u)$, $v_2 = \cdots = v_n = v$, $v = yx$, and $u\theta(u) = v^{n-1}y$ for some non-empty distinct θ-palindromes x, y; and

2. $u_2 = u_3 = u$, $n = 2m+2$ for some $m \geq 1$, $v_2 = \cdots = v_{m+1} = v$, $v_{m+2} = \cdots = v_{2m+2} = \theta(v)$, $v = r(tr)^i(rt)^{i+j}r$, and $u = v^m r(tr)^i(rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty distinct θ-palindromes r, t.

Case 1: In this case, the parity of n matters so that we first consider the subcase when n is odd (see Figure 5.6). Then the border between u_1 and u_2 splits the prefix y of $v_{(n+1)/2}$ into half. Hence, we can let $y = z'\theta(z')$ for some $z' \in \Sigma^+$ and $u =$

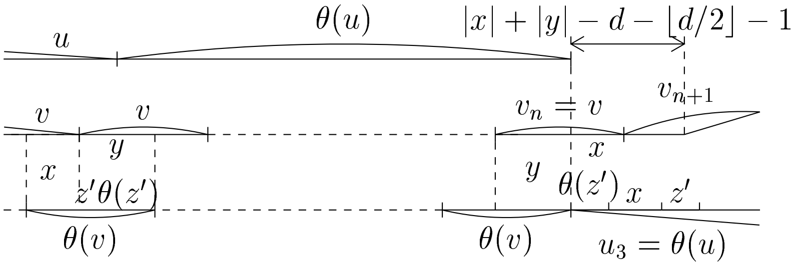

Figure 5.6: For an odd n, $u\theta(u)^2$ and $v^n v_{n+1}$ share the common prefix of length $2|u| + |v| - d - \lfloor d/2 \rfloor - 1$, where $d = \gcd(|u|, |v|)$.

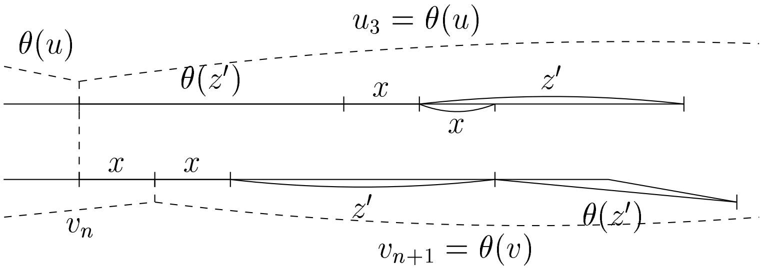

Figure 5.7: When n is odd and $|x| < |z'|$, u_2u_3 and v_nv_{n+1} overlap as shown here.

$v^{(n-1)/2}z'$. The Euclidean algorithm derives $\gcd(|z'|, |x|) = d$ from $\gcd(|u|, |v|) = d$. Focus to the right of border between u_2 and u_3. The rightmost dashed line in Figure 5.6, up to which $u_1u_2u_3$ agree with $v_1v_2 \cdots v_{n+1}$, is located on v_{n+1} because $|y| = 2|z'| \geq 2d$. Thus, the suffix x of v_n is a prefix of the prefix $\theta(z')x$ of u_3. So, if z' were of length d, then due to this prefix relation and $d = \gcd(|z'|, |x|)$, x would be a power of $\theta(z')$, which contradicts the θ-primitivity of v. Therefore, z' has to be of length at least $2d$. This means that the rightmost vertical dashed line in Figure 5.6 is on z' of the prefix $\theta(z')xz'$ of $u_3 = \theta(u)$, and hence, $\theta(z')x \in \mathrm{Pref}(xv_{n+1})$.

In what follows, we prove that in this subcase $\mathrm{BCP}_\theta(u, v) \neq \emptyset$ requires $v_{n+1} = v$ and $|z'| = 2d$. For the sake of contradiction, suppose that v_{n+1} were $\theta(v) = xz'\theta(z')$.

Then the prefix relation just mentioned is written as

$$\theta(z')x \in \mathrm{Pref}(xxz'\theta(z')). \tag{5.12}$$

First of all, $|z'| > |x|$ has to hold because otherwise Relation (5.12) would cause $\theta(z')x \in \mathrm{Pref}(x^2)$, that is, $\rho(z') = \rho(x)$ due to Proposition 5.1, which contradicts the θ-primitivity of v. With this condition, Relation (5.12) gives $x^2 \in \mathrm{Pref}(\theta(z')x)$. If $|z'| = 2d$ (i.e. $|x| = d$), then this prefix relation would result in $\theta(z') = x^2$ and lead us to the same contradiction. Otherwise ($|z'| \geq 3d$), as illustrated in Figure 5.7, $\mathrm{pref}_{|z'|-2d}(z') \in \mathrm{Pref}(x\theta(z'))$. Substituting this into the overlap between $\theta(z')x$ and xxz' implies either $\theta(z') \in \mathrm{Pref}(x^3\theta(z'))$ if $|x| = d$; or $\theta(z')x \in \mathrm{Pref}(x^3\theta(z'))$ otherwise. In the former case, $\theta(z')$ would be a power of x, whereas in the latter case Proposition 5.1 would imply $\rho(\theta(z')) = \rho(x)$. At any rate, we face the contradiction against the θ-primitivity of v. Consequently, v_{n+1} has to be v. Then the prefix relation $\theta(z')x \in \mathrm{Pref}(xv_{n+1})$ is rather equal to $\theta(z')x \in \mathrm{Pref}(xz'\theta(z')x)$, and we can immediately see that $\theta(z')x = xz'$. This is a well-known conjugacy equation and can be solved as

$$\theta(z') = rt, x = r(tr)^k, z' = tr \tag{5.13}$$

for some $k \geq 0$ and words r, t (see, e.g., [5]). The resulting equation $z' = tr$ implies $\theta(z') = \theta(r)\theta(t)$ and combining this with $\theta(z') = rt$ results in $r = \theta(r)$ and $t = \theta(t)$. These r, t have to be distinct and non-empty in light of the θ-primitivity of v.

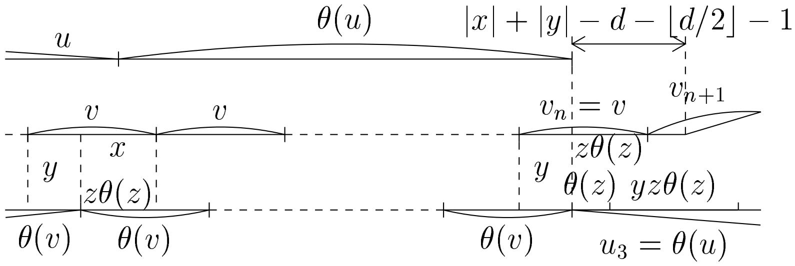

Figure 5.8: For an even n, $u\theta(u)^2$ and $v^n v_{n+1}$ share the common prefix of length $2|u|+|v|-d-\lfloor d/2\rfloor$, where $d=\gcd(|u|,|v|)$.

Next we prove that, under the assumption $v_{n+1}=v$, $|z'|$ has to be $2d$. By applying Euclidean algorithm to Eq. (5.13), we can obtain $d=\gcd(|z'|,|x|)=\gcd(|r|,|t|)$. Recall that $xv_{n+1}(=xz'\theta(z')x)$ and $\theta(z')xz'$ share a prefix of length at least $|\theta(z')xz'|-2d$. Removing the trivial common part $xz'=\theta(z')x$ from this prefix leaves us $|\theta(z')\wedge z'|\geq|z'|-2d$, that is, $|rt\wedge tr|\geq|rt|-2d$. So if $|z'|\geq 3d$, then Lemma 5.3 could be employed to give $\rho(r)=\rho(t)$, which would lead us to the contradiction with the θ-primitivity of v. Having successfully proved that $|z'|=2d$, let us construct a boundary common prefix based on u and v. Based on the presentations of x and z' in Eq. (5.13), we can see $v=tr(rt)^{k+1}r$ and $u=v^{(n-1)/2}tr$. Due to $z'=2d$ and $d=\gcd(|r|,|t|)$, we have $|t|=|r|=d$. By replacing (t,r) with $(\alpha,\beta)\in T_1\cup T_2\cup T_3$, we can get the first pair of presentations of u and v shown in the statement with $i\geq 1$. It is left to the readers to check that $|u\theta(u)^2\wedge v^{n+1}|=2|u|+|v|-d-\lfloor d/2\rfloor-1$.

The second subcase of **Case 1** ($u_3=\theta(u)$) is when n is even. Recall that x,y are θ-palindromes. In this subcase, x can be rather written as $x=z\theta(z)$ for some $z\in\Sigma^+$

(see Figure 5.8). As done before, one can obtain $\gcd(|y|, |z|) = d$ from $\gcd(|u|, |v|) = d$. The overlap between v_n and u_3 gives $z = \theta(z)$. Note that $v_n v_{n+1} = yz^2 v_{n+1}$ and $yu_3 = yzyz^2$ share their prefix of length at least $|y| + |x| + |y| - d - \lfloor d/2 \rfloor - 1$. Hence, after reducing their common prefix yz, still zv_{n+1} and yz^2 share their prefix of length at least $|y| + |z| - 2d$. Since $v_{n+1} \in \{y, z\}^+$, if $|yz| \geq 3d$, then due to Lemma 5.3 this common prefix would give $\rho(y) = \rho(z)$ and we have reached the contradiction. Thus, yz has to be of length $2d$, i.e., $|y| = |z| = d$. Then by replacing (y, z) with $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$, we obtain the first pair of presentations of u, v in the statement with $i = 0$. The boundary common prefix based on u and v is constructed in the same manner as previous case.

(Case 2): Let us remind ourselves of the case: "$u_2 = u_3 = u$, $n = 2m + 2$ for some $m \geq 1$, $v_2 = \cdots = v_{m+1} = v$, $v_{m+2} = \cdots = v_{2m+2} = \theta(v)$, $v = r(tr)^i (rt)^{i+j} r$, and $u = v^m r(tr)^i (rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty distinct θ-palindromes r, t". Note that the following equations hold:

$$u^2 = v^{m+1}\theta(v)^m (rt)^j, \tag{5.14}$$

$$u^3 = v^{m+1}\theta(v)^{m+1}(tr)^j v^{m-1} r(tr)^i (rt)^j. \tag{5.15}$$

Due to Lemma 5.18, if $2|u| - (2m+1)|v| = d$, then $u_2 = \theta(u)$. Since now we assume that $u_2 = u$, $2|u| - (2m+1)|v| \geq 2d$ must hold. Combining this with Eq. (5.14) implies $|(rt)^j| \geq 2d$. With Eq. (5.15), this gives $|(tr)^j \wedge v_{2m+3}| \geq |(tr)^j| - d - \lfloor d/2 \rfloor - 1$.

Note that v_{2m+3} begins with r regardless of whether it is v or $\theta(v)$.

The Euclidean algorithm derives $\gcd(|r(tr)^i|, |(tr)^j|) = d$ from $\gcd(|u|, |v|) = d$. Let $|(tr)^j| = kd$ for some $k \geq 2$. Here we shall see that unless $j = 1$, we could not avoid a contradiction. Suppose $j \geq 2$. If $k \geq 4$, then $|tr| \leq \frac{1}{2}|(tr)^j| \leq |(tr)^j| - 2d$. Thus, $|(tr)^j \wedge v_{2m+3}| \geq |tr|$, and Proposition 5.1 is applicable to this overlap to yield $\rho(r) = \rho(t)$. However, this contradicts the θ-primitivity of v. The same argument works for $k = 3$ and $j \geq 3$. If $k = 3$ and $j = 2$, then $|trtr| = 3d$. The Euclidean algorithm gives either $\gcd(|r|, 2|t|) = d$ (if i is even) or $\gcd(2|r|, |t|) = d$ (otherwise). Combining these with $|trtr| = 3d$ gives either $|r| = 2|t| = d$ (if i is even) or $2|r| = |t| = d$ (otherwise). The overlap between $(tr)^j$ and v_{2m+3} is of length at least d, which is long enough to get $r = t^2$ (if i is even) or $t = r^2$ (otherwise.) In either case, we cannot accept such a conclusion in light of the θ-primitivity of v. As a result, the remaining case is $k = 2$, i.e., $|(tr)^j| = 2d$. Then $\gcd(|r(tr)^i|, |(tr)^j|) = d$ gives $\gcd(|r(tr)^{i \bmod j}|, |(tr)^j|) = d$, and further $\gcd(|r(tr)^{i \bmod j}|, |(tr)^{(-i \bmod j)-1}t|) = d$. This means that $|r(tr)^{i \bmod j}| = |(tr)^{(-i \bmod j)-1}t| = d$ because they are properly shorter than $|(tr)^j| = 2d$. This further implies $i \bmod j = (-i \bmod j) - 1$ and $|r| = |t|$. Hence, j has to be odd, i.e., $j \geq 3$. With the non-emptiness of r and t, $|(tr)^j| = 2d$ now implies $d \geq 3$. As a result, $|(tr)^j \wedge v_{2m+3}| = d - \lfloor d/2 \rfloor - 1 \geq d/j = |t| = |r|$, and thus $t = r$, the same contradiction.

Consequently, the only one possible value of j which may create a boundary common prefix is 1. Then the Euclidean algorithm yields $\gcd(|r|, |t|) = d$ from

$\gcd(|r(tr)^i|, |tr|) = d$. If $|tr| \geq 3d$, then $|tr \wedge v_{2m+3}| \geq |tr| - 2d$ and the contradictory result $\rho(r) = \rho(t)$ would be obtained by Lemma 5.3. Thus, only the case $|tr| = 2d$, that is, $|t| = |r| = d$ remains valid. Actually in this case, substituting $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$ for (r, t) results in the second pair of presentations of (u, v) in the statement. One can easily check that $|u^3 \wedge v^{m+1}\theta(v)^{m+1}\alpha| = 2|u| + |v| - d - \lfloor d/2 \rfloor - 1$; note that α is a prefix of v_{2m+3} not depending on whether it is v or $\theta(v)$. □

Corollary 5.21. $|\mathrm{BCP}_\theta(u, v)| \leq 1$ *for any* $u, v \in \Sigma^+$ *with* $\rho_\theta(u) \neq \rho_\theta(v)$ *and* $|u| > |v| \geq 3 \gcd(|u|, |v|)$.

Proof. As shown in the proof of Theorem 5.20, once u and v are given in one of the presentations present there, there is only one way to construct an element of $\mathrm{BCP}_\theta(u, v)$. Furthermore, v is of length $\gcd(|u|, |v|)$ times an odd number in the first presentation, whereas is of length $\gcd(|u|, |v|)$ times an even number in the second one. □

5.3.3 The improved bound and its optimality

Combining Proposition 5.9 and Theorem 5.20 completes our proof of the improved bound for the extended Fine and Wilf's theorem.

Theorem 5.22. *Let* $u, v \in \Sigma^+$ *with* $|u| > |v| \geq 2 \gcd(|u|, |v|)$. *If a* θ*-power of* u *and a* θ*-power of* v *share a prefix of length* $b'(|u|, |v|)$, *then* $\rho_\theta(u) = \rho_\theta(v)$.

As opposed to the result mentioned in Theorem 5.11, $b'(p, q)$ is not optimal for all (p, q) with $p > q \geq 3\gcd(p, q)$. The presentations of u, v given in Theorem 5.20 make it possible to distinguish the non-optimal cases from the optimal cases.

Corollary 5.23. *For p, q with $d = \gcd(p, q)$ and $p > q \geq 3d$, $b'(p, q)$ is optimal for (p, q) if and only if $(p/d, q/d)$ is either $(m(2i+3)+2, 2i+3)$ or $(4m(i+1)+2i+3, 4(i+1))$ for some $m \geq 1$ and $i \geq 0$.*

Recall that the bound given by the classical Fine and Wilf's theorem is strongly optimal, i.e., for an arbitrary pair (p, q), there exists a word of length $p + q - \gcd(p, q) - 1$ with periods p, q but without $\gcd(p, q)$ as its period; furthermore if p and q are coprime, then such a word is unique up to letter renaming. In contrast, the bound $b'(p, q)$ is not strongly optimal. Indeed, Corollary 5.23 says that there do not exist u, v of respective lengths $9, 5$ with $\mathrm{BCP}_\theta(u, v) \neq \emptyset$. On the other hand, we can obtain an analogous result about the uniqueness of boundary common prefixes based on words of coprime lengths up to letter-renaming. For this purpose, let us construct all the boundary common prefixes according to Theorem 5.20 as well as its proof. The first presentation of u, v in this theorem is $u = (\alpha\beta(\beta\alpha)^i\beta)^m\alpha\beta$ and $v = \alpha\beta(\beta\alpha)^i\beta$ for some $m \geq 1$, $i \geq 0$, and $(\alpha, \beta) \in T_1 \cup T_2 \cup T_3$. The proof of this theorem says that the only boundary common prefix which can be generated based on u and v is the maximal common prefix between $u\theta(u)^2$ and $v^n v_{n+1}$, which is:

$$(\alpha\beta(\beta\alpha)^i\beta)^{2m+1}\alpha\beta x, \tag{5.16}$$

where x is the maximal common prefix between α and β (see the definition of T_1, T_2, T_3). In the similar fashion, for the second presentation in the theorem $u = (\alpha(\beta\alpha)^i(\alpha\beta)^{i+1}\alpha)^m\alpha(\beta\alpha)^i\alpha\beta$ and $v = \alpha(\beta\alpha)^i(\alpha\beta)^{i+1}\alpha$, $u^3 \wedge v^{m+1}\theta(v)^{m+1}v_{2m+3}$ is the only boundary common prefix constructable from u and v, and it is:

$$(\alpha(\beta\alpha)^i(\alpha\beta)^{i+1}\alpha)^{m+1}(\alpha(\beta\alpha)^{i+1}(\alpha\beta)^i\alpha)^{m+1}x, \tag{5.17}$$

where x is the maximal common prefix between α and β. Note that both presentations of u, v admit that $\gcd(|u|, |v|) = \gcd(|\alpha|, |\beta|) = |\alpha| = |\beta|$ due to the Euclidean algorithm. Therefore, all the boundary common prefixes which verify the optimality of $b'(p, q)$ for all the coprime pairs (p, q) for which $b'(p, q)$ is optimal can be obtained by choosing (α, β) in Eq. (5.16) and in Eq. (5.17) from

$$(\Sigma \times \Sigma) \cap (T_1 \cup T_2 \cup T_3) = \{(a, b) \mid a, b \in \Sigma, a \neq b, a = \theta(a), b = \theta(b)\}.$$

Consequently, for pairs of coprime integers, the next result holds, which is analogous to the uniqueness result just mentioned.

Corollary 5.24. *Let (p, q) be a pair of coprime integers with $p > q$. Then all the boundary common prefixes based on words of respective lengths p, q are equal up to renaming.*

Note that this uniqueness result does not hold any more once the coprime assumption is taken out. This is because the choice of x in Eq. (5.16) and in Eq. (5.17) is arbitrary and also even if $\gcd(p,q)=2$, there are two choices about (α,β) from T_2 or from T_3.

We conclude this section by defining two respective sets of boundary common prefixes thus obtained from Eq. (5.16) and Eq. (5.17) by limiting the choice of (α,β) only from $(\Sigma\times\Sigma)\cap(T_1\cup T_2\cup T_3)$. Due to the uniqueness mentioned in Corollary 5.24, we can set $\alpha=a$ and $\beta=b$ without loss of generality. As such, these sets are rather defined as:

$$\begin{aligned}
\mathrm{S_o} &= \{(ab(ba)^i b)^{2m+1}ab \mid i\geq 0, m\geq 1\} \\
\mathrm{S_e} &= \{(a(ba)^i(ab)^{i+1}a)^{m+1}(a(ba)^{i+1}(ab)^i a)^{m+1} \mid i\geq 0, m\geq 1\}.
\end{aligned}$$

The aim of the next section is to discuss the relationship between the words in $\mathrm{S_o}\cup\mathrm{S_e}$ and Sturmian words.

5.4 Sturmian words

It is known that for an arbitrary pair of integers (p,q) with $p>q>\gcd(p,q)$, there is a word of length $p+q-\gcd(p,q)-1$ which has p,q as its periods but $\gcd(p,q)$ is not its period, and hence, the bound $p+q-\gcd(p,q)$ for the Fine and Wilf's theorem is strongly optimal. Furthermore, all of these words can be constructed based on a (binary) word with two coprime periods p, q whose length is $p+q-2$. It is known that the set of all such basic words, denoted by PER, is closely related to Sturmian words. (Infinite) Sturmian words are (one-sided) infinite words which are not ultimately-periodic, and whose number of factors of length n is minimal $(n+1)$ for any $n \geq 1$. *Finite Sturmian words* are any factors of an infinite Sturmian word. Let St be the set of finite Sturmian words. By $F(w)$ we denote the set of all infixes (factors) of w and we can extend this notation to the set of words L as $F(L) = \bigcup_{w \in L} F(w)$. de Luca and Mignosi proved in [11] that $St = F(\mathrm{PER})$, that is, a binary word with two coprime periods whose length is the sum of these two periods minus 2 is a finite Sturmian word.

The aim of this section is to characterize $\mathrm{S_o}$ and $\mathrm{S_e}$, which correspond to PER for the optimal bound of Fine and Wilf's theorem, by finite Sturmian words.

A finite Sturmian word is called *standard* if it appears as an intermediate product (see Definition 1) when constructing an infinite Sturmian word using a procedure called *standard method*.

Definition 1 ([11])**.** Let $\Sigma = \{a, b\}$. The infinite sequence of pairs of words (A_n, B_n), $n \geq 0$, is constructed in the following manner. Set $(A_0, B_0) = (a, b)$. For any $n \geq 0$, the pair (A_{n+1}, B_{n+1}) is obtained from (A_n, B_n) by using one of the following two rules:

1. $(A_{n+1}, B_{n+1}) = (A_n, A_n B_n)$, or

2. $(A_{n+1}, B_{n+1}) = (B_n A_n, B_n)$.

The elements of $\{A_n, B_n \mid n \geq 0\}$ are the standard finite Sturmian words.

A property, called R in [11], plays an important role here. A word $w \in \Sigma^+$ is said to satisfy R if there exist palindromes x, y, z such that $w = zab = xy$. It was proved that a word with the property R is a standard Sturmian word.

Proposition 5.25 ([11]). *If a word has the property R, then it is a standard Sturmian word.*

Lemma 5.26. *For a word w in $\mathrm{S_e}$, the words wab and wba satisfy R.*

Proof. Let $w = ((ab)^i a(ab)^{i+1}a)^{m+1}(a(ba)^{i+1}a(ba)^i)^{m+1}$ for some $i \geq 0$ and $m \geq 1$. Since any word in $\mathrm{S_e}$ is a palindrome, it is enough, for our purpose, to show that wab is a product of two palindromes. In fact, wab can be split into $((ab)^i a(ab)^{i+1}a)^{m+1}a(ba)^i$ and $baa(ba)^i(a(ba)^{i+1}a(ba)^i)^m ab$, which are palindromes. Thus, wab satisfies R. In the same fashion, $wba = ((ab)^i a(ab)^{i+1}a)^m (ab)^i a \cdot (ab)^{i+1}a(a(ba)^{i+1}a(ba)^i)^{m+1}ba$, and hence, wba satisfies R. $\square$

Corollary 5.27. *For a word w in $\mathrm{S_e}$, wab and wba are standard Sturmian words.*

Thus, we can see that all words in $\mathrm{S_e}$ are finite Sturmian words. Combining Lemma 5.26 with the following result obtained in [11], which relates a word with the property R with PER, we can obtain a stronger result than this.

Lemma 5.28 ([11]). *Let $u = zab = xy$ for some palindromes x, y, z. If z contains at least two letters, then z has the periods $p = |x| + 2$ and $q = |y| - 2$ such that* $\gcd(p, q) = 1$.

Corollary 5.29. $\mathrm{S_e} \subseteq \mathrm{PER}$.

Having considered $\mathrm{S_e}$, now we turn our attention to $\mathrm{S_o}$. In a similar manner as above, we can prove that any element of $\mathrm{S_o}$ is a finite Sturmian word.

Lemma 5.30. *For a word w in $\mathrm{S_o}$, there exists a word $u \in \Sigma^+$ such that uw has the property R.*

Proof. Let $w = (ab(ba)^i b)^m ab$ for some $i \geq 0$ and $m \geq 1$. When $i = 0$, let $u = bb$. Then $uw = bb(abb)^m ab$. Since $bb(abb)^m$ is a palindrome and uw can be written as a product of b and $(bab)^{m+1}$, uw satisfies R.

When $i \geq 1$, let $u = (ba)^{i-1}b$ so that $uw = (ba)^{i-1}b(abba(ba)^{i-1}b)^m ab$. Note that uw has as prefix of length $|uw| - 2$ a θ-palindrome, and of length $|uw| - 2$, and it can be split into two palindromes $(ba)^{i-1}bab$ and $ba(ba)^{i-1}b(abba(ba)^{i-1}b)^{m-1}ab$. Thus, we can say that uw has the property R. $\square$

Corollary 5.31. *Words in $\mathrm{S_o}$ are finite Sturmian words.*

Now we know that all the words in $\mathrm{S_e} \cup \mathrm{S_o}$ are finite Sturmian words. We shall differentiate these two sets with respect to PER. Recall that any element of $\mathrm{S_e}$ is included in PER (Corollary 5.29). On the contrary, $\mathrm{S_o}$ and PER are disjoint. This

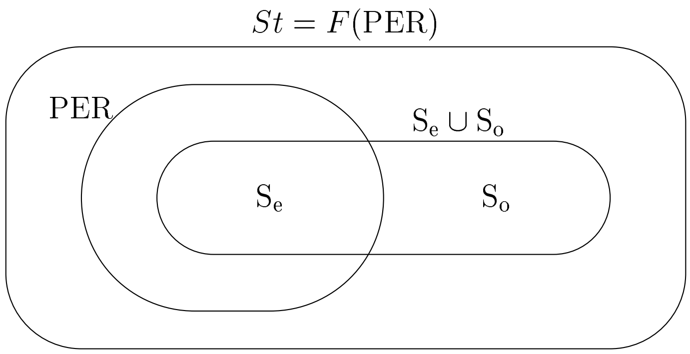

Figure 5.9: The set St of finite Sturmian words, PER, $\mathrm{S_e}$, and $\mathrm{S_o}$

is because an element of PER has been proved to be a palindrome [11], while any element of $\mathrm{S_o}$ is not.

Proposition 5.32. $\mathrm{S_o} \cap \mathrm{PER} = \emptyset$.

To summarize this discussion, Figure 5.9 clarifies the inclusion relations among the set of finite Sturmian words, PER, $\mathrm{S_e}$, and $\mathrm{S_o}$. Due to the fact that a factor of a word in St also belongs to St, $St \supseteq F(\mathrm{S_e} \cup \mathrm{S_o})$ holds, but this inclusion relation is in fact proper. For instance, a word $aaabaaaabaaa$ is of length 12 and has two periods 9 and 5 while $\gcd(9,5)$ is not its period. Hence, this word is in $\mathrm{PER} \subseteq St$, while it is not in $F(\mathrm{S_e} \cup \mathrm{S_o})$ because no word in $\mathrm{S_e} \cup \mathrm{S_o}$ has a continuous run of the same four letters as its infix. Moreover, the infix $baaaab$ of this example word shows $\mathrm{PER} \cup (\mathrm{S_e} \cup \mathrm{S_o}) \subsetneq St$. For the reason mentioned above, it is clear that $baaaab \in F(\mathrm{PER})$ but $baaaab \notin \mathrm{S_e} \cup \mathrm{S_o}$. In addition, $baaaab$ has only one period which is strictly smaller than its length, and hence, $baaaab \notin \mathrm{PER}$.

5.5 Concluding remarks

In this paper, we improved the bound for the extension of the Fine and Wilf's theorem of [9] from $b(p,q)$ to $b'(p,q) = b(p,q) - \lfloor \gcd(p,q)/2 \rfloor$. The complete characterization of boundary common prefixes given here allows us to distinguish the case when this improved bound is optimal in terms of the lengths of given words. In particular, this improved bound is optimal for any (p,q) with $p > q = 2\gcd(p,q)$. We also discussed the relationship between finite Sturmian words and the boundary common prefixes.

One open case is finding optimal bound for a pair (p,q) with $d = \gcd(p,q)$ and $p > q \geq 3d$, for which the improved bound $b'(p,q) = 2p+q-d-\lfloor d/2 \rfloor$ is not optimal due to Corollary 5.23. Note that for such (p,q), the bound $b'(p,q)-1$ remains good, while in Section 5.3.2, $2p + \lceil d/2 \rceil - 1$ was proved not to be good. Thus, the optimal bound for such (p,q) exists between $2p + \lceil d/2 \rceil$ and $b'(p,q) - 1$.

Bibliography

[1] J. Berstel and L. Boasson. Partial words and a theorem of Fine and Wilf. *Theoretical Computer Science*, 218(1):135–141, 1999.

[2] F. Blanchet-Sadri and R. A. Hegstrom. Partial words and a theorem of Fine and Wilf revisted. *Theoretical Computer Science*, 270:401–419, 2002.

[3] M. Gabriella Castelli, F. Mignosi, and A. Restivo. Fine and Wilf's theorem for three periods and a generalization of Sturmian words. *Theoretical Computer Science*, 218(1):83–94, 1999.

[4] S. Cautis, F. Mignosi, J. Shallit, M-w. Wang, and S. Yazdani. Periodicity, morphisms, and matrices. *Theoretical Computer Science*, 295:107–121, 2003.

[5] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 329–438. Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[6] S. Constantinescu and L. Ilie. Generalized Fine and Wilf's theorem for arbitrary number of periods. *Theoretical Computer Science*, 339(1):49–60, 2005.

[7] S. Constantinescu and L. Ilie. Fine and Wilf's theorem for Abelian periods. *Bulletin of the EATCS*, 89:167–170, June 2006.

[8] E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schützenberger result to pseudoperiodic words. In V. Diekert and D. Nowotka, editors, *Proc. DLT09*, volume 5583 of *Lecture Notes in Computer Science*, pages 183–194, Berlin, 2009. Springer-Verlag.

[9] E. Czeizler, L. Kari, and S. Seki. On a special class of primitive words. *Theoretical Computer Science*, 411(3):617–630, 2010.

[10] A. de Luca and A. De Luca. Pseudopalindrome closure operators in free monoids. *Theoretical Computer Science*, 362:282–300, 2006.

[11] A. de Luca and F. Mignosi. Some combinatorial properties of Sturmian words. *Theoretical Computer Science*, 136:361–385, 1994.

[12] N. J. Fine and H. S. Wilf. Uniqueness theorem for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, February 1965.

[13] J. Justin. On a paper by Castelli, Mignosi, Restivo. *RAIRO - Theoretical Informatics and Applications*, 34:373–377, 2000.

[14] L. Kari, B. Masson, and S. Seki. Properties of pseudo-primitive words and their applications. Submitted, available at `http://hal.archives-ouvertes.fr/hal-00458695/fr/`, 2009.

[15] F. Mignosi, A. Restivo, and P. V. Silva. On Fine and Wilf's theorem for bidimensional words. *Theoretical Computer Science*, 292:245–262, 2003.

Chapter 6

Improvement on the results of the extended Lyndon-Schützenberger equation

In this chapter, we introduce latest updates on the extended Lyndon-Schützenberger equation, which we discussed in Chapter 4. These results are under review of International Journal of Foundations of Computer Science (as of August 13, 2010).

Summary: A pseudo-primitive word with respect to an antimorphic involution θ is a word which cannot be written as a catenation of occurrences of a strictly shorter word t and $\theta(t)$. Properties of pseudo-primitive words are investigated in this paper. These properties link pseudo-primitive words with essential notions in combinatorics on words such as primitive words, (pseudo)-palindromes, and (pseudo)-commutativity. Their applications include an improved solution to the extended Lyndon-Schützenberger equation $u_1u_2\cdots u_\ell = v_1\cdots v_nw_1\cdots w_m$, where $u_1,\ldots,u_\ell \in \{u,\theta(u)\}$, $v_1,\ldots,v_n \in \{v,\theta(v)\}$, and $w_1,\ldots,w_m \in \{w,\theta(w)\}$ for some words u, v, w, integers $\ell, n, m \geq 2$, and an antimorphic involution θ. We prove that for $\ell \geq 4$, $n, m \geq 3$, this equation implies that u, v, w can be expressed in terms of a common word t and its image $\theta(t)$. Moreover, several cases of this equation where $\ell = 3$ are examined.

Properties of pseudo-primitive words and their applications

Lila Kari[1], Benoît Masson[2], and Shinnosuke Seki[1]

[1] Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

[2] IRISA (INRIA), Campus de Beaulieu, 35042 Rennes Cedex, France.

6.1 Introduction

For elements u, v, w in a free group, the equation of the form $u^\ell = v^n w^m$ $(\ell, n, m \geq 2)$ is known as the *Lyndon-Schützenberger equation* (LS equation for short). Lyndon and Schützenberger [13] investigated the question of finding all possible solutions for this equation in a free group, and proved that if the equation holds, then u, v, and w are all powers of a common element. This equation can be also considered on the semigroup of all finite words over a fixed alphabet Σ, and an analogous result holds.

Theorem 6.1 (see, e.g., [7, 13, 14]). *For words $u, v, w \in \Sigma^+$ and integers $\ell, n, m \geq 2$, the equation $u^\ell = v^n w^m$ implies that u, v, w are powers of a common word.*

The Lyndon-Schützenberger equation has been generalized in several ways; e.g., the equation of the form $x^k = z_1^{k_1} z_2^{k_2} \cdots z_n^{k_n}$ was investigated by Harju and Nowotka [8] and its special cases in [1, 11]. Czeizler et al. [3] have recently proposed another extension, which was originally motivated by the information encoded as DNA strands for DNA computing. In this framework, a DNA strand is modeled by a word w and encodes the same information as its Watson-Crick complement. In formal language theory, the Watson-Crick complementarity of DNA strands is modeled by an antimorphic involution θ [9, 15], i.e., a function θ on an alphabet Σ^* that is *(a)* antimorphic, $\theta(xy) = \theta(y)\theta(x)$, $\forall x, y \in \Sigma^*$, and *(b)* involution, $\theta^2 = id$, the identity. Thus, we can model the property whereby a DNA single strand binds to and is completely equivalent to its Watson-Crick complement, by considering a word u and its image $\theta(u)$ equivalent, for a given antimorphic involution θ.

For words u, v, w, integers $\ell, n, m \geq 2$, and an antimorphic involution θ, an extended Lyndon-Schützenberger equation (ExLS equation) is of the form

$$u_1 u_2 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m, \tag{6.1}$$

with $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. The question arises as to whether an equation of this form implies the existence of a word t such that $u, v, w \in \{t, \theta(t)\}^+$. A given triple (ℓ, n, m) of integers is said to *impose pseudo-periodicity, with respect to θ, on u, v, w*, or simply, to *impose θ-periodicity on u, v, w* if (6.1) implies $u, v, w \in \{t, \theta(t)\}^+$ for some word t. Furthermore, we say that the triple (ℓ, n, m) *imposes θ-periodicity* if it imposes θ-periodicity on all u, v, w. The known results on ExLS equations [3] are summarized in Table 6.1.

This paper is a step towards solving the unsettled cases of Table 6.1, by using the following strategy. Concise proofs exist in the literature for Theorem 6.1, that

l	n	m	θ-periodicity
≥ 5	≥ 3	≥ 3	YES
3 or 4	≥ 3	≥ 3	?
2	≥ 2	≥ 2	?
≥ 3	2	≥ 2	NO

Table 6.1: Summary of the known results regarding the extended Lyndon-Schützenberger equation.

make use of fundamental properties such as:

(i) The periodicity theorem of Fine and Wilf (FW theorem),

(ii) The fact that a primitive word cannot be a proper infix of its square, and

(iii) The fact that the class of primitive words is closed under cyclic permutation.

(For details of each, see [2].) In contrast, the proof given in [3] for the result about ExLS equations, stating that $(\geq 5, \geq 3, \geq 3)$ imposes θ-periodicity, involves techniques designed for this specific purpose only. Should Properties (i), (ii), (iii) be generalized so as to take into account the informational equivalence between a word u and $\theta(u)$, they could possibly form a basis for a concise proof of the solutions to the ExLS equation. The approach we use in this paper is thus to seek analog properties for this extended case, and use the results we obtain to approach the unsettled cases in Table 6.1.

Czeizler, Kari, and Seki generalized Property (i) in [4]. There, first the notion of primitive words was extended to that of pseudo-primitive words with respect to a given antimorphic involution θ (or simply, θ-primitive words). A word u is said to be *θ-primitive* if there does not exist another word t such that $u \in t\{t, \theta(t)\}^+$. For example, if θ is the mirror image over $\{a, b\}^*$ (the identity function on $\{a, b\}$ extended to an antimorphism on $\{a, b\}^*$), $aabb$ is θ-primitive, while $abba$ is not because it can be written as $ab\theta(ab)$. Based on the θ-primitivity of words, Property (i) was generalized as follows: "For words u, v, if a word in $u\{u, \theta(u)\}^*$ and a word in $v\{v, \theta(v)\}^*$ share a long enough prefix (for details, see Theorems 6.5 and 6.6), then $u, v \in t\{t, \theta(t)\}^*$ for some θ-primitive word t."

In contrast, little is known about Properties (ii) and (iii) except that they cannot be generalized as suggested in the previous example: non-trivial overlaps between two words in $\{t, \theta(t)\}^+$ are possible, and cyclic permutations do not in general preserve the θ-primitivity of words. As a preliminary step towards an extension of Property (ii), Czeizler et al. examined the non-trivial overlap of the form $v_1 \cdots v_m x = y v_{m+1} \cdots v_{2m}$, where $m \geq 1$, v_i is either v or $\theta(v)$ for some θ-primitive word v $(1 \leq i \leq 2m)$, and both x and y are properly shorter than v [3]. Some of the results obtained there will be employed in this paper.

One purpose of this paper is to explore further the extendability of Properties (ii) and (iii). The main result here is Theorem 6.12, which states that for a θ-primitive word x, neither $x\theta(x)$ nor $\theta(x)x$ can be a proper infix of a word $x_1x_2x_3$, where $x_1, x_2, x_3 \in \{x, \theta(x)\}$. Based on this result, we consider two problems: For a θ-primitive word x, (1) does $v, yvz \in \{x, \theta(x)\}^+$ imply that y and z are in $\{x, \theta(x)\}^*$?;

and (2) if the catenation of words u, v is in $\{x, \theta(x)\}^+$, under what conditions does $u, v \in \{x, \theta(x)\}^*$ hold? In particular, our investigation into the second problem will reveal close relationships between primitive words, θ-primitive words, and θ-palindromes (fixed points of θ). These relationships further present several cyclic permutations under which the θ-primitivity of words is preserved.

The results thus obtained enable us to prove that the triple $(4, \geq 3, \geq 3)$ imposes θ-periodicity (Theorem 6.48) in a much simpler manner than the proof in [3] for $(\geq 5, \geq 3, \geq 3)$. Even for $(3, n, m)$ ExLS equations, these results give some insight and narrow down the open cases of ExLS equations.

The paper is organized as follows: in the next section, we provide required notions and notation. Section 6.3 begins with the proof of some basic properties of θ-primitive words, and then proves some consequences of overlaps between θ-primitive words of a similar flavour with Properties (ii) and (iii) (e.g., Theorem 6.12, Corollary 6.23). These tools are used in Section 6.4, where we prove that the $(4, \geq 3, \geq 3)$ ExLS equation has only θ-periodic solutions (Theorem 6.48), and study particular cases of $(3, n, m)$ ExLS equations.

6.2 Preliminaries

An alphabet is a finite and non-empty set of symbols. In the sequel, we shall use a fixed non-singleton alphabet Σ. The set of all words over Σ is denoted by Σ^*, which includes the empty word λ, and let $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$. The length of a word $w \in \Sigma^*$ is denoted by $|w|$. A word v is an *infix* (resp. *prefix*, *suffix*) of a word w if $w = xvy$ (resp. $w = vy$, $w = xv$) for some $x, y \in \Sigma^*$; in any case, if $w \neq v$, then the infix (prefix, suffix) is said to be *proper*. For a word w, denote by $\mathrm{Pref}(w)$ the set of prefixes of w and by $\mathrm{Suff}(w)$ the set of its suffixes.

A language L is a subset of Σ^*. For a non-negative integer $n \geq 0$, we write L^n for the language consisting of all words of the form $w_1 \cdots w_n$ such that each w_i is in L. We also write $L^{\geq n}$ for $L^n \cup L^{n+1} \cup L^{n+2} \cup \cdots$. Analogously, we can define $L^{\leq n} = L^0 \cup L^1 \cup \cdots \cup L^n$. For $L^{\geq 0}$ and $L^{\geq 1}$, we employ the traditional notation L^* and L^+.

A mapping $\theta : \Sigma^* \to \Sigma^*$ is called an *antimorphic involution* of Σ^* if $\theta(xy) = \theta(y)\theta(x)$ for any $x, y \in \Sigma^*$ (antimorphism), and θ^2 is equal to the identity (involution). Throughout this paper, θ denotes an antimorphic involution. The *mirror image*, which maps a word to its reverse, is a typical example of antimorphic involution. A word $w \in \Sigma^*$ is called a *θ-palindrome* if $w = \theta(w)$. A word which is a θ-palindrome with respect to a given but unspecified antimorphic involution θ is also called a *pseudo-palindrome* [5].

A non-empty word $w \in \Sigma^+$ is said to be *primitive* if $w = v^n$ implies $n = 1$ for any word $v \in \Sigma^+$. It is known that any non-empty word $w \in \Sigma^+$ can be written as a power of a unique primitive word, which is called the *primitive root* of w, and denoted by $\rho(w)$. Two words which *commute* share a primitive root, that is, $uv = vu$ implies

$\rho(u) = \rho(v)$ (see [2]). In literature, it is said that $uv = vu$ causes a *defect effect* (for details of defect effects and defect theorems, see [2, 14]). The LS equation also causes defect effect, since $u^\ell = v^n w^m$ with $\ell, n, m \geq 2$ implies $\rho(u) = \rho(v) = \rho(w)$ (Theorem 6.1). The following results describe other relations causing a defect effect.

Lemma 6.2 ([4]). *Let $u \in \Sigma^+$ such that $u = pq$ for some θ-palindromes $p, q \in \Sigma^+$. If $q \in \mathrm{Pref}(u)$ and $|q| \geq |p|$, then $\rho(p) = \rho(q) = \rho(u)$.*

Theorem 6.3 ([2]). *Let $u, v \in \Sigma^+$. If there exist $\alpha(u, v) \in u\{u, v\}^*$ and $\beta(u, v) \in v\{u, v\}^*$ which share a prefix of length at least $|u| + |v|$, then $\rho(u) = \rho(v)$.*

The notion of primitive word was generalized into that of pseudo-primitive word by Czeizler, Kari, and Seki [4]. For an antimorphic involution θ, a non-empty word $w \in \Sigma^+$ is said to be *pseudo-primitive with respect to θ*, or simply *θ-primitive*, if $w \in \{v, \theta(v)\}^n$ implies $n = 1$ for any word $v \in \Sigma^+$. In [4] it was proved that for any non-empty word $w \in \Sigma^+$, there exists a unique θ-primitive word t satisfying $w \in t\{t, \theta(t)\}^*$. Such a word t is called the *θ-primitive root* of w. The next lemma describes a property of the θ-primitive root of a θ-palindrome of even length.

Lemma 6.4. *Let $x \in \Sigma^+$ be a θ-primitive word and p be a θ-palindrome of even length. If $p = x_1 x_2 \cdots x_m$ for some $m \geq 1$ and $x_1, \ldots, x_m \in \{x, \theta(x)\}$, then m has to be even.*

Proof. Suppose that the equality held for some odd m. Then x must be of even length because $|p|$ is even. Hence $x_{(m-1)/2}$ becomes a θ-palindrome. Thus $x = y\theta(y)$

for some $y \in \Sigma^+$. However, this contradicts the θ-primitivity of x. □

The *theorem of Fine and Wilf* (FW theorem) is one of the fundamental results on periodicity [6]. It states that for two words $u, v \in \Sigma^+$, if a power of u and a power of v share a prefix of length at least $|u| + |v| - \gcd(|u|, |v|)$, then $\rho(u) = \rho(v)$, where $\gcd(\cdot, \cdot)$ denotes the greatest common divisor of two arguments (for its proof, see, e.g., [2]). This theorem has been generalized in [4], by taking into account the equivalence between a word and its image under θ, in the following two forms.

Theorem 6.5 ([4]). *Let $u, v \in \Sigma^+$. If a word in $\{u, \theta(u)\}^*$ and a word in $\{v, \theta(v)\}^*$ share a prefix of length at least $\mathrm{lcm}(|u|, |v|)$, then $u, v \in \{t, \theta(t)\}^+$ for some θ-primitive word $t \in \Sigma^+$, where $\mathrm{lcm}(\cdot, \cdot)$ denotes the least common multiple of two arguments.*

Theorem 6.6 ([4]). *Let $u, v \in \Sigma^+$ with $|u| \geq |v|$. If a word in $\{u, \theta(u)\}^*$ and a word in $\{v, \theta(v)\}^*$ share a prefix of length at least $2|u| + |v| - \gcd(|u|, |v|)$, then $u, v \in \{t, \theta(t)\}^+$ for some θ-primitive word $t \in \Sigma^+$.*

In a way, we can say that these theorems describe relations causing a *weak defect effect* because they all imply that $u, v \in \{t, \theta(t)\}^+$ for some θ-primitive word $t \in \Sigma^+$, which is strictly weaker than the usual defect effect $\rho(u) = \rho(v)$ [4]. Various relations causing such a weak defect effect were presented in [4].

Besides, the commutativity $xy = yx$ was extended to the θ-commutativity $xy = \theta(y)x$ in [10]. This is a special case of $xy = zx$, whose solutions are given as

$x = r(tr)^i$, $y = (tr)^j$, and $z = (rt)^j$ for some $i \geq 0$, $j \geq 1$, and $r, t \in \Sigma^*$ such that rt is primitive (see, e.g., [2]). The next proposition immediately follows from this; note that the θ-commutativity equation guarantees that both r, t are θ-palindromes.

Proposition 6.7 ([10]). *For $x, y \in \Sigma^+$, the solutions of $xy = \theta(y)x$ are given by $x = r(tr)^i$ and $y = (tr)^j$ for some $i \geq 0$, $j \geq 1$, and θ-palindromes r, t such that rt is primitive.*

Although this equation does not cause even a weak defect effect, one encounters it often when considering word equations which involve θ. Note that for words $u, v \in \Sigma^*$, it was proved in [4] that the system $uv = \theta(uv)$ and $vu = \theta(vu)$ causes a weak defect effect: $u, v \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$. Thus for words x, y, z satisfying $xy = zx$, if both y and z are θ-palindromes, then the representation of solutions of $xy = zx$ implies $tr = \theta(tr)$ and $rt = \theta(rt)$. Hence the next result holds.

Proposition 6.8 ([3]). *For a word $x \in \Sigma^+$ and two θ-palindromes $y, z \in \Sigma^+$, the equation $xy = zx$ implies that $x, y, z \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$.*

6.3 Properties of Pseudo-Primitive Words

The primitivity of words is one of the most essential notions in combinatorics on words. The past few decades saw a considerable number of studies on this topic (see e.g., [2, 12, 16]). In contrast, research on the pseudo-primitivity of words has just been initiated in [3, 4]. For instance, although the class of pseudo-primitive words was proved to be properly included in that of primitive words [4], nothing else is known about the relation between these two classes. The purpose of this section is to prove various properties of pseudo-primitive words.

Throughout this section, θ is assumed to be a given antimorphic involution. We begin this section with a simple extension of a known result on the primitive root (Lemma 6.9) to the θ-primitive root (Lemma 6.10).

Lemma 6.9 (e.g., [16]). *For words $u, v \in \Sigma^+$ and a primitive word $w \in \Sigma^+$, the following properties hold:*

1. *$u^n \in w^+$ implies $u \in w^+$;*
2. *$uv, u \in w^+$ or $uv, v \in w^+$ implies $u, v \in w^+$.*

Lemma 6.10. *For words $u, v \in \Sigma^+$ and a θ-primitive word $x \in \Sigma^+$, the following properties hold:*

1. *for some $n \geq 1$, $u^n \in \{x, \theta(x)\}^+$ implies $u \in \{x, \theta(x)\}^+$;*
2. *$uv, u \in \{x, \theta(x)\}^+$, or $uv, v \in \{x, \theta(x)\}^+$ implies $u, v \in \{x, \theta(x)\}^+$;*
3. *$\theta(u)v, u \in \{x, \theta(x)\}^+$, or $u\theta(v), v \in \{x, \theta(x)\}^+$ implies $u, v \in \{x, \theta(x)\}^+$.*

Proof. The first property follows from Theorem 6.5, while the others are immediately proved by comparing the length of words. $\square$

As mentioned in the introduction, if a word w is primitive, then the equation $w^2 = ywz$ implies either $y = \lambda$ or $z = \lambda$. Since a θ-primitive word is primitive, this applies to θ-primitive words, too; a θ-primitive word x cannot be a proper infix of x^2. However, due to the informational equivalence between x and $\theta(x)$, we should consider equations like $x^2 = y\theta(x)z$ as well, and in fact this equation can hold with non-empty y and z. Nevertheless, we can state an analogous theorem based on the next lemma.

Lemma 6.11 ([4])**.** *Let $x \in \Sigma^+$ be a θ-primitive word, and x_1, x_2, x_3, $x_4 \in \{x, \theta(x)\}$. If $x_1x_2y = zx_3x_4$ for some non-empty words $y, z \in \Sigma^+$ with $|y|, |z| < |x|$, then $x_2 \neq x_3$.*

Theorem 6.12. *For a θ-primitive word $x \in \Sigma^+$, neither $x\theta(x)$ nor $\theta(x)x$ can be a proper infix of a word in $\{x, \theta(x)\}^3$.*

Proof. Let $x_1, x_2, x_3 \in \{x, \theta(x)\}$ and suppose that $x\theta(x)$ is a proper infix of $x_1x_2x_3$. That is to say, there exist words $y, z, y', z' \in \Sigma^+$, $0 < |y|, |z|, |y'|, |z'| < |x|$ such that $zx\theta(x) = x_1x_2y$ and $x\theta(x)y' = z'x_2x_3$. By Lemma 6.11, the first equation implies that $x_2 \neq x$ and the second that $x_2 \neq \theta(x)$, this is in contradiction with $x_2 \in \{x, \theta(x)\}$. We prove similarly that $\theta(x)x$ cannot be a proper infix of $x_1x_2x_3$. $\square$

This theorem will lead us to two propositions (Propositions 6.16 and 6.20), as well as to several other results. The main usage of these propositions in this paper is the following "splitting strategy," which shall prove useful in solving ExLS equations in Section 6.4. Given "complicated" words in $\{x, \theta(x)\}^+$ for a θ-primitive word x, these propositions make it possible to split such words into "simple" component words which are still in $\{x, \theta(x)\}^+$. Then, Lemmas 6.9 and 6.10 are often applicable to subdivide these simple components into smaller units in $\{x, \theta(x)\}^+$.

Recall that a primitive word cannot be a proper infix of its square. It is hence evident that for a primitive word w, if a word u in w^+ contains w as its infix like $u = ywz$ for some $y, z \in \Sigma^*$, then $y, z \in w^*$. For such w, more generally, $v, yvz \in w^+$ implies $y, z \in w^*$. This raises a naturally extended question of whether for a θ-primitive word x, if $v, yvz \in \{x, \theta(x)\}^+$, then $y, z \in \{x, \theta(x)\}^*$ holds or not. Although this is not always the case, we provide some positive cases based on the following lemma, which is a natural consequence of Theorem 6.12.

Lemma 6.13. *Let x be a θ-primitive word, and $v \in \Sigma^+$. For $y, z \in \Sigma^*$, either $yx\theta(x)z \in \{x, \theta(x)\}^*$ or $y\theta(x)xz \in \{x, \theta(x)\}^*$ implies $y, z \in \{x, \theta(x)\}^*$.*

Proof. We prove that $yx\theta(x)z \in \{x, \theta(x)\}^*$ implies $y, z \in \{x, \theta(x)\}^*$. Let $yx\theta(x)z = x_1 \cdots x_n$ for some $n \geq 2$ and $x_1, \ldots, x_n \in \{x, \theta(x)\}$. In light of Theorem 6.12, there must exist such i that $y = x_1 \cdots x_{i-1}$, $x\theta(x) = x_i x_{i+1}$, and $z = x_{i+2} \cdots x_n$. □

Lemma 6.14. *Let x be a θ-primitive word, and $v \in \Sigma^+$. If $v, yvz \in \{x, \theta(x)\}^*$ for some $y, z \in \Sigma^*$ and either $x\theta(x)$ or $\theta(x)x$ is an infix of v, then $y, z \in \{x, \theta(x)\}^*$.*

Proof. Here we consider only the case when $x\theta(x)$ is an infix of v. Due to Lemma 6.13, we can let $v = x'x\theta(x)x''$ for some $x', x'' \in \{x, \theta(x)\}^*$. Thus, $yvz = yx'x\theta(x)x''z \in \{x, \theta(x)\}^{\geq 2}$. From this, the same lemma derives $yx', x''z \in \{x, \theta(x)\}^*$. Based on Lemma 6.10, we obtain $y, z \in \{x, \theta(x)\}^*$. □

Lemma 6.14 is a generalization of Lemma 6.13, and makes it possible to prove the following two propositions.

Proposition 6.15. *Let x be a θ-primitive word, and $v \in \Sigma^+$. If $v, yvz \in \{x, \theta(x)\}^{\geq 2}$ for some $y, z \in \Sigma^*$ and v is primitive, then $y, z \in \{x, \theta(x)\}^*$.*

Proof. Let $v = x_1 \cdots x_m$ for some $m \geq 2$ and $x_1, \ldots, x_m \in \{x, \theta(x)\}$. Since v is primitive, there exists $1 \leq i \leq m$ such that $x_i x_{i+1} \in \{x\theta(x), \theta(x)x\}$. Now we can employ Lemma 6.14 to get this result. □

Proposition 6.16. *Let x be a θ-primitive word, and $v \in \Sigma^+$. If $v, yvz \in \{x, \theta(x)\}^+$ for some $y, z \in \Sigma^*$ and v is a non-empty θ-palindrome, then $y, z \in \{x, \theta(x)\}^*$.*

Proof. Let $v = x_1 \cdots x_n$ for some $n \geq 1$ and $x_1, \ldots, x_n \in \{x, \theta(x)\}$. If n is odd, then $v = \theta(v)$ implies $x_{(n+1)/2} = \theta(x_{(n+1)/2})$ and this means $x = \theta(x)$. Thus we have $v, yvz \in x^+$, and hence $y, z \in x^*$. If n is even, then $x_{n/2} x_{n/2+1} \in \{x\theta(x), \theta(x)x\}$ so that $y, z \in \{x, \theta(x)\}^*$ due to Lemma 6.14. □

From now on, we address the following question: "for a θ-primitive word x and two words $u, v \in \Sigma^*$ such that $uv \in \{x, \theta(x)\}^+$, under what conditions on u, v, we can say $u, v \in \{x, \theta(x)\}^*$?". Here we provide several such conditions. Among them is Proposition 6.20, which serves for the splitting strategy. As its corollary, we will obtain relationships between primitive words and θ-primitive words (Corollaries 6.21 and 6.22).

Proposition 6.17. *Let x be a θ-primitive word, $u \in \mathrm{Suff}(\{x, \theta(x)\}^+)$, and $v \in \mathrm{Pref}(\{x, \theta(x)\}^+)$. If $uv = x_1 \cdots x_m$ for some integer $m \geq 2$ and $x_1, \ldots, x_m \in \{x, \theta(x)\}$, then either $u, v \in \{x, \theta(x)\}^+$ or $x_1 = \cdots = x_m$.*

Proof. Let us prove that when $u, v \notin \{x, \theta(x)\}^+$, $x_1 = \cdots = x_m$ must hold. Let $u = z'_s x'_{i-1} \cdots x'_1$ for some $i \geq 1$, $x'_i, \ldots, x'_1 \in \{x, \theta(x)\}$, and some non-empty words $z'_p, z'_s \in \Sigma^+$ such that $z'_p z'_s = x'_i$. We can also let $v = x''_1 \cdots x''_{j-1} z''_p$ for some $j \geq 1$, $x''_1, \ldots, x''_j \in \{x, \theta(x)\}$, and $z''_p, z''_s \in \Sigma^+$ such that $z''_p z''_s = x_j$. Now we have $x'_i \cdots x'_1 x''_1 \cdots x''_j = z'_p uv z''_s = z'_p x_1 \cdots x_m z''_s$. Since $0 < |z'_p| < |x|$, Theorem 6.12 implies $x_1 = \cdots = x_m$. $\square$

Corollary 6.18. *Let x be a θ-primitive word, and $u \in \mathrm{Suff}(\{x, \theta(x)\}^+)$, $v \in \mathrm{Pref}(\{x, \theta(x)\}^+)$. If uv is in $\{x, \theta(x)\}^{\geq 2}$ and primitive, then $u, v \in \{x, \theta(x)\}^+$.*

Proposition 6.17 gives the following two propositions which play an important role in investigating the ExLS equation.

Proposition 6.19. *Let x be a θ-primitive word, and $u, v \in \Sigma^+$. If $uv, vu \in \{x, \theta(x)\}^n$ for some $n \geq 2$, then one of the following statements holds:*

1. *$u, v \in \{x, \theta(x)\}^+$;*

2. *$uv = x^n$ and $vu = \theta(x)^n$;*

3. *$uv = \theta(x)^n$ and $vu = x^n$.*

Proof. We have $v \in \mathrm{Pref}(\{x, \theta(x)\}^+)$ and $u \in \mathrm{Suff}(\{x, \theta(x)\}^+)$ because $vu \in \{x, \theta(x)\}^n$. Proposition 6.17 implies that either the first property holds or $uv \in \{x^n, \theta(x)^n\}$. Here we consider only the case when $uv = x^n$. Then $u = x^i x_p$ and $v = x_s x^{n-i-1}$ for some $1 \leq i \leq n$ and $x_p, x_s \in \Sigma^+$ with $x = x_p x_s$. Thus, we have $x_p vu x_s = x^{n+1}$, from which can deduce $vu = \theta(x)^n$ with the aid of Theorem 6.12 and the fact that x cannot be a proper infix of its square. $\square$

Proposition 6.20. *Let $x \in \Sigma^+$ be a θ-primitive word, and $p, q \in \Sigma^+$ be θ-palindromes. If pq is primitive, and $pq = x_1 \cdots x_n$ for some $n \geq 2$ and $x_1, \ldots, x_n \in \{x, \theta(x)\}$, then there are integers $k, m \geq 1$ such that $n = 2m$, $p = x_1 \cdots x_{2k}$, and $q = x_{2k+1} \cdots x_{2m}$.*

Proof. It is clear from $pq = x_1 \cdots x_n$ that $p \in \mathrm{Pref}(\{x, \theta(x)\}^+)$ and $q \in \mathrm{Suff}(\{x, \theta(x)\}^+)$. Since both p and q are θ-palindromes, these mean that $p \in \mathrm{Suff}(\{x, \theta(x)\}^+)$ and $q \in \mathrm{Pref}(\{x, \theta(x)\}^+)$. Hence we can apply Proposition 6.17 to obtain $p = x_1 \cdots x_i$ and $q = x_{i+1} \cdots x_n$ for some i (since pq is primitive, the case $x_1 = \cdots = x_n$ is impossible).

The integer i has to be even ($i = 2k$ for some $k \geq 1$). Suppose not, then p being a θ-palindrome implies that $x_{(i+1)/2}$ is a θ-palindrome, and hence so is x. As a result, $pq = x^n$ but this contradicts the assumption that pq is primitive. Similarly, $n - i$ proves to be even, too, and we obtain $n = 2m$. $\square$

The next two corollaries follow from Proposition 6.20. The first one provides us with a sufficient condition for a primitive word that is a catenation of two non-empty

θ-palindromes to be θ-primitive.

Corollary 6.21. *For non-empty θ-palindromes p, q, if pq is primitive but there does not exist any x such that $p, q \in \{x, \theta(x)\}^+$, then pq is θ-primitive.*

Corollary 6.22. *Let p, q be non-empty θ-palindromes such that pq is primitive. Then some word in $\{p, q\}^+$ is θ-primitive if and only if pq is θ-primitive.*

Proof. The converse implication is trivial because $pq \in \{p, q\}^+$. The direct implication can be proved by considering its contrapositive, which is immediately given by Proposition 6.20. □

Note that in the statement of Corollary 6.22 we cannot replace the quantifier "some" with "all". A trivial example is $(pq)^2 \in \{p, q\}^+$, which is not even primitive. We can also provide a non-trivial example as follows:

Example 16. Let θ be the mirror image over $\{a, b\}^*$, $p = a$, and $q = baaab$. It is clear that $pq = abaaab$ is θ-primitive. On the other hand, $qppp = (baaa)^2 \in \{p, q\}^+$ is not even primitive.

Corollary 6.22 gives a further corollary about the case in which a word obtained from a θ-primitive word by cyclic permutation remains θ-primitive.

Corollary 6.23. *For two non-empty θ-palindromes p, q, if pq is θ-primitive, then qp is θ-primitive.*

Proof. Since pq is θ-primitive, it is primitive and hence its conjugate qp is also primitive. Applying Corollary 6.22 to qp gives the result. □

Corollary 6.23 gives a partial answer to one of our questions on the preservation of θ-primitivity under cyclic permutation.

Now let us examine the equation $pq = x_1 \cdots x_n$ from a different perspective to get some results useful in Section 6.4. Here we see that the assumptions considered in Proposition 6.20: pq being primitive and both of p, q being a θ-palindrome are critical to obtain $p, q \in \{x, \theta(x)\}^+$.

Lemma 6.24. *For a θ-primitive word $x \in \Sigma^+$ and $k \geq 2$, let $x_1, x_2, \ldots, x_k \in \{x, \theta(x)\}$. If $pz = x_1 x_2 \cdots x_k$ for some θ-palindrome p and non-empty word $z \in \Sigma^+$ with $|z| < |x|$, then $x_1 = x_2 = \cdots = x_{k-1}$. Moreover, if z is also a θ-palindrome, then $x_k = x_{k-1}$.*

Proof. Due to the length condition on z, we can let $x_k = yz$ for some non-empty word $y \in \Sigma^+$. Hence we have $p = x_1 x_2 \cdots x_{k-1} y$. Since p is a θ-palindrome, $p = \theta(y)\theta(x_{k-1}) \cdots \theta(x_1)$. This means that $\theta(x_{k-1}) \cdots \theta(x_1)$ is a proper infix of $x_1 \cdots x_k$, and we can say that $x_1 = \cdots = x_{k-1}$ using Theorem 6.12 (we can assume $k \geq 3$, since if $k = 2$ the consequence is trivial).

Now we consider the additional result when $z = \theta(z)$. Without loss of generality, we can assume that $x_1 = x$. So we have $p = x^{k-1} y = \theta(y)\theta(x)^{k-1}$. Since $|y| < |\theta(x)|$, this equation gives $\theta(x) = qy$ for some non-empty word q. Actually q is a θ-palindrome. Indeed, we have $qy \in \mathrm{Suff}(p) = \mathrm{Suff}(x^{k-1}y)$, hence as $|q| < |x|$, $q \in \mathrm{Suff}(x)$. Moreover, by definition, $q \in \mathrm{Pref}(\theta(x))$, therefore $\theta(q) \in \mathrm{Suff}(x)$ and thus q has to be a θ-palindrome.

Thus, if $x_k = \theta(x)$, then $\theta(x) = qy = yz$ and hence $\theta(x)$ could not be θ-primitive due to Proposition 6.8, raising a contradiction. $\square$

For two θ-palindromes p, q, a θ-primitive word x, and $x_1, \ldots, x_k \in \{x, \theta(x)\}$ ($k \geq 1$), if $|q| < |x|$, then the equation $pq = x_1 \cdots x_k$ turns into $pq = x^k$ due to Lemma 6.24 and its solution is $x = p'q$ for some θ-palindrome p' such that $p = x^{k-1}p'$. If we replace q in this equation with a word z, which is not assumed to be a θ-palindrome, and if $k \geq 3$, then we can still find an intriguing non-trivial solution to the equation $pz = x^{k-1}\theta(x)$.

Example 17. Let p be a θ-palindrome, x be a θ-primitive word, and $z \in \Sigma^+$ with $|z| < |x|$. For some $i \geq 0$, $j \geq 1$, $k \geq 3$, and θ-palindromes r, t such that rt is primitive, we can see that $x = [r(tr)^i]^2(tr)^j$, $p = x^{k-1}r(tr)^i$, and $z = (tr)^j r(tr)^i$ satisfy $pz = x^{k-1}\theta(x)$.

Note that r and t in this example are given by Proposition 6.7. Further research on the properties of words in $\{r(tr)^i, (tr)^j\}^*$ may shed light on the properties of θ-primitive words. In Section 6.4.2, we will provide some results along this line, such as the ones in Propositions 6.34 and 6.35.

6.4 Extended Lyndon-Schützenberger equation

As an application of the results obtained in Section 6.3, we address some open cases of the extended Lyndon-Schützenberger equation in this section.

For $u, v, w \in \Sigma^+$, the ExLS equation under consideration is of the form

$$u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m,$$

where $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$, for $\ell, n, m \geq 2$. The open cases are $\ell \in \{2, 3, 4\}$ and $m, n \geq 3$ (see Table 6.1). It suffices to consider the case when both v and w are θ-primitive; otherwise we simply replace them with their θ-primitive roots and increase the parameters n and m. The words $v_1 \cdots v_n$ and $w_1 \cdots w_m$ being symmetric with respect to their roles in the equation, it is also legitimate to assume that $|v_1 \cdots v_n| \geq |w_1 \cdots w_m|$.

Throughout Subsections 6.4.1 to 6.4.4, we prove that the triple $(4, \geq 3, \geq 3)$ imposes θ-periodicity. First of all, in Subsection 6.4.1, the problem which we actually work on is formalized as Problem 6.1, and we solve some special instances of ExLS equation to which the application of the generalized Fine and Wilf's theorem (Theorem 6.5) immediately proves the existence of a word t satisfying $u, v, w \in \{t, \theta(t)\}^+$. We call such instances *trivial ExLS equations*. In Subsection 6.4.2, we provide additional conditions which can be assumed for non-trivial ExLS equations. Several lemmas and propositions are also proved there. They are interesting in their own and our proof techniques for them probably include various applications beyond the investigation on the non-trivial ExLS equations in Subsection 6.4.3 (the case when $u_2 = u_1$) and Subsection 6.4.4 (the case when $u_2 \neq u_1$). In each of these subsections, we analyze four cases depending on the values of u_3 and u_4 one at a time. All of these proofs merely consist of direct applications of the results obtained so far and in Subsection 6.4.2.

In Subsection 6.4.5, we prove that for $n, m \geq 2$, the triple $(3, n, m)$ does not impose θ-periodicity. We provide several (parametrized) examples which verify that for some specific values of n, m, the triple $(3, n, m)$ does not impose θ-periodicity. Our survey will expose complex behaviors of $(3, n, m)$ ExLS equations.

6.4.1 Problem setting for the ExLS equation $\ell = 4$

Taking the assumptions mentioned above into consideration, the problem which we are addressing is described as follows:

Problem 6.1. Let $u, v, w \in \Sigma^+$ and integers $n, m \geq 3$. Let $u_1, u_2, u_3, u_4 \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. Does the equation $u_1 u_2 u_3 u_4 = v_1 \cdots v_n w_1 \cdots w_m$ imply $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$ under all of the following conditions?

1. v and w are θ-primitive,

2. $|v_1 \cdots v_n| \geq |w_1 \cdots w_m|$,

3. $u_1 = u$, $v_1 = v$, and $w_m = w$,

4. $|v|, |w| < |u|$.

The condition 2 means that $2|u| \leq n|v|$. Besides, the condition 4 follows from the conditions 1 and 2 as shown in the next lemma.

Lemma 6.25. *Let $u, v, w \in \Sigma^+$ such that v, w are θ-primitive. If $u_1u_2u_3u_4 = v_1 \cdots v_n w_1 \cdots w_m$ for some $n, m \geq 3$, $u_1, u_2, u_3, u_4 \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$, then $|v| < |u|$ and $|w| < |u|$.*

Proof. Due to Condition 2, $|v_1 \cdots v_n| \geq |w_1 \cdots w_m|$. This means that $m|w| \leq 2|u|$, which in turn implies $|w| \leq \frac{2}{3}|u|$ because $m \geq 3$. Thus $|w| < |u|$.

Now suppose that the ExLS equation held with $|v| \geq |u|$. Then $v_1 \cdots v_n$ is a prefix of $u_1u_2u_3u_4$ of length at least $3|v| \geq 2|v| + |u|$, and hence $u, v \in \{t, \theta(t)\}^+$ for some θ-primitive word $t \in \Sigma^+$ due to Theorem 6.6. Unless $|v| = |u|$, we reach the contradiction that v would not be θ-primitive. Even if $|v| = |u|$, we have $u_4 = w_1 \cdots w_m$. Therefore $v_1 = u_1$ could not be θ-primitive. $\square$

The next lemma reduces the number of steps required to prove a positive answer to Problem 6.1.

Lemma 6.26. *Under the setting of Problem 6.1, if $u, v \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$, then $w \in \{t, \theta(t)\}^+$.*

In fact, we can say more strongly that if two of u, v, w are proved to be in $\{t, \theta(t)\}^+$ for some t, then the other one is also in this set.

First of all, we distinguish the case in which the existence of such t that $u, v, w \in \{t, \theta(t)\}^+$ is trivial due to the generalized Fine and Wilf theorem (Theorem 6.5).

Theorem 6.27. *Under the setting of Problem 6.1, if there exists an index i, $1 \leq i \leq n$, such that $u_1 u_2 = v_1 \cdots v_i$, then $u, v, w \in \{t, \theta(t)\}^+$ for some word $t \in \Sigma^+$.*

Proof. Since v is assumed to be θ-primitive, Theorem 6.5 implies $u \in \{v, \theta(v)\}^+$. Then $w \in \{v, \theta(v)\}^+$ due to Lemma 6.26 (in fact, $w \in \{v, \theta(v)\}$ because w is also assumed to be θ-primitive). $\square$

If a given $(4, n, m)$ ExLS equation satisfies the condition in Theorem 6.27, then we say that this equation is *trivial*. Before initiating our study on non-trivial ExLS equations, we provide one important condition which makes the equation trivial according to the generalized Fine and Wilf theorem (Theorem 6.6).

Proposition 6.28. *Under the setting of Problem 6.1, if $n|v| \geq 2|u| + |v|$, then the equation is trivial.*

Proof. We can employ Theorem 6.6 to obtain $u, v \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$. In fact, t is either v or $\theta(v)$ because v is assumed to be θ-primitive. Hence we can find such i stated in Theorem 6.27, and by definition this equation is trivial. $\square$

6.4.2 Non-trivial $(4, \geq 3, \geq 3)$ ExLS equations and related combinatorial results

Now we shift our attention to the non-trivial $(4, \geq 3, \geq 3)$ ExLS equation. What we will actually prove here is that under the setting of Problem 6.1, any *non-trivial* equation cannot hold. Along with Theorem 6.27, this implies that $(4, \geq 3, \geq 3)$ imposes θ-periodicity.

From this theorem and Proposition 6.28, the equation is *non-trivial* if and only if $(n-1)|v| < 2|u| < n|v|$. Thus, the next proposition, which was proposed in [3] to decrease the amount of case analyses for the $(5, \geq 3, \geq 3)$ ExLS equation, is still available for the investigation of non-trivial $(4, \geq 3, \geq 3)$ ExLS equations.

Proposition 6.29 ([3]). *Let $u, v \in \Sigma^+$ such that v is θ-primitive, $u_2, u_3 \in \{u, \theta(u)\}$, and $v_2, \ldots, v_n \in \{v, \theta(v)\}$ for some integer $n \geq 3$. If $vv_2 \cdots v_n \in \mathrm{Pref}(uu_2u_3)$ and $(n-1)|v| < 2|u| < n|v|$, then there are only two possible cases.*

1. *$u_2 = \theta(u)$: and $v_2 = \cdots = v_n = v$ with $u\theta(u) = (pq)^{n-1}p$ and $v = pq$ for some non-empty θ-palindromes p, q.*

2. *$u_2 = u$: n is even, $v_2 = \cdots = v_{n/2} = v$, and $v_{n/2+1} = \cdots = v_n = \theta(v)$ with $v = r(tr)^i(rt)^{i+j}r$ and $u = v^{n/2-1}r(tr)^i(rt)^j$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t such that rt is primitive.*

This proposition helps in proving that non-trivial $(4, \geq 3, \geq 3)$ ExLS equations verify the one more condition that $|v| \neq |w|$ as shown in the next proposition.

Proposition 6.30. *Non-trivial ExLS equations under the setting of Problem 6.1 imply* $|v| \neq |w|$.

Proof. Suppose that the equation were non-trivial with $|v| = |w|$. Combining $|v| = |w|$ and the non-trivial length condition together implies $m = n-1$ and furthermore the border between u_2 and u_3 splits v_n into exactly halves. Hence if $u_3 = \theta(u_2)$, then $v_n = x\theta(x)$ for some $x \in \Sigma^+$, contradicting the θ-primitivity of v. Besides, due to the condition 4 of Problem 6.1, if $u_4 = \theta(u_1)$, then $w = \theta(v)$, and hence $u_1u_2u_3u_4 \in \{v, \theta(v)\}^+$. Taking $(n-1)|v| < 2|u| < n|v|$ into account, this implies that v is not θ-primitive, raising a contradiction. Therefore, the only possible solutions verify $u_3 = u_2$ and $u_4 = u_1 = u$.

If $u_2 = u_3 = u$, then according to Proposition 6.29, n is even, and by substituting the representations of u and v given there into $u^4 = v^{n/2}\theta(v)^{n/2}w_1 \cdots w_m$, we obtain that $w_1 \cdots w_m = (tr)^j[r(tr)^i r(tr)^{i+j}]^{n/2-1}[r(tr)^{i+j}r(tr)^i]^{n/2-1}(rt)^j$, which is a θ-palindrome of even length. Since w is θ-primitive, m has to be even (Lemma 6.4). It is however impossible because $m = n-1$ and n is even.

If $u_2 = u_3 = \theta(u)$, then Proposition 6.29 gives $v = pq$ and $u_1u_2 = u\theta(u) = (pq)^{n-1}p$ for some θ-palindromes $p, q \in \Sigma^+$. Note that the left side of the ExLS equation is as long as its right side ($4|u| = n|v| + m|w| = (2n-1)|pq|$). Substituting $2|u| = (n-1)|pq| + |p|$ into this yields $|p| = |q|$ and it in turn implies that both p

and q are of even length. Let $p = p'\theta(p')$ and $q = q'\theta(q')$ for some $p', q' \in \Sigma^+$ of the same length. Then $u_1 = u$ ends with either $\theta(p')qp'$ or $\theta(q')pq'$, and so w_m is either of them. However, neither is θ-primitive. This contradiction proves that the equation is trivial. □

Supposing that some non-trivial $(4, \geq 3, \geq 3)$ ExLS equation held, the next claim would follow from this proposition. Although our conclusion in this section will prove that this claim cannot hold, the equation proposed there, $u_3u_4 = qw_1 \cdots w_m$, or more generally the relation $qw_1 \cdots w_m \in \{u, \theta(u)\}^{\geq 2}$ provides in its own right challenging themes.

Claim 6.31. *Under the setting of Problem 6.1, if the ExLS equation were non-trivial, then we would have $u_3u_4 = qw_1 \cdots w_m$ for some non-empty θ-palindrome q.*

Proof. According to the presentations of u and v given in Proposition 6.29, if $u_2 = \theta(u)$, then $u\theta(u)q = v^n$ and hence $u_3u_4 = qw_1 \cdots w_m$; otherwise, $uu[r(tr)^i]^2 = v^{n/2}\theta(v)^{n/2}$ so that $u_3u_4 = [r(tr)^i]^2 w_1 \cdots w_m$. Since q, r, t are θ-palindromes, this claim holds. □

As we shall see soon in Claim 6.33, the next lemma is of use when considering non-trivial ExLS equations with $u_3 \neq u_4$, that is, u_3u_4 being a θ-palindrome.

Lemma 6.32. *Let p, q be non-empty θ-palindromes and let w be a θ-primitive word.*

For some $k \geq 1$ and words $w_1, \ldots, w_k \in \{w, \theta(w)\}$, if $p = qw_1 \cdots w_k$ holds, then either $p, q \in \{w, \theta(w)\}^+$ or $w_1 = \cdots = w_k$.

Proof. First we prove that $q \in \mathrm{Suff}((w_1 \cdots w_k)^+)$. Since $w_1 \cdots w_k \in \mathrm{Suff}(p)$, p being a θ-palindrome implies $\theta(w_1 \cdots w_k) \in \mathrm{Pref}(p)$. Thus if $|q| \leq k|w|$, then $q \in \mathrm{Pref}(\theta(w_1 \cdots w_k))$, that is, $q \in \mathrm{Suff}(w_1 \cdots w_k)$ and we are done. Otherwise, $w_1 \cdots w_k \in \mathrm{Suff}(q)$ so that $(w_1 \cdots w_k)^2 \in \mathrm{Suff}(p)$. By repeating this process, eventually we will find some integer $i \geq 1$ such that $q \in \mathrm{Suff}((w_1 \cdots w_k)^i)$.

If $q \in \{w, \theta(w)\}^+$, then $p \in \{w, \theta(w)\}^+$. Otherwise, let $q = w' w_{j+1} \cdots w_k (w_1 \cdots w_k)^i$ for some $1 \leq j \leq k$ and $i \geq 0$, where w' is a non-empty proper suffix of w_j. Then, $p = w' w_{j+1} \cdots w_k (w_1 \cdots w_k)^{i+1}$ overlaps in a non-trivial way with $p = \theta(p) = (\theta(w_k) \cdots \theta(w_1))^{i+1} \theta(w_k) \cdots \theta(w_{j+1}) \theta(w')$, and Theorem 6.12 implies that $w_1 = \cdots = w_k$. $\square$

Claim 6.33. *Under the setting of Problem 6.1, if the ExLS equation were non-trivial and $u_3 \neq u_4$, then $w_1 = \cdots = w_m = w$ and $u_3 u_4 \in \mathrm{Suff}(w^+)$.*

Proof. We have $u_3 u_4 = x w_1 \cdots w_m$ for some non-empty θ-palindrome $x \in \Sigma^+$ due to Proposition 6.29. As suggested before, we can employ Lemma 6.32 to get either $x, u_3 u_4 \in \{w, \theta(w)\}^+$ or $w_1 = \cdots = w_m$. In the first case, Theorem 6.5 implies $u \in \{w, \theta(w)\}^+$ because w is assumed to be θ-primitive. Then the ExLS equation in turn implies that $v_1 \cdots v_n \in \{w, \theta(w)\}^+$ and hence $v \in \{w, \theta(w)\}$ for the same reason. As a result the equation would be trivial. Consequently $w_1 = \cdots = w_m$. $\square$

The main strategy used in the analyses of non-trivial ExLS equations is to split $w_1 \cdots w_m$ into smaller components which are still in $\{w, \theta(w)\}^+$, until we reach a contradiction. The split is mainly achieved by Propositions 6.16 and 6.20. Note that the word to which Proposition 6.20 is applied must be primitive. The next two lemmas work for this purpose in Subsection 6.4.3, but we provide them in more general form. An interesting point is that Lyndon and Schützenberger's original result (Theorem 6.1) plays an essential role in their proofs; hence for the ExLS equation.

Proposition 6.34. *Let $r, t \in \Sigma^+$ such that rt is primitive. For any $i \geq 0$, $j, k \geq 1$, and $n \geq 2$, $(tr)^j[(r(tr)^i)^n(tr)^j]^k$ is primitive.*

Proof. Suppose that the given word were not primitive; namely, for some $\ell \geq 2$ and a primitive word x, let $(tr)^j[(r(tr)^i)^n(tr)^j]^k = x^\ell$. Catenating $(r(tr)^i)^n$ to the left to the both sides of this equation gives $[(r(tr)^i)^n(tr)^j]^{k+1} = (r(tr)^i)^n x^\ell$. As $k \geq 1$ and $n, \ell \geq 2$, we can apply Theorem 6.1 to this equation to obtain $\rho((r(tr)^i)^n(tr)^j) = \rho(r(tr)^i) = x$. Using Lemma 6.9, one can obtain $\rho((tr)^j) = x$, and furthermore, $\rho(tr) = x$. Combining this with $\rho(r(tr)^i) = x$ gives us $\rho(r) = \rho(t)$ and hence rt would not be primitive, which contradicts the hypotheses. $\square$

Proposition 6.35. *Let $r, t \in \Sigma^+$ such that rt is primitive. For any $i \geq 0$, $j, k, m \geq 1$, $(tr)^j[(r(tr)^i)^m(tr)^j]^{k-1}(r(tr)^i)^{m-1}(rt)^j$ is primitive.*

Proof. Suppose that we had $(tr)^j[(r(tr)^i)^m(tr)^j]^{k-1}(r(tr)^i)^{m-1}(rt)^j = x^\ell$ for some primitive word x and $\ell \geq 2$. Catenating $(r(tr)^i)^{m+1}$ to the right to the both sides of this equation gives $[(tr)^j(r(tr)^i)^m]^{k+1} = x^\ell(r(tr)^i)^{m+1}$. Now as in the proof of Proposition 6.34, we reach the contradicting conclusion that rt is not primitive. $\square$

There are some results which can be used for the splitting strategy, once we apply Proposition 6.29 to non-trivial ExLS equations with $u_1 \neq u_2$, which will be considered in Subsection 6.4.4. As before, they are provided in more general form than required for the purpose.

Lemma 6.36. *Let $z, w \in \Sigma^+$ with $|z| < |w|$ and let p be a θ-palindrome. If $zp = w^n$ for some $n \geq 2$, then $z = \theta(z)$.*

Proof. Let $w = zy$ for some $y \in \Sigma^+$. Then $p = y(zy)^{n-1}$, from which we can obtain $y = \theta(y)$ and $z = \theta(z)$ because $p = \theta(p)$ and $n - 1 \geq 1$. $\square$

Proposition 6.37. *Let x be a θ-primitive word, $u \in \Sigma^+$, and q be a non-empty θ-palindrome. If for some $n \geq 2$ and $\ell \geq 1$, $u[\theta(u)q^nu]^\ell \in \{x, \theta(x)\}^{\geq 2}$, then $u, q \in \{x, \theta(x)\}^+$.*

Proof. Let $u[\theta(u)q^nu]^\ell = x_1 \cdots x_m$ for some $m \geq 2$ and $x_1, \ldots, x_m \in \{x, \theta(x)\}$. Let $u = x_1 \cdots x_{k-1}z_1$ and $[\theta(u)q^nu]^\ell = z_2x_{k+1} \cdots x_m$ for some $1 \leq k \leq m$ with $x_k = z_1z_2$ and $z_1 \neq \lambda$, i.e. $|z_2| < |x|$. If $z_2 = \lambda$, then $u, [\theta(u)q^nu]^\ell \in \{x, \theta(x)\}^+$. Lemma 6.10 implies $\theta(u)q^nu \in \{x, \theta(x)\}^+$ and the same lemma further gives $q^n \in \{x, \theta(x)\}^+$, that is, $q \in \{x, \theta(x)\}^+$.

Now we prove that z_2 cannot be non-empty. Without loss of generality, we assume $x_m = x$. So suppose $z_2 \neq \lambda$ $(0 < |z_1| < |x|)$. We can apply Lemma 6.24 to $z_1[\theta(u)q^n u]^\ell = x_k \cdots x_m$ to get $x_{k+1} = \cdots = x_m = x$ because $[\theta(u)q^n u]^\ell$ is a θ-palindrome and $|z_1| < |x|$. Thus if $|x| \leq |u|$, then $|z_2| < |u|$ and so $[\theta(u)q^n u]^\ell = z_2 x^{k-1}$ gives $x \in \mathrm{Suff}(u)$ and hence $\theta(x) \in \mathrm{Pref}(\theta(u))$. These further imply that $x \in \mathrm{Suff}(x_1 \cdots x_{k-1} z_1)$ and $\theta(x) \in \mathrm{Pref}(z_2 x_{k+1} \cdots x_m)$. Thus $x\theta(x)$ is a proper infix of $x_{k+1}x_k x_{k-1}$, which is in contradiction with the θ-primitivity of x by Theorem 6.12.

Therefore, $|x| > |u|$, which means $k = 1$, that is, we have $x_2 = \cdots = x_m = x$. Note that $x \neq \theta(x)$ must hold because of $z_2 x^{m-1}$ being a θ-palindrome, $0 < |z_2| < |x|$ and x is primitive (and cannot be a proper infix of its square). If $x_1 = \theta(x)$, then $u \in \mathrm{Pref}(\theta(x)) \cap \mathrm{Suff}(x)$ holds and so $u = \theta(u)$. Now Lemma 6.24 would imply $x_1 = x$, which contradicts $x \neq \theta(x)$. Otherwise $(x_1 = x)$, $u[\theta(u)q^n u]^\ell = x^m$ and from this Lemma 6.36 derives $u = \theta(u)$. Then we have $u(uq^n u)^\ell = x^m$; in other words, $(uq^n u)^{\ell+1}$ and x^m share a suffix of length at least $\eta = \max(m|x|, \ell|uq^n u|)$. If $\ell \geq 2$, then $\eta \geq |x| + |uq^n u|$, and the Fine and Wilf theorem implies $\rho(uq^n u) = x$. With $u(uq^n u)^\ell = x^m$, this implies $\rho(u) = x$. However, this contradicts $|u| < |x|$. If $\ell = 1$, then $uuq^n u = x^m$. Using cyclic permutation, we obtain $u^3 q^n = x'^m$, where x' is a conjugate of x. This is of the form of LS equation, and Theorem 6.1 concludes $\rho(u) = \rho(q) = x'$. Now we reached the same contradiction because $|x'| = |x|$. □

Lemma 6.38. *Let w be a θ-primitive word, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$ for some $m \geq 2$. Let $u, q \in \Sigma^+$ such that q is a θ-palindrome with $|q| < |u|$. If $u^2 = qw_1 \cdots w_m$,* *then either $u, q \in \{w, \theta(w)\}^+$ or $u = qr$ for some non-empty θ-palindrome r.*

Proof. It is trivial that the case $u, q \in \{w, \theta(w)\}^+$ is possible. Hence assume that $u, q \notin \{w, \theta(w)\}^+$. Without loss of generality, we can also assume that $w_m = w$. Let $u = qr$ for some $r \in \Sigma^+$. Then $rqr = w_1 \cdots w_m$. We prove that r is a θ-palindrome. Let $r = w_1 \cdots w_{k-1} z_1 = z_2 w_{m-k+2} \cdots w_m$ for some $k \geq 1$, where $z_1 \in \mathrm{Pref}(w_k)$ and $z_2 \in \mathrm{Suff}(w_{m-k+1})$ with $|z_1| = |z_2| < |w|$. If $z_1 = \lambda$, then $r \in \{w, \theta(w)\}^+$ and then $rqr = w_m \cdots w_1$ implies $q \in \{w, \theta(w)\}^+$ by Lemma 6.10, but this contradicts the assumption. Thus $z_1 \neq \lambda$. Then we have two cases, $k \geq 2$ and $k = 1$. Lemma 6.11 (for $k = 2$) or Theorem 6.12 (for $k \geq 3$) works to give $w_1 = \cdots = w_{k-1} = \theta(w)$ and $w_{m-k+2} = \cdots = w_m = w$. Thus, $z_2 = \theta(z_1)$ and hence $r = \theta(r)$. Even for $k = 1$, if $w_1 \neq w_m$, then $r \in \mathrm{Pref}(\theta(w)) \cap \mathrm{Suff}(w)$ so that $r = \theta(r)$. Otherwise $w = rq_p = q_s r$ for some $q_p \in \mathrm{Pref}(q)$ and $q_s \in \mathrm{Suff}(q)$. Since $q = \theta(q)$, $q_s = \theta(q_p)$ so that we have $rq_p = \theta(q_p)r$. According to Proposition 6.7, $r = \theta(r)$. □

Proposition 6.39. *Let w be a θ-primitive word, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$ for some odd integer $m \geq 3$. Let $u, q \in \Sigma^+$ such that q is a θ-palindrome with $|q| < |u|$. If $u^2 = qw_1 \cdots w_m$, then $w = \theta(w)$. If additionally $|u| \geq 2|q|$ holds, then $\rho(u) = \rho(q) = w$.*

Proof. Lemma 6.38 implies that either $q, u \in \{w, \theta(w)\}^+$ or $u = qr$ for some non-empty θ-palindrome r. In the former case, let $u \in \{w, \theta(w)\}^k$ for some $k \geq 1$ and we can see $q \in \{w, \theta(w)\}^{2k-m}$ and $2k - m$ is odd because m is odd. Then $q = \theta(q)$

implies $w = \theta(w)$, and hence $u, q \in w^+$. In the latter case, we have $rqr = w_1 \cdots w_m$. This implies $w_{(m+1)/2} = \theta(w_{(m+1)/2})$ (i.e, $w = \theta(w)$) because rqr is a θ-palindrome and m is odd.

Now we consider the additional hypothesis $|u| \geq 2|q|$. Since $2|u| = |q| + m|w|$, $|u| = (|q| + m|w|)/2 \geq 2|q|$, which leads to $|q| \leq \frac{1}{3}m|w|$. As seen above, $rqr = w^m$, hence $|r| = (m|w| - |q|)/2 \geq \frac{1}{3}m|w| \geq |w|$ as $m \geq 3$. With this, the equation $rqr = w^m$ gives $r = w^k w'_p = w'_s w^k$ for some $k \geq 1$, $w'_p \in \mathrm{Pref}(w)$, and $w'_s \in \mathrm{Suff}(w)$. Since w is primitive, w'_p and w'_s have to be empty. Consequently $\rho(r) = \rho(q) = w$ and hence $\rho(u) = w$ by using Lemma 6.10. □

6.4.3 ExLS equation of the form $u^2 u_3 u_4 = v_1 \cdots v_n w_1 \cdots w_m$

In this subsection, we prove that an ExLS equation of the form $u^2 u_3 u_4 = v_1 \cdots v_n w_1 \cdots w_m$ implies that $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$. We have already seen that for this purpose it suffices to show that any non-trivial equation of this form cannot hold. Recall that we assumed $u_1 = u$, $v_1 = v$, and $w_m = w$, and that Proposition 6.30 allows us to assume $|v| \neq |w|$.

We can apply Proposition 6.29 to the non-trivial equation to obtain that n is an even integer except 2, $v_1 = \cdots = v_{n/2} = v$ and $v_{n/2+1} = \cdots = v_n = \theta(v)$ (i.e., $v_1 \cdots v_n$ is a θ-palindrome), $u = [r(tr)^i r(tr)^{i+j}]^{n/2-1} r(tr)^i (rt)^j$, and $v = r(tr)^i r(tr)^{i+j}$ for some $i \geq 0$, $j \geq 1$, and non-empty θ-palindromes r, t such that rt is primitive. Actually rt has to be θ-primitive due to Corollary 6.22 because $v \in \{r, t\}^+$ is assumed to be θ-primitive. Let us now study all possible values of u_3u_4.

Proposition 6.40. *Under the setting of Problem 6.1, if* $u_1u_2u_3u_4 = u^4$*, then* $u, v, w \in \{t, \theta(t)\}^+$ *for some* $t \in \Sigma^+$.

Proof. According to the representations of u and v in terms of r and t, we obtain

$$w_1 \cdots w_m = (tr)^j[(r(tr)^i)^2(tr)^j]^{n/2-1}[(rt)^j(r(tr)^i)^2]^{n/2-1}(rt)^j \ .$$

This expression is a θ-palindrome of even length and hence m has to be even (Lemma 6.4). Therefore, $w_1 \cdots w_{m/2} = [(tr)^j(r(tr)^i)^2]^{n/2-1}(tr)^j$, and this was proved to be primitive in Proposition 6.34. Moreover, its right hand side is the catenation of two θ-palindromes $p_1 = (tr)^j[r(tr)^ir(tr)^{i+j}]^{n/2-2}r(tr)^i(rt)^j$ and $p_2 = r(tr)^i$. Proposition 6.20 gives $p_2 = r(tr)^i \in \{w, \theta(w)\}^+$. Furthermore, applying Proposition 6.16 to $p_1p_2 = (tr)^j[r(tr)^ir(tr)^{i+j}]^{n/2-2}r(tr)^i \cdot p_2 \cdot (tr)^j$ gives $(tr)^j \in \{w, \theta(w)\}^+$. Finally Lemma 6.10 derives $r, t \in \{w, \theta(w)\}^+$ from $r(tr)^i, (tr)^j \in \{w, \theta(w)\}^+$, but this contradicts the θ-primitivity of rt. As a result, there are no solutions to the non-trivial equation. $\square$

Proposition 6.41. *Under the setting of Problem 6.1, if* $u_1u_2u_3u_4 = u^3\theta(u)$*, then* $u, v, w \in \{t, \theta(t)\}^+$ *for some* $t \in \Sigma^+$.

Proof. Since u_4 is $\theta(u)$ instead of u, we have $w_1 \cdots w_m = x^2(r(tr)^i)^2$, where $x = (tr)^j[(r(tr)^i)^2(rt)^j]^{n/2-1}r(tr)^i(rt)^j$. Claim 6.33 gives that $w_1 = \cdots = w_m = w$, and hence $w^m = x^2(r(tr)^i)^2$. This is a classical LS equation; thus Theorem 6.1 is applicable to conclude that $\rho(x) = \rho(r(tr)^i)$. However, this contradicts the primitivity of x obtained in Proposition 6.35 because $|x| > |r(tr)^i|$. □

Proposition 6.42. *Under the setting of Problem 6.1, if $u_1u_2u_3u_4 = u^2\theta(u)u$, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. Since $u_3 \neq u_4$, $w_1 = \cdots = w_m = w$ due to Claim 6.33. Using the representations of u and v by r and t, we can see that $u_3u_4 = \theta(u)u$ is equal to both sides of the following equation:

$$(tr)^j r(tr)^i[(rt)^j(r(tr)^i)^2]^{n/2-1}[(r(tr)^i)^2(tr)^j]^{n/2-1}r(tr)^i(rt)^j = (r(tr)^i)^2 w^m \ .$$

By catenating $(r(tr)^i)^4$ to the left of both sides, we get $(r(tr)^i)^6 w^m = x^2$, where $x = (r(tr)^i)^2[(r(tr)^i)^2(tr)^j]^{n/2-1}r(tr)^i(rt)^j$. Then, Theorem 6.1 implies that $\rho(x) = \rho(r(tr)^i) = w$. Since x contains $r(tr)^i$ as its infix, the share of primitive root between x and $r(tr)^i$ gives $\rho(r(tr)^i) = \rho((rt)^j)$. We deduce from this using Lemma 6.9 that rt would not be primitive, which contradicts our hypothesis. □

Proposition 6.43. *Under the setting of Problem 6.1, if $u_1u_2u_3u_4 = u^2\theta(u)^2$, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. Recall that $v_1 \cdots v_n$ is a θ-palindrome. Since $u^2\theta(u)^2$ is a θ-palindrome, $\theta(w_1 \cdots w_m)$ is one of its prefixes and the assumption $|w_1 \cdots w_m| < |v_1 \cdots v_n|$ implies that $\theta(w_1 \cdots w_m) \in \mathrm{Pref}(v_1 \cdots v_n)$. Hence $w_1 \cdots w_m \in \mathrm{Suff}(v_1 \cdots v_n)$ and now we have $(w_1 \cdots w_m)^2 \in \mathrm{Suff}(u^2\theta(u)^2)$.

We prove that this suffix is long enough to apply the extended Fine and Wilf theorem. Since $(n-1)|v| < 2|u|$ and $n \geq 4$, we have $|v| < \frac{2}{3}|u|$ and, in turn, $n|v| < 2|u| + \frac{2}{3}|u| = \frac{8}{3}|u|$. From this we obtain $m|w| > \frac{4}{3}|u|$. Then, $2m|w| - (|w| + 2|u|) > (2m-1)|w| - \frac{3}{2}m|w| = (\frac{1}{2}m - 1)|w| > 0$ since $m \geq 3$. Thus, $u^2\theta(u)^2$ and $(w_m \cdots w_1)^2$ share a suffix of length at least $2|u|+|w|$ and Theorem 6.6 concludes that $u \in \{w, \theta(w)\}^+$ because w is θ-primitive. Now it is clear that also $v \in \{w, \theta(w)\}^+$, but in fact $v \in \{w, \theta(w)\}$ must hold because v is also θ-primitive. However this contradicts the assumption that $|v| \neq |w|$. $\square$

6.4.4 ExLS equation of the form $u\theta(u)u_3u_4 = v_1 \cdots v_n w_1 \cdots w_m$

Note that in the following propositions, we consider only the non-trivial equations; hence Proposition 6.30 allows to assume $|v| \neq |w|$.

Using Proposition 6.29, $u\theta(u) = (pq)^{n-1}p$ and $v_1 = \cdots = v_n = v = pq$ for some non-empty θ-palindromes p, q. Unlike the case considered before, in the current case n can be odd. In fact, if n is odd, then $u = (pq)^{(n-1)/2}y$, where $p = y\theta(y)$ for some $y \in \Sigma^+$; while if n is even, then $u = (pq)^{n/2-1}px$, where $q = x\theta(x)$ for some $x \in \Sigma^+$. Again, we consider the four cases associated with the four possible values of u_3u_4. The last two, $u_3 = u_4 = u$ and $u_3 = u_4 = \theta(u)$, are merged and studied in two separate propositions depending on the parity of m instead.

Proposition 6.44. *Under the setting of Problem 6.1, if* $u_1u_2u_3u_4 = u\theta(u)u\theta(u)$*, then* $u, v, w \in \{t, \theta(t)\}^+$ *for some* $t \in \Sigma^+$.

Proof. In this setting, $u_3u_4 = u\theta(u) = qw_1 \cdots w_m$. Since both $u\theta(u)$ and q are θ-palindromes, we can employ Claim 6.33 to obtain $w_1 = \cdots = w_m = w$. Now the equation turns into the LS equation $(u\theta(u))^2 = v^n w^m$, and hence $\rho(v) = \rho(w)$ due to Theorem 6.1. Both v and w being primitive, this contradicts the assumption $|v| \neq |w|$ and consequently the existence of non-trivial solutions. □

Proposition 6.45. *Under the setting of Problem 6.1, if* $u_1u_2u_3u_4 = u\theta(u)\theta(u)u$*, then* $u, v, w \in \{t, \theta(t)\}^+$ *for some* $t \in \Sigma^+$.

Proof. Recall that $u\theta(u) = (pq)^{n-1}p$. Claim 6.33 implies that $\theta(u)u = qw^m$ with $q = w'w^{k-1}$ for some $1 \leq k \leq m$ and a non-empty proper suffix w' of w.

Case 1 (n is odd): Then we have $\theta(u)u = qw^m = x_s x$, where $x_s = \theta(y)q(pq)^{(n-1)/2-1}y$ and $x = \theta(y)(pq)^{(n-1)/2}y$; note that $x_s \in \mathrm{Suff}(x)$. One can easily calculate that $|w| = \frac{1}{m}[n|p| + (n-2)|q|]$ and $|x_s| = \frac{1}{2}(n-1)(|p| + |q|)$, and hence $|x_s| - |w| = \frac{(m-2)(n-1)-2}{2m}|p| + \frac{(m-2)(n-1)+2}{2m}|q|$, which is positive because $n, m \geq 3$. Thus we can say that x^2 and w^{m+k} share a prefix of length at least $|x| + |w|$ so that by the Fine and Wilf theorem, $\rho(x) = \rho(w) = w$. Starting from $\theta(y)yqw^m = \theta(y)yx_s x = x^2$, we can verify that $2|x| - m|w| = |pq|$, that is, $|pq|$ is a multiple of $|w|$. The suffix of x of length $|pq|$ is $\theta(y)qy$, which is w^j for some $j \geq 2$ because $|pq| = |v| \neq |w|$. Therefore, this conjugate of v is not primitive, either. This is a contradiction with the θ-primitivity of v.

Case 2 (n is even): In this case, $u = (pq)^{n/2-1}px$ for some $x \in \Sigma^+$ such that $q = x\theta(x)$. Substituting this into $\theta(u)u = qw^m$ gives

$$[\theta(x)px]^{n/2-1}\theta(x)p^2x[\theta(x)px]^{n/2-1} = x\theta(x)w^m. \tag{6.2}$$

From this equation, we can obtain $x = \theta(x)$ and hence $px = xz$ for some $z \in \Sigma^+$. If $|x| \geq |p|$, then Lemma 6.2 implies $\rho(x) = \rho(p)$ so that $v = pq = px^2$ would not be primitive. Hence $|x| < |p|$ must hold and under this condition, the solution of $px = xz$ is given by $p = xy$ and $z = yx$ for some $y \in \Sigma^+$. Since $p = \theta(p)$, we have $p = xy = \theta(y)x$. Proposition 6.7 gives $x = r(tr)^i$ and $y = (tr)^j$ for some $i \geq 0$, $j \geq 1$, and θ-palindromes r, t such that rt is primitive. Both of r and t should be non-empty; otherwise, $\rho(p) = \rho(x)$ and $v = pq = px^2$ would not be primitive. Substituting these into Eq. (6.2) yields the following equation.

$$\big[(tr)^j r(tr)^i[r(tr)^i r(tr)^{i+j} r(tr)^i]^{n/2-1}\big]^2 = w^m.$$

Since w is θ-primitive, this equation means that m has to be even. Then $w^{m/2} = (tr)^j r(tr)^i[r(tr)^i r(tr)^{i+j} r(tr)^i]^{n/2-1}$. By catenating $(r(tr)^i)^2$ from the left to the both sides of this equation, we obtain an LS equation $[r(tr)^i]^2 w^{m/2} = [r(tr)^i r(tr)^{i+j} r(tr)^i]^{n/2}$. Theorem 6.1 gives $\rho(r(tr)^i) = \rho(r(tr)^i r(tr)^{i+j} r(tr)^i)$ and Lemma 6.9 reduces it to

$\rho(r) = \rho(t)$, but this contradicts the primitivity of $pq = r(tr)^{i+j}(r(tr)^i)^2$. □

Proposition 6.46. *Under the setting of Problem 6.1, if $u_1u_2 = u\theta(u)$, $u_3 = u_4$, and m is odd, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. We have $u_3u_4 = qw_1 \cdots w_m$. Since $u_3 = u_4$ and $|q| < |u|$, we can employ Proposition 6.39 to obtain $w = \theta(w)$. Moreover, when $n \geq 5$, we have $|u| \geq 2|q|$ and the proposition also gives $\rho(u_3) = \rho(q) = w$. Since $w = \theta(w)$, we can see that $\rho(u) = w$. Then $\rho(p) = w$ because $\rho(u) = \rho(q) = w$ and $pq \in \mathrm{Pref}(u)$. However, $\rho(p) = \rho(q)$ means that $v = pq$ would not be even primitive. Therefore in the following let n be either 3 or 4.

First we consider the case when $u_3 = u$. Then we have either $(pqy)^2 = qw^m$ (when $n = 3$) where $p = y\theta(y)$, or $(pqpx)^2 = qw^m$ (when $n = 4$) where $q = x\theta(x)$, for some $x, y \in \Sigma^+$. In both cases, if $|p| \leq |q|$, Lemma 6.2 can be applied and we have $\rho(p) = \rho(q)$, so $v = pq$ would not be even primitive. Hence $|p| > |q|$ must hold, but then $|u| \geq 2|q|$ and then Proposition 6.39 implies $\rho(p) = \rho(q)$.

Next we consider the case when $u_3 = \theta(u)$ and $n = 3$. Then $\theta(u) = \theta(y)qp$ so that $\theta(y)qp\theta(y)qp = qw^m$. Let $\theta(y)q = qz$ for some z with $|y| = |z|$. Using $pq = y\theta(y)q = yqz$, from $\theta(y)qp\theta(y)qp = qw^m$ we can obtain $zyqzzy\theta(y) = w^m$. Since $w = \theta(w)$, this equation gives $z = y = \theta(y)$. Then $\theta(y)q = qz$ turns into $yq = qy$ and hence $\rho(y) = \rho(q)$ by Theorem 6.3. This however implies that $v = y\theta(y)q$ would not be θ-primitive.

Finally we consider the case when $u_3 = \theta(u)$ and $n = 4$. Then we have $[\theta(x)pqp]^2 = qw^m$, which gives $x = \theta(x)$ because $q = x\theta(x)$. Then $\theta(u)^2 = x^2w^m$, which is an LS equation and Theorem 6.1 implies $\rho(\theta(u)) = \rho(x) = w$. However since $x^2p = qp \in \mathrm{Suff}(\theta(u))$, we also get $\rho(p) = w$ (otherwise w would be a proper infix of its square in x^2). This leads to the usual contradiction that $v = px^2$ would not be primitive. $\square$

Proposition 6.47. *Under the setting of Problem 6.1, if $u_1u_2 = u\theta(u)$, $u_3 = u_4$, and m is even, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. As before we consider only non-trivial equation so that we have $u_3u_4 = qw_1 \cdots w_m$ and $|v| \neq |w|$. Lemma 6.38 gives two cases, but actually it suffices to consider the case when $u = qr$ for some non-empty θ-palindrome r.

First we consider the case when $u_3 = u$ and n is even. Then $[(pq)^{n/2-1}px]^2 = qw_1 \cdots w_m$, where $q = x\theta(x)$ for some $x \in \Sigma^+$. If $|p| \leq |q|$, then $pq = qp$ and v would not be even primitive. Hence let $p = qz_1$ for some $z_1 \in \Sigma^+$. Then $r = z_1x\theta(x)(pq)^{n/2-2}x\theta(x)z_1x$. Since $r = \theta(r)$, this equation gives $z_1x = \theta(z_1x)$ and $x = \theta(x)$. Thus we have $z_1x = x\theta(z_1)$ and $p = x^2z_1 = \theta(z_1)x^2$. Then $x^3z_1 = x\theta(z_1)x^2 = z_1x^3$ so that $\rho(x) = \rho(z_1)$ by Theorem 6.3. However, this result contradicts the primitivity of $v = pq = x^2z_1x^2$.

The second case is when $u_3 = u$ an n is odd. We have $[(pq)^{(n-1)/2}y]^2 = qw_1 \cdots w_m$, where $p = y\theta(y)$. From this equation, q is of even length so let $q = x\theta(x)$. If $|p| \leq |q|$, then we can apply Lemma 6.2 to the equation above to prove that $\rho(p) = \rho(q)$, which contradicts the primitivity of v. Thus we can let $y = xz_2$ for some $z_2 \in \Sigma^+$. Then $[(xz_2\theta(z_2)\theta(x)x\theta(x))^{(n-1)/2}xz_2]^2 = x\theta(x)w_1 \cdots w_m$. We can easily check that $w_{m/2+1} \cdots w_m = z_2[\theta(z_2)\theta(x)x\theta(x)xz_2]^{(n-1)/2}$. According to Proposition 6.37, we can deduce from this that $z_2, \theta(x)x \in \{w, \theta(w)\}^+$ and this further implies $x \in \{w, \theta(w)\}^+$. However then $v = pq = xz_2\theta(z_2)\theta(x)x\theta(x)$ would not be θ-primitive.

Thirdly we consider the case when $u_3 = \theta(u)$ and n is even. We have $[\theta(x)p(qp)^{n/2-1}]^2 = x\theta(x)w_1 \cdots w_m$, and this equation immediately gives $x = \theta(x)$. Then $p(qp)^{n/2-1}xp(qp)^{n/2-1} = xw_1 \cdots w_m$. Since the left-hand side and x are θ-palindromes, we have either $x \in \{w, \theta(w)\}^+$ or $w_1 = \cdots = w_m = w$ by Lemma 6.32. In the former case, $\theta(u)^2 = x^2w_1 \cdots w_m \in \{w, \theta(w)\}^+$ and hence $\theta(u), u \in \{w, \theta(w)\}^+$ (Lemma 6.10). Then $v^n = u\theta(u)x\theta(x) \in \{w, \theta(w)\}^+$, and hence $v \in \{w, \theta(w)\}$ because of Lemma 6.10 and the θ-primitivity of v, w. However, this contradicts the assumption $|v| \neq |w|$. In the latter case, we have $\theta(u)^2 = x^2w^m$ and hence $\rho(\theta(u)) = \rho(x) = w$ (Theorem 6.1). However since $qp = x^2p \in \mathrm{Suff}(\theta(u))$, we reach the contradictory result $\rho(p) = w$.

The final case is when $u_3 = \theta(u)$ and n is odd. Then $[\theta(y)(qp)^{(n-1)/2}]^2 = qw_1 \cdots w_m$, where $p = y\theta(y)$ for some $y \in \Sigma^+$. Let $\theta(y)q = qz_4$ for some z_4 with $|y| = |z_4|$. Then $r = z_4(y\theta(y)q)^{(n-1)/2}y\theta(y)$, which is a θ-palindrome so that $z_4 = y = \theta(y)$. Now we can transform $\theta(y)q = qz_4$ into $yq = qy$ and hence $\rho(y) = \rho(q)$ (Theorem 6.3). However, then $v = y\theta(y)q$ would not be θ-primitive. □

Combining the results obtained in this section, we can give a positive answer to Problem 6.1. Furthermore, with the result proved in [3] (also see Table 6.1), this positive answer concludes the following theorem, the strongest positive result we obtain on the ExLS equation.

Theorem 6.48. *Let $u, v, w \in \Sigma^+$ and let $u_1, \ldots, u_\ell \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. For $\ell \geq 4$ and $n, m \geq 3$, the equation $u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m$ implies $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

6.4.5 The case $\ell \leq 3$ of the ExLS equation

We conclude this section with some examples which prove that an extended Lyndon-Schützenberger theorem cannot be stated for $\ell = 2$, and for some particular cases when $\ell = 3$.

Example 18. Let $\Sigma = \{a, b\}$ and θ be an antimorphic involutions on Σ^* defined as $\theta(a) = a$ and $\theta(b) = b$. Let $v = a^{2m}b^2$ and $w = aa$ (i.e., $w = \theta(w)$) for some $m \geq 1$. Then $v^n w^m = (a^{2m}b^2)^n a^{2m}$. By letting either $u = (a^{2m}b^2)^{n/2}a^m$ if n is even or $u = (a^{2m}b^2)^{(n-1)/2}a^{2m}b$ otherwise, we have $u\theta(u) = v^n w^m$. Nevertheless, there cannot exist a word t such that $u, v, w \in \{t, \theta(t)\}^+$ because v contains b, while w does not. In conclusion, for arbitrary $n, m \geq 2$, $(2, n, m)$ does not impose θ-periodicity.

Next we examine briefly the $(3, n, m)$ ExLS equation. The actual problem which we address is formalized as follows:

Problem 6.2. Let $u, v, w \in \Sigma^+$ and integers $n, m \geq 3$. Then, let $u_1, u_2, u_3 \in \{u, \theta(u)\}$, $v_1, \ldots, v_n \in \{v, \theta(v)\}$, and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. Does the equation $u_1u_2u_3 = v_1 \cdots v_n w_1 \cdots w_m$ imply $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$ under all of the following conditions?

1. v and w are θ-primitive,

2. $|v_1 \cdots v_n| \geq |w_1 \cdots w_m|$,

3. $u_1 = u$, $v_1 = v$, and $w_m = w$.

As shown from now by examples, the general answer is "No". More significant is the fact that depending on the values of variables u_2, u_3 and on the lengths of $v_1 \cdots v_n$ and $w_1 \cdots w_m$, the $(3, n, m)$ ExLS equation exhibits very complicated behavior.

First we present a parameterized example to show that for arbitrary $m \geq 2$, $(3, 3, m)$ does not impose θ-periodicity.

Example 19. Let $\Sigma = \{a, b\}$ and θ be the mirror image over Σ^*. For $u = (abb)^{2m-1}ab$, $v = (abb)^{m-1}ab$, and $w = (bba)^3$, we have $u^2\theta(u) = v\theta(v)^2w^m$ for any $m \geq 2$. Nevertheless, there does not exist a word $t \in \Sigma^+$ satisfying $u, v, w \in \{t, \theta(t)\}^+$.

In this example, the border between $v\theta(v)^2$ and w^m is located at u_2. Intriguingly, as long as $u_1u_2u_3 = uu\theta(u)$ we cannot shift the border to u_3 without imposing $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.

Proposition 6.49. *For any $n, m \geq 3$, if $uu\theta(u) = v_1 \cdots v_n w_1 \cdots w_m$ and $n|v| > 2|u|$, then $u, v, w \in \{t, \theta(t)\}^+$ for some $t \in \Sigma^+$.*

Proof. It suffices to consider the case when $(n-1)|v| < 2|u| < n|v|$, otherwise Theorem 6.6 applies. As done in the analyses on the ExLS equation with $\ell = 4$, we can assume that both v and w are θ-primitive. Then, using Proposition 6.29, we obtain that n is even, $u = [r(tr)^i r(tr)^i (tr)^j]^{n/2-1} r(tr)^i (rt)^j$ and $v = r(tr)^i r(tr)^i (tr)^j$ for some $i \geq 0$, $j \geq 1$, and two non-empty θ-palindromes r, t such that rt is primitive. Moreover, $\theta(u) = (tr)^j r(tr)^i [(rt)^j r(tr)^i r(tr)^i]^{n/2-1} = r(tr)^i r(tr)^i w_1 \cdots w_m$. Hence if $i \geq 1$, then $tr = rt$, which contradicts the primitivity of rt (Theorem 6.3). Thus we have

$$(tr)^j r[(rt)^j r^2]^{n/2-1} = r^2 w_1 \cdots w_m. \tag{6.3}$$

If $|t| \leq |r|$, then $t \in \mathrm{Pref}(r)$ from which $rt \in \mathrm{Pref}(r^2 w_1 \cdots w_m)$, and finally $rt = tr$, contradicting the primitivity of rt again. If $|r| < |t| \leq 2|r|$, then we can write $rrs = tr$ for some $s \in \Sigma^+$ such that $|r| + |s| = |t|$. Since $s \in \mathrm{Suff}(r)$ and r is a θ-palindrome, $\theta(s) \in \mathrm{Pref}(r)$, i.e., $r = \theta(s) r_1$ for some $r_1 \in \Sigma^+$. Then, $rrs = r\theta(s) r_1 s = tr$, so $r\theta(s) = t$ because their length is the same. Since $\theta(s) \in \mathrm{Suff}(t)$ and t is a θ-palindrome, it holds that $s \in \mathrm{Pref}(t)$ and $rrs \in \mathrm{Pref}(rrt)$. Therefore, rrt and tr share a prefix of length $|t| + |r|$ so that Theorem 6.3 concludes that $\rho(r) = \rho(t)$, contradicting the primitivity of rt.

Thus both $i = 0$ and $|t| > 2|r|$ must hold. Eq. (6.3) implies that $r^2 \in \mathrm{Pref}(t)$, that is, $r^2 \in \mathrm{Suff}(t)$ (t is a θ-palindrome), and hence $r^4 \in \mathrm{Suff}((rt)^j r^2)$. So we can let $r^4 = z_1 w_{k+1} \cdots w_m$ for some $k \geq 1$ and $z_1 \in \mathrm{Suff}(w_k)$. If $z_1 = \lambda$, then this equation gives $r \in \{w, \theta(w)\}^+$ because w is assumed to be θ-primitive due to Theorem 6.5. Then Eq. (6.3) means $(tr)^j r[(rt)^j r^2]^{n/2-1} \in \{w, \theta(w)\}^+$. Using Proposition 6.16, we obtain $t \in \{w, \theta(w)\}^+$, but this contradicts the θ-primitivity of v. Otherwise, catenating r^2 from the left to the both sides of Eq. (6.3) gives us $r[(rt)^j r^2]^{n/2} = z_1 w_{k+1} \cdots w_m w_1 \cdots w_m$. Note that the left hand side of this equation is a θ-palindrome so that Lemma 6.24 implies $w_1 = \cdots = w_m = w$. Now catenating r in the same way to Eq. (6.3) gives $[(rt)^j r^2]^{n/2} = r^3 w^m$. This is in the form of LS equation and Theorem 6.1 implies $\rho((rt)^j r^2) = \rho(r) = w$ because w is primitive. From this we further deduce that $\rho(t) = w$. However, then rt would not be primitive.

□

Once we change $u_1 u_2 u_3$ from $u^2\theta(u)$ to $u\theta(u)^2$, it becomes possible to construct a parameterized example for $(3, 3, m)$ with the border between $v_1 \cdots v_n$ and $w_1 \cdots w_m$ on u_3, though it works only when m is a multiple of 3.

Example 20. Let $\Sigma = \{a, b\}$ and θ be the mirror image over Σ^*. For $i, j \geq 0$, let $u = (ab)^{i+j+1}(ba)^{2i+2j+2}b(ab)^j$, $v = (ab)^{i+j+1}(ba)^{i+2j+1}b$, and $w = ab$. Then $u\theta(u)^2 = v^3 w^{2(i+j+1)}\theta(w)^{i+j+1}$, but we cannot find such t that $u, v, w \in \{t, \theta(t)\}^+$.

Next we increase n to 4, and prove that still we can construct a parameterized example of the $(3, 4, 2i)$ ExLS equation.

Example 21. Let $\Sigma = \{a, b\}$ and θ be the mirror image over Σ. For $i \geq 1$, let $u = a^4(ba^3)^i(a^3b)^i$, $v = a^4(ba^3)^{i-1}ba^2$, and $w = ba^3$. Then we have $u^3 = v^2\theta(v)^2 w^i\theta(w)^i$,

l	n	m	θ-periodicity	
≥ 4	≥ 3	≥ 3	YES	Theorem 6.48
3	≥ 5	≥ 5	?	
3	4	odd	?	
3	4	even	NO	Example 21
3	3	≥ 3	NO	Example 19
one of them is 2			NO	Example 18

Table 6.2: Updated summary on the results regarding the extended Lyndon-Schützenberger equation

but there does not exist a word $t \in \Sigma^+$ satisfying $u, v, w \in \{t, \theta(t)\}^+$.

The cases $(3, n, m)$ when $n = 4$ and m is odd, as well as when $m, n \geq 5$, remain open.

6.5 Conclusion

In this paper, we proved several consequences of the overlap between pseudo-primitive words. They made it possible to prove that, for a given antimorphic involution θ and words $u, v, w \in \Sigma^+$, if $\ell \geq 4$ and $n, m \geq 3$, then the ExLS equation $u_1 \cdots u_\ell = v_1 \cdots v_n w_1 \cdots w_m$ implies that $u, v, w \in \{t, \theta(t)\}^+$ for some t. This is the strongest result obtained so far on the ExLS equation. Our case analyses on $(3, \geq 3, \geq 3)$ ExLS equations demonstrated that these tools may not be sufficient to provide a complete characterization of ExLS equations. Further investigation on the overlaps of θ-primitive words, reduction schemes from ExLS equations to LS equations, and the weak defect effect seems promising and required to fill the gap in Table 6.2.

Acknowledgments

This research was supported by Natural Sciences and Engineering Research Council of Canada Discovery Grant R2824A01, and Canada Research Chair Award to L.K.

Bibliography

[1] K. I. Appel and F. M. Djorup. On the equation $z_1^n z_2^n \cdots z_k^n = y^n$ in a free semigroup. *Transactions of the American Mathematical Society*, 134:461–470, 1968.

[2] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, pages 329–438. Springer-Verlag, Berlin-Heidelberg-New York, 1997.

[3] E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schützenberger result to pseudoperiodic words. In V. Diekert and D. Nowotka, editors, *Proc. DLT09*, volume 5583 of *Lecture Notes in Computer Science*, pages 183–194, Berlin, 2009. Springer-Verlag.

[4] E. Czeizler, L. Kari, and S. Seki. On a special class of primitive words. In *Proc. of MFCS 2008*, volume 5162 of *Lecture Notes in Computer Science*, pages 265–277, Berlin-Heidelberg, 2008. Springer.

[5] A. de Luca and A. De Luca. Pseudopalindrome closure operators in free monoids. *Theoretical Computer Science*, 362:282–300, 2006.

[6] N. J. Fine and H. S. Wilf. Uniqueness theorem for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, February 1965.

[7] T. Harju and D. Nowotka. The equation $x^i = y^j z^k$ in a free semigroup. *Semigroup Forum*, 68:488–490, 2004.

[8] T. Harju and D. Nowotka. On the equation $x^k = z_1^{k_1} z_2^{k_2} \ldots z_n^{k_n}$ in a free semigroup. *Theoretical Computer Science*, 330(1):117–121, 2005.

[9] L. Kari, R. Kitto, and G. Thierrin. Codes, involutions and dna encoding. In W. Brauer, H. Ehrig, J. Karhumäki, and A. Salomaa, editors, *Formal and Natural Computing*, volume 2300 of *Lecture Notes in Computer Science*, pages 376–393. Springer-Verlag, Berlin, 2002.

[10] L. Kari and K. Mahalingam. Watson-Crick conjugate and commutative words. In *Proc. of DNA 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 273–283, 2008.

[11] A. Lentin. Sur l'équation $a^m = b^n c^p d^q$ dans un monoïde libre. *Comptes Rendus de l'Académie des Sciences Paris*, 260:3242–3244, 1965.

[12] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications.* Addison-Wesley, 1983.

[13] R. C. Lyndon and M. P. Schützenberger. The equation $a^m = b^n c^p$ in a free group. *Michigan Mathematical Journal*, 9:289–298, 1962.

[14] J. Maňuch. *Defect Theorems and Infinite Words.* PhD thesis, University of Turku, Department of Mathematics, FIN 20014 Turku, Finland, 2002.

[15] Gh. Păun, G. Rozenberg, and T. Yokomori. Hairpin languages. *International Journal of Foundations of Computer Science*, 12(6):837–847, 2001.

[16] H. J. Shyr. *Free Monoids and Languages.* Hon Min book company, Taichung, Taiwan, 3 edition, 2001.

Part III

Results in Formal Language Theory

Chapter 7

On pseudoknot-freeness

This chapter introduces the author's first journal publication[1] in his Ph.D. program:

L. Kari and S. Seki.

On pseudoknot-bordered words and their properties.

Journal of Computer and System Sciences, 75:113-121, 2009.

The conference version of this paper was presented at 2nd International Workshop on Natural Computing (IWNC):

L. Kari and S. Seki.

Towards the sequence design preventing pseudoknot formation.

In *IWNC*, PICT1, pages 101-110, Springer, 2009.

Summary: We study a generalization of the classical notions of bordered and unbordered words, motivated by biomolecular computing. DNA strands can be viewed

[1] A version of this chapter has been published.

as finite strings over the alphabet {A, G, C, T}, and are used in biomolecular computing to encode information. Due to the fact that Ais Watson-Crick complementary to Tand Gto C, DNA single strands that are Watson-Crick complementary can bind to each other or to themselves forming so-called secondary structures. Most of these secondary structures are undesirable for biomolecular computational purposes since the strands they involve cannot further interact with other strands. This paper studies pseudoknot-bordered words, a mathematical formalization of pseudoknot-like inter- and intra-molecular structures. In this context, pseudoknot-unbordered words model DNA or RNA strands that will be free of such secondary structures. We obtain several properties of pseudoknot-bordered and -unbordered words. We also address following problem: Given a pseudoknot-unbordered word u, does $\{u\}^+$ consist of pseudoknot-unbordered words only? We show that this is not generally true. We find that a sufficient condition for $\{u\}^+$ to consist of pseudoknot-unbordered words only is that u be not primitive. All of our results hold for arbitrary antimorphic involutions, of which the DNA Watson-Crick complementarity function is a particular case.

On pseudoknot-bordered words and their properties

Lila Kari and Shinnosuke Seki

Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

7.1 Introduction

In this paper we study pseudoknot-bordered and pseudoknot-unbordered words, which are generalizations of the classical notions of bordered and unbordered words, motivated by the need of optimally encoding information as DNA strands for biomolecular computing purposes.

A DNA single strand is a linear chain made up of four different types of nucleotides, each consisting of a sugar-phosphate unit and a base (Adenine, Cytosine, Guanine or Thymine). The sugar-phosphate units are linked together by strong covalent bonds, to form the backbone of the DNA strand. Since nucleotides may differ only by their bases, a DNA single strand can be viewed as a string over the DNA alphabet of bases $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. A DNA single strand has an orientation, with one end known as the 3' end, and the other known as the 5' end, based on their chemical properties. By convention, a word over the DNA alphabet represents a DNA single strand in its 5' to 3' orientation. An essential biochemical property of DNA single strands is that of Watson-Crick complementarity, wherein A can bind to T, and C can bind to G by weak hydrogen bonds. (In the case of RNA, T is replaced by U, and U is complementary to A, though the binding U-G may also occur.) Two Watson-Crick complementary DNA single strands of opposite orientation can bind to each other to form a DNA double strand. This and other biochemical properties of DNA have all been harnessed in biomolecular computing [1], in which information is encoded as DNA single strands, and processed through bio-operations [4].

One of the problems encountered when encoding information as DNA single strands is that the Watson-Crick complementarity often results in information-encoding DNA single strands either folding onto themselves to form intra-molecular structures, or interacting with each other to form inter-molecular structures. While these so-called secondary structures optimize biochemical determinants such as the Gibbs free-energy [19] and often have a significant role in determining the biochemical functions of real-life nucleic acids (DNA or RNA), in DNA computing they are often seen as a disadvantage. This is because it is very likely that the secondary structure formation of DNA strands will prevent them from interacting with other DNA strands in the expected, pre-programmed ways. Consequently, the property of a set of information-encoding strands to be free of unwanted intra- and inter-molecular structures has been intensively studied from many different points of view. These include design of algorithms based on free energy [2, 3, 16], algebra [13], and formal language theory [8, 9, 10, 11].

In this context, the notion of *antimorphic involution* θ was proposed, as the most natural mathematical formalization of the notion of DNA Watson-Crick com-

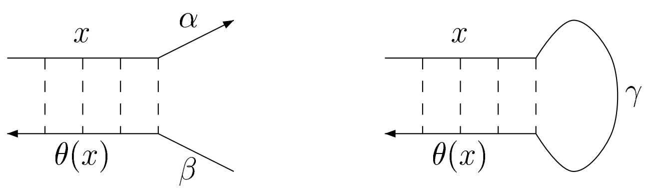

Figure 7.1: Inter- and intra-molecular structures which θ-unbordered words avoid.

plementarity [7, 10, 12]. Using this notion, Kari and Mahalingam [11] introduced and investigated the concept of a θ-unbordered word, as a formalization of DNA strands that avoid some of the most common inter- and intra-molecular structures. A θ-bordered word is a nonempty word which has a nonempty prefix x, and a suffix $\theta(x)$. If the alphabet under consideration is the DNA alphabet, and θ is the Watson-Crick complementarity function, then a θ-unbordered word represents a population of identical DNA single strands that are free from both inter-molecular structures such as the ones shown in Figure 7.1 (Left), and *hairpins* (words of the form $x\gamma\theta(x)$, shown in Figure 7.1, (Right)), one of the most common DNA intra-molecular structures. In addition to being relevant for DNA computing, the notions of θ-bordered and θ-unbordered words are generalizations of classical notions in combinatorics of words, namely those of bordered [5] (a.k.a. overlapping [22, 25], unipolar [23] words), respectively unbordered words (a.k.a. d-primitive, dipolar words).

The pseudoknot is another intra-molecular structure of biological significance,

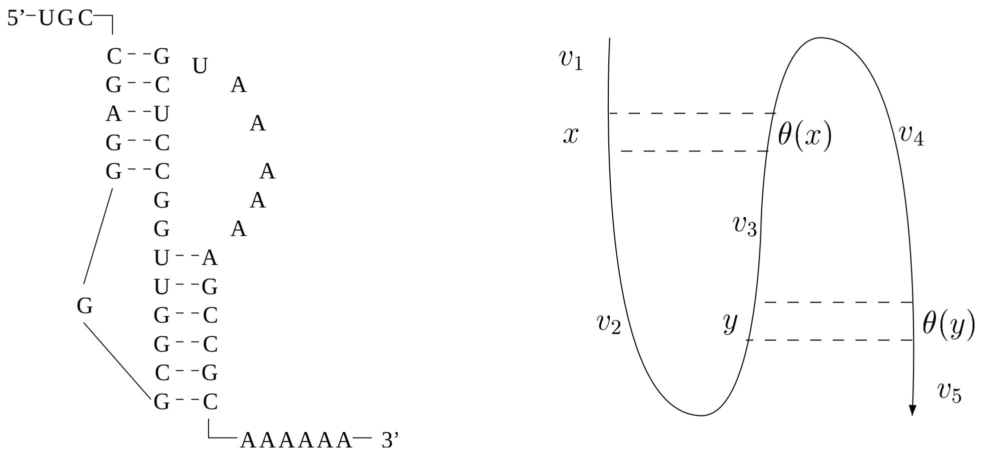

Figure 7.2: *Left:* A pseudoknot found in *E. Coli* transfer-messenger-RNA. (From Rfam [6]). *Right:* A depiction of a string modelling the pseudoknot in *Left*, as a word $v_1 x v_2 y v_3 \theta(x) v_4 \theta(y) v_5$. Here, $v_1 = \texttt{UGC}$, $x = \texttt{CGAGG}$, $v_2 = \texttt{G}$, $y = \texttt{GCGGUU}$, $v_3 = \texttt{GG}$, $v_4 = \texttt{UAAAAA}$, and $v_5 = \texttt{AAAAAA}$.

formed primarily by RNA strands. A pseudoknot found in *E. Coli* transfer-messenger-RNA is shown in Figure 7.2 (Left). This type of pseudoknot, which is the simplest and hence the most common, can be modelled as a word of the form $v_1 x v_2 y v_3 \theta(x) v_4 \theta(y) v_5$, as shown in Figure 7.2 (Right).

In this paper, we investigate not only such θ-pseudoknot-bordered words, but θ-pseudoknot-unbordered words, the latter being models of DNA or RNA strands that will not form pseudoknot-like inter- and intra-molecular structures. A nonempty word w is θ-pseudoknot-bordered if $w = xy\alpha = \beta\theta(yx)$ for some words x, y, α, and β. Thus, θ-pseudoknot-unbordered words avoid both inter-molecular bonds between identical strands of the type depicted in Figure 7.3 (Left), and intra-molecular structures of the form $xy\gamma\theta(x)\theta(y)$ shown in Figure 7.3 (Right). Note that this a particular case of the general model of pseudoknots, namely the case

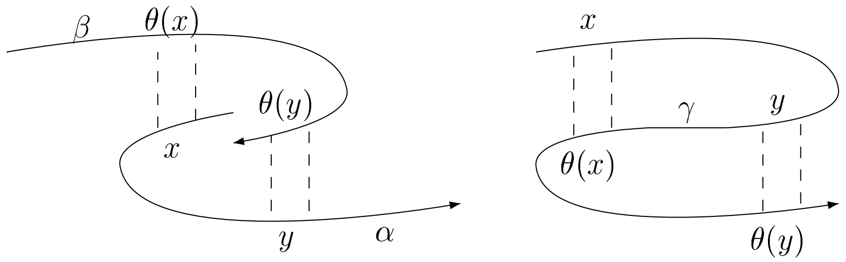

Figure 7.3: An inter-molecular structure and intra-molecular structure which θ-pseudoknot-unbordered words avoid.

where $v_1 = v_2 = v_4 = v_5 = \lambda$.

The paper is organized as follows. Using the notations and terminology given in Section 7.2, we propose the notion of θ-pseudoknot-bordered words in Section 7.3 and present some of their basic properties. We also show that the notion of θ-pseudoknot-bordered word is a proper generalization of that of θ-bordered word, and thus also properly generalizes the classical notion of bordered word. Since information-encoding DNA single strands often need to be concatenated together in the course of biocomputations, another problem of interest, which we address in Section 7.4, is whether the property of being pseudoknot-unbordered is preserved by catenation. Here we address the simplest case of this problem: Given a pseudoknot-unbordered word u, are all the words in $\{u^+\}$ still pseudoknot-unbordered? This turns out not to be always the case. However, we find a sufficient condition for a θ-pseudoknot-unbordered word u to satisfy the property that any power of u remains θ-pseudoknot-unbordered: the condition is that u be non-primitive (Corollary 7.13).

Section 7.5 discusses possible further directions of research.

7.2 Preliminaries

In this section we introduce the terminology and notations used in the paper. For details, we refer the reader to [21, 22, 25].

Let Σ be a finite alphabet. We denote by Σ^* the set of all words over Σ, and by Σ^+ the set of all nonempty words over Σ. Let λ be the empty word. Then $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$. For a word $w \in \Sigma^*$, $|w|$ denotes the length of w. A word u is said to be a *prefix* (*suffix*) of w if $w = uv$ (resp. $w = vu$) for some $v \in \Sigma^*$; here if $v \neq \lambda$, then the prefix (suffix) u is said to be *proper*. Let $\mathrm{Pref}(w)$ ($\mathrm{Suff}(w)$) be the set of all prefixes (resp. suffixes) of w.

A word $z \in \Sigma^*$ is said to be a *border* of a word $w \in \Sigma^*$ if $w = uz = zv$ for some words u and v in Σ^*. A nonempty word is said to be *bordered* if it admits a nonempty border, and it is said to be *unbordered* otherwise. A word $w \in \Sigma^+$ is called *primitive* if it cannot be written as a power of another word, i.e., $w = u^n$ with $u \in \Sigma^+$ implies $n = 1$. For a word $w \in \Sigma^+$, the shortest $u \in \Sigma^+$ satisfying $w = u^n$ for some $n \geq 1$ is called the *primitive root* of w. It is well known [17] that every nonempty word has a unique primitive root. Moreover, we have the following result due to Lyndon and Schützenberger.

Theorem 7.1. *For $u, v \in \Sigma^+$, $uv = vu$ implies that u and v have the same primitive* *root.*

For a word $w \in \Sigma^*$, a word $v \in \Sigma^*$ is called a *cyclic permutation* of w if there exist two words $x, y \in \Sigma^*$ such that $w = xy$ and $v = yx$. We denote the set of all cyclic permutations of w by $\mathrm{Cp}(w)$, that is, $\mathrm{Cp}(w) := \{yx \mid w = xy, x, y \in \Sigma^*\}$. Moreover, for a language $L \subseteq \Sigma^*$, we define $\mathrm{Cp}(L) := \bigcup_{w \in L} \mathrm{Cp}(w)$.

An *involution* $\theta : \Sigma \to \Sigma$ of a set Σ is a function such that θ^2 equals the identity function, i.e., $\theta(\theta(a)) = a$ for all $a \in \Sigma$. A *morphism* (*antimorphism*) θ on Σ^* is a function such that $\theta(xy) = \theta(x)\theta(y)$ (resp. $\theta(xy) = \theta(y)\theta(x)$) for all $x, y \in \Sigma^*$. A *d-morphism* is a generic term that refers to a function that is either a morphism or an antimorphism. An involution θ can be extended to a function $\theta : 2^{\Sigma^*} \to 2^{\Sigma^*}$, for a given language $L \subseteq \Sigma^*$, as follows: $\theta(L) := \{\theta(w) \mid w \in L\}$. In order to prove that the notion of θ-pseudoknot-bordered word is a proper generalization of the notion of a bordered word, in Section 7.3 we consider both morphic and antimorphic involutions. However, the problem of investigating whether catenations of pseudoknot-unbordered words have the same property is motivated mainly by DNA/RNA computing. Thus, in Section 7.4 we focus only on the mathematical formalization of the Watson-Crick complementarity, i.e., we consider only the case of antimorphic involutions.

A few words about morphic and antimorphic involutions are in order. Note that, if the alphabet Σ has m letters, and if we regard involutions that are isomorphic to each other as identical, the number of different involutions on Σ^* is $\lfloor m/2 \rfloor + 1$.

For example, on a binary alphabet $\Sigma = \{a, b\}$, there exist only two essentially different involutions: θ defined as $\theta(a) := b$ and $\theta(b) := a$, and the identity function. Each of these $\lfloor m/2 \rfloor + 1$ involutions can be extended to a morphic or antimorphic involution. With applications to the Watson-Crick complementarity in mind, herein we deal only with functions that are not the identity function. Thus, implicitly, we also exclude singleton alphabet sets. Note also that for any d-morphic involution θ that is not the identity, there exist two distinct characters $a, b \in \Sigma$ such that $\theta(a) = b$ and $\theta(b) = a$. We assume that in all the examples of this paper, for a given non-identity d-morphic involution θ, such $a, b \in \Sigma$ are chosen.

7.3 θ-pseudoknot-bordered words

In this section we propose the notion of *θ-pseudoknot-bordered words* for a morphic or antimorphic involution θ. If we consider the DNA alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$, wherein θ is the Watson-Crick complementarity function, then a word that is θ-pseudoknot-unbordered will not form pseudoknot-like secondary structures such as the ones in Figure 7.3. We show that the notion of θ-pseudoknot-bordered word is a proper generalization of the notion of θ-bordered word proposed in [11], and thus a proper generalization of the notion of bordered word. We also provide several properties of θ-pseudoknot-(un)bordered words.

Let θ be a d-morphic involution. A word $u \in \Sigma^*$ is said to be a *proper θ-border* of a word $w \in \Sigma^+$ if u is a proper prefix of w and $\theta(u)$ is a proper suffix of w, i.e., $w = u\alpha = \beta\theta(u)$ for some $\alpha, \beta \in \Sigma^+$. $\mathrm{L}_{\mathrm{d}}^{\theta}(w)$ denotes the set of all proper θ-borders of a nonempty word w. Note that $\lambda \in \mathrm{L}_{\mathrm{d}}^{\theta}(w)$ for all $w \in \Sigma^+$. A word $w \in \Sigma^+$ is said to be *θ-bordered* if it has a proper θ-border other than λ, i.e., $|\mathrm{L}_{\mathrm{d}}^{\theta}(w)| \geq 2$; otherwise, it is *$\theta$-unbordered*. Define now $\mathrm{D}_{\theta}(i) := \{w \in \Sigma^+ \mid |\mathrm{L}_{\mathrm{d}}^{\theta}(w)| = i\}$. Then $\mathrm{D}_{\theta}(1)$ is the set of all θ-unbordered words.

We call a word $u \in \Sigma^*$ a *θ-pseudoknot-border* (or θ-pk-border) of a word $w \in \Sigma^*$ if there exists a cyclic permutation v of u such that $w = u\alpha = \beta\theta(v)$ for some $\alpha, \beta \in \Sigma^*$. We also employ the expression "*xy is a θ-pk-border of w*" to mean "v is a θ-pk-border of w such that $v = xy$ and $w = xy\alpha = \beta\theta(yx)$ for some $\alpha, \beta \in \Sigma^*$". Let $\mathrm{L}_{\mathrm{cd}}^{\theta}(w)$ denote the set of all θ-pk-borders of a nonempty word w, and $\mathrm{K}_{\theta}(i) := \{w \in \Sigma^+ \mid |\mathrm{L}_{\mathrm{cd}}^{\theta}(w)| = i\}$. We call a nonempty word *θ-pseudoknot-bordered* (or θ-pk-bordered) if it has a nonempty θ-pk-border; otherwise, it is *θ-pseudoknot-unbordered*. Note that $\lambda \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w)$ for all $w \in \Sigma^+$. Note also that no word in $\mathrm{K}_{\theta}(1)$ has θ-pk-borders other than λ, and hence $\mathrm{K}_{\theta}(1)$ is the set of all θ-pk-unbordered words.

Example 22. Let θ be an antimorphic involution on Σ^* and $w = aababbb$. As mentioned in Section 7.2, $a, b \in \Sigma$ are chosen so as to satisfy $\theta(a) = b$ and $\theta(b) = a$. Then $\mathrm{L}_{\mathrm{cd}}^{\theta}(w) = \{\lambda, a, aa, aaba\}$. In particular, setting $x = aab$ and $y = a$ shows that $w = xybbb = aab\theta(yx)$ and hence $aaba \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w)$. Note that $w \in \mathrm{K}_{\theta}(4)$.

A word may have itself as its θ-pk-border, in both cases of θ being morphic and being antimorphic, as shown by the following examples.

Example 23. Let θ be a morphic involution on Σ^* and $w = abbaabba$. Then w can be written as $w = xy = \theta(yx)$ by letting $x = abbaab$ and $y = ba$.

Example 24. Let θ be an antimorphic involution on Σ^* and $w = ababbbaa$. Then $w = xy = \theta(yx)$ by letting $x = abab$ and $y = bbaa$.

Observe that the definitions of $\mathrm{L}^{\theta}_{\mathrm{d}}(w)$ and $\mathrm{L}^{\theta}_{\mathrm{cd}}(w)$ are different in that the former does not contain w while the latter can, if w is a θ-pk-border of itself. This scenario is different also from the classical case of bordered words, a particular case of θ-pk-bordered words where θ, as well as the permutation involved, are the identity. In the classical case, $\mathrm{L}_{\mathrm{d}}(w)$ denotes the set of proper borders of w, i.e., it does not contain w, since w is trivially always a border of itself. This definition was followed closely when defining $\mathrm{L}^{\theta}_{\mathrm{d}}(w)$, the set of proper θ-borders of a word w. However, in the case of θ-pseudoknot-bordered words we strayed from this model in defining $\mathrm{L}^{\theta}_{\mathrm{cd}}(w)$. This was because a word may, or may not, be a θ-pk-border of itself, and thus it is meaningful to observe for a word w, whether or not w belongs to $\mathrm{L}^{\theta}_{\mathrm{cd}}(w)$. This choice implies that, while all other notions proposed here are strict generalizations of the corresponding notions related to θ-bordered and bordered words, $\mathrm{L}^{\theta}_{\mathrm{cd}}(w)$ does not strictly generalize $\mathrm{L}^{\theta}_{\mathrm{d}}(w)$ and $\mathrm{L}_{\mathrm{d}}(w)$. Observe, however, that all the results obtained in this paper hold for the other definition choice for $\mathrm{L}^{\theta}_{\mathrm{cd}}(w)$ as well, either unchanged or augmented by a weak additional condition.

Since a word is a cyclic permutation of itself, if a word has a θ-border, then the θ-border also becomes a θ-pk-border of the word. Hence, the following lemma and its corollary hold.

Lemma 7.2. *Let θ be a d-morphic involution on Σ^* and $w \in \Sigma^+$. Then $\mathrm{L}_{\mathrm{d}}^{\theta}(w) \subseteq \mathrm{L}_{\mathrm{cd}}^{\theta}(w)$ holds.*

Corollary 7.3. *Let θ be a d-morphic involution on Σ^*. Then $\mathrm{K}_{\theta}(1) \subseteq \mathrm{D}_{\theta}(1)$.*

As shown in the following example, there exist a word w and a d-morphic involution θ for which $\mathrm{L}_{\mathrm{d}}^{\theta}(w)$ is strictly included in $\mathrm{L}_{\mathrm{cd}}^{\theta}(w)$.

Example 25. Let θ be a d-morphic involution on Σ^* and $w = aababbb$. For both cases of θ being morphic or antimorphic, $\mathrm{L}_{\mathrm{d}}^{\theta}(w) = \{\lambda, a, aa\}$ but $\mathrm{L}_{\mathrm{cd}}^{\theta}(w) = \{\lambda, a, aa, aaba\}$.

In the preceding example, $\mathrm{L}_{\mathrm{cd}}^{\theta}(w)$ happens to be the same whether the involution defined as $\theta(a) = b$ and vice versa is extended to a morphism, or to an antimorphism of Σ^*. This is not always the case, as indicated in the following two examples.

Example 26. Let θ be a d-morphic involution on Σ^* and $w = aabbabaababb$. When θ is morphic, $w = xyaababb = aabbab\theta(yx)$ for $x = aa$ and $y = bbab$, and hence $aabbab \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w)$. On the other hand, $aabbab \notin \mathrm{L}_{\mathrm{cd}}^{\theta}(w)$ if θ is antimorphic.

Example 27. Let θ be a d-morphic involution on Σ^* and $w' = aabbabbbabaa$. When θ is antimorphic, $w = xybbabaa = aabbab\theta(yx)$ for $x = aa$ and $y = bbab$, and hence $aabbab \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w')$. On the other hand, $aabbab \notin \mathrm{L}_{\mathrm{cd}}^{\theta}(w')$ for θ being morphic.

There exist alphabets Σ, and d-morphic involutions θ on Σ^*, for which the inclusion relation of Corollary 7.3 is proper. Indeed, let us consider a morphic involution θ, and a word $w \in \mathrm{D}_\theta(1)$ such that $w \notin \mathrm{K}_\theta(1)$. This implies that $w = xy\alpha = \beta\theta(y)\theta(x)$ for some $x, y, \alpha, \beta \in \Sigma^*$. If x were a proper prefix of w, then w would be θ-bordered, and hence $w = x = \theta(x)$. Hence, if there exists $c \in \Sigma$ such that $\theta(c) = c$, a word $w \in \mathrm{D}_\theta(1) \setminus \mathrm{K}_\theta(1)$ exists, and the inclusion relation is proper as seen by choosing $w = c$; otherwise $\mathrm{D}_\theta(1) = \mathrm{K}_\theta(1)$. For an antimorphic involution θ, we have the following example.

Example 28. Let θ be an antimorphic involution on Σ^* and $w = aba$. Then $w \in \mathrm{D}_\theta(1)$, but $w \notin \mathrm{K}_\theta(1)$ because $w = xya = a\theta(yx)$ for $x = a$ and $y = b$.

For a given d-morphic involution θ on Σ^*, a few remarks are in order regarding the set of all θ-pseudoknot-bordered words over Σ, i.e., the complement of $\mathrm{K}_\theta(1)$. For an antimorphic involution, which is of the most interest because of the biological motivation of this study, the cross-dependency existing in any θ-pk-bordered word indicates that the set of all θ-pk-bordered words is not context-free. This can indeed be proved by using the Pumping Lemma for context-free languages by choosing, e.g., an alphabet Σ, an antimorphic involution θ that maps a to b and vice versa, and the θ-pk-bordered word $a^n b^n a^n$, where n is the constant given by the Pumping Lemma. The fact that several (mild-)context-sensitive grammars or their stochastic variants were proposed to model pseudoknot structures [18, 20, 24] suggests that, for an antimorphic involution θ, the set of all θ-pk-bordered words over Σ is context-sensitive. This is indeed true, but we omit here the lengthy but straightforward construction of such a context-sensitive grammar, and the proof.

We conclude this section with some basic properties of θ-pk-borders, which will be used mainly in the proofs of the next section.

Lemma 7.4. *Let θ be a d-morphic involution on Σ^*. The following hold:*

1. *If a word $w \in \Sigma^+$ has a θ-pk-border of length n, then, for every $a \in \Sigma$, the number of occurrences of the letter a in the prefix of length n of w is equal to the number of occurrences of the letter $\theta(a)$ in the suffix of length n of w.*

2. *For all $a \in \Sigma$ such that $a \neq \theta(a)$, a^k is θ-pk-unbordered for all $k \geq 1$.*

3. *For words $v, w \in \Sigma^+$ and $n \geq 1$, if $v \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w^n)$ and $|w^{m-1}| < |v| \leq |w^m|$ for some $m \geq 1$, then $v \in \mathrm{L}_{\mathrm{cd}}^{\theta}(w^k)$ for all k with $m \leq k \leq n$.*

7.4 Primitive and θ-pseudoknot-unbordered words

One of the processes that are essential and often unavoidable in biocomputing algorithms is the concatenation of information-encoding DNA single strands. Thus, a question that is often asked is: Given some DNA strands having a certain "good" encoding property, will the catenation of these strands preserve this property? In this section we make steps towards answering this question in the case of the property of a word being θ-pseudoknot-unbordered. That is, for an antimorphic involution θ, we first address the following question: "Given a θ-pk-unbordered word u, is every word in $\{u\}^+$ also θ-pk-unbordered?" This question was answered positively for θ-unbordered words in [11]: A power of a θ-unbordered word is always θ-unbordered. We show that, in contrast, the question is answered negatively for θ-pk-unbordered words. Moreover, we provide a sufficient condition for a θ-pk-unbordered word to satisfy the condition that all of its powers are θ-pk-unbordered (Corollary 7.13).

We begin by providing a necessary and sufficient condition for a word to be θ-pk-unbordered, which follows directly from the definition of a θ-pk-bordered word.

Lemma 7.5. *Let θ be an antimorphic involution on Σ^*. Then a word $u \in \Sigma^+$ is θ-pk-unbordered if and only if $\theta(\mathrm{Cp}(\mathrm{Pref}(u))) \cap \mathrm{Suff}(u) = \emptyset$.*

For a d-morphic involution θ on Σ^*, a word $w \in \Sigma^*$ is called *θ-palindrome* if $w = \theta(w)$. Let P_θ denote the set of all θ-palindromes over Σ.

Lemma 7.6. *Let θ be an antimorphic involution on Σ^*, and x, y be θ-palindromes such that $xy \neq \lambda$. If a word $u \in \Sigma^+$ has xy as both its prefix and suffix, then u is θ-pk-bordered.*

Proof. Let $u = xy\alpha = \beta xy$ for some $\alpha, \beta \in \Sigma^*$. The fact that $x, y \in \mathrm{P}_\theta$ implies that $u = \beta\theta(x)\theta(y) = \beta\theta(yx)$, Therefore, u is θ-pk-bordered. $\square$

Recall the following result from [11].

Lemma 7.7. *Let θ be an antimorphism on Σ^* and let $u \in \Sigma^+$. Then $u \in \mathrm{D}_\theta(1)$ if and only if $u^+ \subseteq \mathrm{D}_\theta(1)$.*

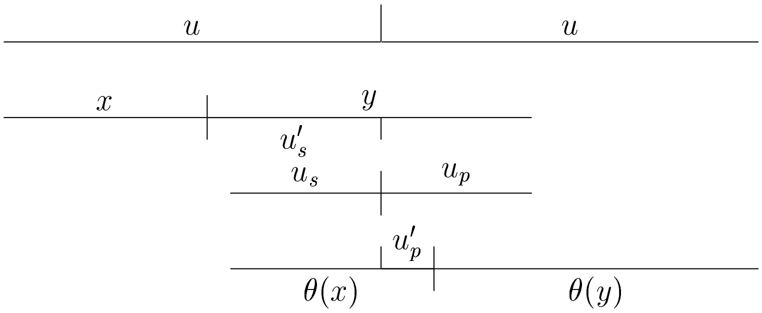

Figure 7.4: A pictorial representation of Case 2 of the proof of Proposition 7.8.

In contrast, the following example shows that there exist θ-pk-unbordered words u such that at u^k is θ-pk-bordered for some $k \geq 2$.

Example 29. Let θ be an antimorphic involution on Σ^* and $u = aabbbbaba$. Although u is θ-pk-unbordered, u^2 is θ-pk-bordered. In fact, $u^2 = xyabbbbaba = aabbbbab\theta(x)\theta(y)$ for $x = aabbb$ and $y = babaa$.

In the following, we give a characterization of θ-pk-unbordered words u with the property that u^k is θ-pk-bordered for some $k \geq 2$, that takes into account the relative length of the θ-pk-borders of u^2.

Proposition 7.8. *Let θ be an antimorphic involution on Σ^*. Then for a θ-pk-unbordered word u, if there exists $k \geq 2$ such that u^k has a nonempty θ-pk-border w, then $|u| < |w| < \frac{4}{3}|u|$ holds.*

Proof. Suppose for some $k \geq 2$, there were a $w \in \mathrm{L}^{\theta}_{\mathrm{cd}}(u^k)$ such that either $|w| \leq |u|$ or $\frac{4}{3}|u| \leq |w|$ hold. If $|w| \leq |u|$, then this w leads us to a contradiction immediately.

Next we consider the case $\frac{4}{3}|u| \leq |w| < 2|u|$. Then $w \in \mathrm{L}^{\theta}_{\mathrm{cd}}(u^k)$ implies $w \in \mathrm{L}^{\theta}_{\mathrm{cd}}(u^2)$. In other words, there exists a decomposition $w = xy$ such that $uu = xy\alpha = \beta\theta(x)\theta(y)$ for some $\alpha, \beta \in \Sigma^+$. Since $|w| \geq \frac{4}{3}|u|$, we have $xy = uu_p$ and $\theta(x)\theta(y) = u_s u$, where $u_p \in \mathrm{Pref}(u)$, and $u_s \in \mathrm{Suff}(u)$. Now we have the following two cases:

1. $|x| \geq |u|$ or $|y| \geq |u|$ holds,
2. $|x| < |u|$ and $|y| < |u|$ hold.

In the first case, for reasons of symmetry, we only have to consider the case $|x| \geq |u|$. Since $\theta(x)\theta(y) = u_s u$, we can write $\theta(x) = u_s u'_p$ for some $u'_p \in \mathrm{Pref}(u)$. Let $u = u'_p u'_s$, and we can easily check that $u'_s \in \mathrm{Suff}(u_s)$. Therefore, $u'_s u'_p \in \mathrm{Suff}(\theta(x))$, which equals $\theta(u'_p)\theta(u'_s) \in \mathrm{Pref}(x)$. This means that $\theta(u'_p)\theta(u'_s) = u$ because u and $\theta(u'_p)\theta(u'_s)$ are prefixes of x and they have equal lengths. Since $u = u'_p u'_s$, we conclude that both u'_p and u'_s are θ-palindromes. The application of Lemmata 7.5 and 7.6 leads now to a contradiction.

Next we consider the second case (see Figure 7.4). This figure shows $xy = uu_p$ and $\theta(x)\theta(y) = u_s u$. Since both x and y are shorter than u, these equations imply that $u = xu'_s = u'_p\theta(y)$, where $u'_p \in \mathrm{Pref}(u)$ and $u'_s \in \mathrm{Suff}(u)$. Comparing this equation with $xy = uu_p$ we derive $y = u'_s u_p$, and hence $u = u'_p\theta(u_p)\theta(u'_s)$. This result, together with $u = xu'_s$, implies that u'_s is a θ-palindrome and $x = u'_p\theta(u_p)$. Substituting this x and $u = u'_p\theta(y)$ into $\theta(x)\theta(y) = u_s u$ gives $u_p\theta(u'_p)\theta(y) = u_s u'_p\theta(y)$, which means that $u_p = u_s$ and u'_p is a θ-palindrome.

Let us bring now into the picture the original condition $\frac{4}{3}|u| \leq |w| < 2|u|$. Since $|w| = |u| + |u_p|$, $\frac{4}{3}|u| \leq |w|$ means $\frac{1}{3}|u| \leq |u_p|$. Hence, $|xy| = |uu_p| \leq 4|u_p|$. This implies that either $|x| \leq 2|u_p|$ or $|y| \leq 2|u_p|$ holds. We assume the former case holds. Then $\theta(x) = u_s u'_p$ implies $|u'_p| \leq |u_s|$ because $|\theta(x)| = |x| \leq 2|u_p| = 2|u_s|$. Let $u_s = u_1 u_2$ such that $|u_1| = |u'_p|$. Note that $u_s \in \text{Pref}(x)$ because $u_p, x \in \text{Pref}(u)$, $|u_s| < |x|$, and $u_p = u_s$. Comparing $u_s = u_1 u_2$ with $x = \theta(u_1 u_2 u'_p)$ based on $u_s \in \text{Pref}(x)$ results in $u_2 = \theta(u_2)$ and $u_1 = \theta(u'_p)$, which in turn implies $u_1 = \theta(u_1)$ because $u'_p = \theta(u'_p)$. Now Lemmata 7.5 and 7.6 lead to a contradiction because u contains the concatenation of two θ-palindromes u_1 and u_2 as its prefix u_p and suffix u_s.

What remains to consider is the case where $|w| \geq 2|u|$. Let $w = xy = u^n u_p$ and $\theta(x)\theta(y) = u_s u^n$ for some $n \geq 2$, $u_p \in \text{PPref}(u)$, and $u_s \in \text{PSuff}(u)$. Then either $|x| \geq |u|$ or $|y| \geq |u|$ holds. Let us assume that $|x| \geq |u|$ holds. Then $\theta(x) = u_s u^m u'_p$ and $\theta(y) = u'_s u^{n-m-1}$ for some $m < n$, where $u'_p u'_s = u$. If $u'_s \in \text{Suff}(u_s)$, then $u'_s u^m u'_p \in \text{Suff}(\theta(x))$, which implies $\theta(u'_p)\theta(u)^m\theta(u'_s) \in \text{Pref}(x)$. Since x is not shorter than u, $u = \theta(u'_p)\theta(u'_s)$. Thus, u can be factorized into two θ-palindromes, a contradiction. If $u_s \in \text{PSuff}(u'_s)$, then m must be at least 1; otherwise, $|\theta(x)| = |u_s u'_p| < |u'_s u'_p| = |u|$. Therefore, $u'_s u'_p \in \text{Suff}(\theta(x))$, i.e., $\theta(u'_p)\theta(u'_s) \in \text{Pref}(x)$. Now we have $u = \theta(u'_p)\theta(u'_s)$, and this leads to the same contradiction as above. □

Note that, in Example 29, the θ-pk-border xy of u^2 satisfies $|u| < |xy| < \frac{4}{3}|u|$.

Corollary 7.9. *Let θ be an antimorphic involution on Σ^*. For a word $u \in \mathrm{K}_\theta(1)$, $u^+ \not\subseteq \mathrm{K}_\theta(1)$ if and only if $u^2 \notin \mathrm{K}_\theta(1)$.*

The next lemma is a consequence of the proof of Proposition 7.8, and will be a useful tool in obtaining several additional properties of θ-pk-unbordered words whose square is θ-pk-bordered.

Lemma 7.10. *Let θ be an antimorphic involution on Σ^*, and u be a θ-pk-unbordered word. If $xy \in \mathrm{L}^\theta_{\mathrm{cd}}(u^2)$ such that $xy = uu_p$ for some $u_p \in \mathrm{Pref}(u)$, then $2|u_p| < |x| < |u|$ and $2|u_p| < |y| < |u|$ hold.*

In what follows, we give a characterization of θ-pk-unbordered words whose square is θ-pk-bordered.

Lemma 7.11. *Let θ be an antimorphic involution on Σ^*, and u be a θ-pk-unbordered word. Then u^2 is θ-pk-bordered if and only if $u = u_p\alpha\theta(u_p)\beta u_p$ for some $u_p, \alpha, \beta \in \Sigma^+$ such that $u_p\alpha, \beta u_p$ are θ-palindromes.*

Proof. **(Only if)** Let $u^2 = xy\gamma_1 = \gamma_2\theta(x)\theta(y)$ for xy such that $|u| < |xy| < \frac{4}{3}|u|$. Then $xy = uu_p$ and $\theta(x)\theta(y) = u_s u$ for some $u_p \in \mathrm{Pref}(u)$ and $u_s \in \mathrm{Suff}(u)$, which satisfy $|u_p| = |u_s| < \frac{1}{3}|u|$. In addition, Lemma 7.10 enables us to assume that $2|u_p| < |x| < |u|$ and $2|u_p| < |y| < |u|$. Now we have $\theta(x) = u_s u_p \alpha$ and $y = \beta u_s u_p$ for some $\alpha, \beta \in \Sigma^+$. Then $x = \theta(u_p\alpha)\theta(u_s)$ and $\theta(y) = \theta(u_p)\theta(\beta u_s)$. Substituting these into $xy = uu_p$ and $\theta(x)\theta(y) = u_s u$ gives that $u = \theta(u_p\alpha)\theta(u_s)\beta u_s = u_p\alpha\theta(u_p)\theta(\beta u_s)$. This means that both $u_p\alpha$ and βu_s are θ-palindromes and $\theta(u_p) = \theta(u_s)$. Thus, $u_p = u_s$ and then $u = u_p\alpha\theta(u_p)\beta u_p$. **(If)** Let $x = u_p\alpha\theta(u_p)$ and $y = \beta u_p u_p$. Then $\theta(x)\theta(y) = u_p\theta(u_p\alpha)\theta(u_p)\theta(\beta u_p)$. We can rewrite the right-hand side as $u_p u_p\alpha\theta(u_p)\beta u_p$ because $u_p\alpha, \beta u_p \in \mathrm{P}_\theta$. This means $\theta(x)\theta(y) = u_p u \in \mathrm{Suff}(uu)$. $\square$

Lemma 7.12. *Let θ be an antimorphic involution on Σ^*, and u be a θ-pk-unbordered word. If u^2 is θ-pk-bordered, then u is primitive.*

Proof. Since u^2 has a θ-pk-border uu_p for some $u_p \in \mathrm{Pref}(u)$, Lemma 7.11 implies that u can be written as $u_p\alpha\theta(u_p)\beta u_p$ for some $\alpha, \beta \in \Sigma^*$ such that $u_p\alpha, \beta u_p \in \mathrm{P}_\theta$. Suppose u were not primitive, i.e., $u = w^r$ for some $w \in \Sigma^+$ and $r \geq 2$. To begin with, we consider the case $|u_p| \leq |w|$. This case has the two subcases depending on whether there exists an integer n such that $|u_p\alpha| < |w^n| < |u_p\alpha\theta(u_p)|$, where $1 \leq n \leq r-1$, or not. If such n exists, the infix $\theta(u_p)$ overlaps with the n-th occurrence of w and with the $(n+1)$-th occurrence of w, counted from the left. Let $\theta(u_p) = \theta(u_2)\theta(u_1)$ such that $\theta(u_2) \in \mathrm{Suff}(w)$ and $\theta(u_1) \in \mathrm{Pref}(w)$. Then we have $u_p = u_1u_2$. Both u_p and w are prefixes of u and $|u_p| \leq |w|$ so that $u_1 \in \mathrm{Pref}(w)$, and hence $u_1 = \theta(u_1)$. In the same way, $u_2 = \theta(u_2)$. Then Lemma 7.6 leads to a contradiction with the fact that $u \in \mathrm{K}_\theta(1)$.

Next we consider the other subcase. We can rewrite this subcase as follows: There exists an integer n such that $|w^n| \leq |u_p\alpha|$ and $|u_p\alpha\theta(u_p)| \leq |w^{n+1}|$, where

$0 \le n \le r-1$. Depending on the value of n, there exist 3 possibilities to be taken into account: (a) $n = 0$, (b) $n = r-1$, and (c) otherwise. In case (a), we can write $w = u_p\alpha\theta(u_p)\beta_p$, $\beta_i = w^{r-2}$, and $w = \beta_s u_p$ for some $\beta_p, \beta_i, \beta_s \in \Sigma^*$ such that $\beta_p\beta_i\beta_s = \beta$. Then $\beta u_p = \beta_p w^{r-1}$. Replacing one occurrence of w in the right-hand of this equation with $u_p\alpha\theta(u_p)\beta_p$ gives $\beta u_p = \beta_p u_p\alpha\theta(u_p)\beta_p w^{r-2} = \beta_p w^{r-2} u_p\alpha\theta(u_p)\beta_p$. This means that both β_p and $u_p\alpha\theta(u_p)$ are θ-palindromes because $\beta u_p \in \mathrm{P}_\theta$. Therefore, w is the concatenation of two θ-palindromes. Since u has w as its prefix and suffix, Lemma 7.6 leads to a contradiction. Case (b) is similar. In case (c), let $w = u_p\alpha_p$, $w^{n-1} = \alpha_i$, $w = \alpha_s\theta(u_p)\beta_p$, $w^{r-n-2} = \beta_i$, and $w = \beta_s u_p$ for $\alpha_p, \alpha_i, \alpha_s, \beta_p, \beta_i, \beta_s \in \Sigma^*$ such that $\alpha_p\alpha_i\alpha_s = \alpha$ and $\beta_p\beta_i\beta_s = \beta$. Then one has $u_p\alpha = w^n\alpha_s$. Substituting $w = \alpha_s\theta(u_p)\beta_p$ into one occurrence of w in the right-hand side of this equation gives $u_p\alpha = \alpha_s\theta(u_p)\beta_p w^{n-1}\alpha_s = w^{n-1}\alpha_s\theta(u_p)\beta_p\alpha_s$. Since $u_p\alpha \in \mathrm{P}_\theta$, both α_s and $\theta(u_p)\beta_p$ are θ-palindromes. Then Lemma 7.6 leads to a contradiction as above.

Finally, we consider the case where w is shorter than u_p. Then there exist two integers n and h satisfying $|w^n| \le |u_p\alpha| < |w^{n+1}|$ and $|w^{n+1+h}| < |u_p\alpha\theta(u_p)| \le |w^{n+1+h+1}|$, where $h \ge 0$. Hence, we can write $\alpha_s\theta(u_p)\beta_p = w^{h+2}$ for some $\alpha_s \in \mathrm{Suff}(\alpha)$ and $\beta_p \in \mathrm{Pref}(\beta)$ such that $|\alpha_s|, |\beta_p| < |w|$. Thus, $w = \gamma\beta_p$ for some $\gamma \in \mathrm{Suff}(\theta(u_p))$, and hence $\alpha_s\theta(u_p)\beta_p = (\gamma\beta_p)^h\gamma\beta_p\gamma\beta_p$. This equation means $\beta_p\gamma \in \mathrm{Suff}(\theta(u_p))$ because $|w| < |\theta(u_p)|$. Therefore, $\theta(\beta_p\gamma) = \theta(\gamma)\theta(\beta_p) \in \mathrm{Pref}(u_p) \subseteq \mathrm{Pref}(u)$. In addition, $w = \gamma\beta_p \in \mathrm{Suff}(u)$. Thus, $u = \theta(\gamma)\theta(\beta_p)v\gamma\beta_p$ for some $v \in \Sigma^*$, which conflicts with $u \in \mathrm{K}_\theta(1)$. $\square$

Corollary 7.13. *Let θ be an antimorphic involution on Σ^*. If u is a non-primitive θ-pk-unbordered word, then u^2 is θ-pk-unbordered. This further implies that any power of u is θ-pk-unbordered.*

Example 30. Let θ be an antimorphic involution on Σ^* and $u = abaaabaa$, which is clearly not primitive. It is easy to see that neither u nor u^2 is θ-pk-bordered. Hence, u^k is θ-pk-unbordered for any $k \geq 1$.

For a θ-pk-unbordered word u whose square is θ-pk-bordered, we now investigate the primitivity of θ-pk-borders of u^2.

Theorem 7.14. *Let θ be an antimorphic involution on Σ^*, and u be a θ-pk-unbordered word whose square is θ-pk-bordered. Then any θ-pk-border of u^2 is primitive.*

Proof. Let uu_p be a θ-pk-border of u^2. Lemma 7.11 says $u = u_p\alpha\theta(u_p)\beta u_p$ for $u_p, \alpha, \beta \in \Sigma^+$ such that $u_p\alpha, \beta u_p \in \mathrm{P}_\theta$. Suppose uu_p is not primitive, that is, $uu_p = w^r$ for $w \in \Sigma^+$ and $r \geq 2$.

To begin with, we assume $|w| = |u_p|$. Then we have $u = w^{r-1}$. The condition $|u_p| < \frac{1}{3}|u|$, which is necessary for $u^+ \not\subseteq K_\theta(1)$, implies $r > 4$. Thus, u would be not primitive, a contradiction.

Next we assume $|w| > |u_p|$. As in the proof of Lemma 7.12, we have to consider the following two cases:

1. There exists an integer n such that $|u_p\alpha| < |w^n| < |u_p\alpha\theta(u_p)|$, where $1 \leq n \leq r-1$,

2. There exists an integer n such that $|w^n| \leq |u_p\alpha|$ and $|u_p\alpha\theta(u_p)| \leq |w^{n+1}|$, where $0 \leq n \leq r-1$.

In case 1, let $\theta(u_p) = \theta(u_2)\theta(u_1)$ such that $\theta(u_2) \in \mathrm{Suff}(w)$ and $\theta(u_1) \in \mathrm{Pref}(w)$. Note that w has u_p as both its prefix and suffix because w^r $(= uu_p)$ has u_p both as its prefix and as its suffix, and $|u_p| < |w|$. Hence, we have $\theta(u_2) \in \mathrm{Suff}(u_p)$ and $\theta(u_1) \in \mathrm{Pref}(u_p)$. Since $u_p = u_1u_2$, we also have $u_1 \in \mathrm{Pref}(u_p)$ and $u_2 \in \mathrm{Suff}(u_p)$, which means that both u_1 and u_2 are θ-palindromes. Then Lemma 7.6 leads to a contradiction to the fact that $u \in \mathrm{K}_\theta(1)$ because u has the product of two θ-palindromes u_1u_2 both as its prefix and as its suffix.

In case 2, there are three possibilities depending on the value of n: (a) $n = 0$, (b) $n = r-1$, and (c) otherwise.

In subcase 2(a), we can write $w = u_p\alpha\theta(u_p)\beta_p$, $w^{r-3} = \beta_i$, $\beta_s u_p = ww_p$, and $w = w_pu_p$ for some $w_p \in \mathrm{Pref}(w)$ and $\beta_p, \beta_i, \beta_s \in \Sigma^*$ such that $\beta = \beta_p\beta_i\beta_s$. Now we can rewrite $\theta(u_p)\beta u_pu_p = \theta(u_p)\beta_p\beta_i\beta_s u_pu_p = \theta(u_p)\beta_p\beta_i w^2 = \theta(u_p)\beta_p\beta_i wu_p\alpha\theta(u_p)\beta_p$. Then we can say that $\theta(u_p)\beta_p$ is a θ-palindrome because $\theta(u_p)\beta u_pu_p \in \mathrm{P}_\theta$. Therefore, $w = u_p\alpha\theta(u_p)\beta_p = u_p\alpha\theta(\beta_p)u_p$. Compared to $w = w_pu_p$, we have $w_p = u_p\alpha\theta(\beta_p)$.

Then,

$$
\begin{aligned}
w = u_p\alpha\theta(u_p)\beta_p \in \mathrm{Pref}(u) \ &\Rightarrow\ \beta_p u_p \alpha\theta(u_p) \in \mathrm{Cp}(\mathrm{Pref}(u)),\\
&\Leftrightarrow\ \beta_p\theta(\alpha)\theta(u_p)\theta(u_p) \in \mathrm{Cp}(\mathrm{Pref}(u)),\\
&\Leftrightarrow\ u_p u_p \alpha\theta(\beta_p) \in \theta(\mathrm{Cp}(\mathrm{Pref}(u))),\\
&\Leftrightarrow\ u_p w_p \in \theta(\mathrm{Cp}(\mathrm{Pref}(u))).
\end{aligned}
$$

The third implication is due to the fact that $u_p\alpha \in \mathrm{P}_\theta$. Since $w^2 = w_p u_p w_p u_p$ is the suffix of uu_p, $u_p w_p \in \mathrm{Suff}(u)$, and hence $\theta(\mathrm{Cp}(\mathrm{Pref}(u))) \cap \mathrm{Suff}(u) \neq \emptyset$, which is a contradiction.

In subcase 2(b), we can write $w = u_p\alpha_p$, $w^{r-2} = \alpha_i$, $w_p = \alpha_s\theta(u_p)\beta u_p$, and $w_p u_p = w$ for some $w_p \in \mathrm{Pref}(w)$ and $\alpha_p, \alpha_i, \alpha_s \in \Sigma^*$ such that $\alpha = \alpha_p\alpha_i\alpha_s$. As in the cases above, since $u_p\alpha = \theta(u_p\alpha)$, α_s is also a θ-palindrome. Starting from $w \in \mathrm{Pref}(u)$, now we can show $u_p w_p \in \theta(\mathrm{Cp}(\mathrm{Pref}(u))) \cap \mathrm{Suff}(u)$.

In subcase 2(c), let $w = u_p\alpha_1$, $w^{n-1} = \alpha_2$, $w = \alpha_3\theta(u_p)\beta_1$, $w^{r-n-3} = \beta_2$, $\beta_3 u_p = ww_p$, and $w = w_p u_p$ for some $w_p \in \mathrm{Pref}(w)$ and $\alpha_p, \alpha_i, \alpha_s, \beta_p, \beta_i, \beta_s \in \Sigma^*$ such that $\alpha = \alpha_p\alpha_i\alpha_s$ and $\beta = \beta_p\beta_i\beta_s$. Using these notations, we have $u_p\alpha = w^n\alpha_s = w^{n-1}\alpha_s\theta(u_p)\beta_p\alpha_s = \alpha_s\theta(u_p)\beta_p w^{n-1}\alpha_s$. Since $u_p\alpha \in \mathrm{P}_\theta$, we can observe that both α_s and $\theta(u_p)\beta_p$ are also θ-palindromes. Thus, $w = \alpha_s\theta(u_p)\beta_p = \alpha_s\theta(\beta_p)u_p$. Compared to $w = w_p u_p$, we can say $w_p = \alpha_s\theta(\beta_p)$. Now we can obtain a contradiction to

$u \in K_\theta(1)$ as above. This completes the discussion of the case $|w| > |u_p|$ with its two possibilities (1) and (2).

Finally, we consider the case where $|w| < |u_p|$. This means that there exist two integers n and h satisfying $|w^n| \leq |u_p\alpha| < |w^{n+1}|$ and $|w^{n+1+h}| < |u_p\alpha\theta(u_p)| \leq |w^{n+1+h+1}|$, where $h \geq 0$. Hence, we have $\alpha_s\theta(u_p)\beta_p = w^{h+2}$ for some $\alpha_s \in \mathrm{Suff}(\alpha)$ and $\beta_p \in \mathrm{Pref}(\beta)$ such that $|\alpha_s|, |\beta_p| < |w|$. Therefore, $w = \gamma\beta_p$ for some $\gamma \in \mathrm{Suff}(\theta(u_p))$, and hence $\alpha_s\theta(u_p)\beta_p = (\gamma\beta_p)^h\gamma\beta_p\gamma\beta_p$. This implies $\beta_p\gamma \in \mathrm{Suff}(\theta(u_p))$ because $|w| < |\theta(u_p)|$. This means $\theta(\gamma)\theta(\beta_1) \in \mathrm{Pref}(u_p)$. Moreover $w = \gamma\beta_p \in \mathrm{Suff}(u_p)$ because the rightmost occurrence of u_p in $uu_p = w^r$ has w as its suffix. Thus, $u = \theta(\gamma)\theta(\beta_1)v\gamma\beta_1$ for some $v \in \Sigma^*$ because u has u_p both as its prefix and as its suffix. This conclusion contradicts $u \in \mathrm{K}_\theta(1)$. $\square$

We conclude this section by showing that, for a word $u \in \mathrm{K}_\theta(1)$, if $xy, x'y' \in \mathrm{L}_{\mathrm{cd}}^{\theta}(u^2)$ with $|xy| = |x'y'|$, then $x = x'$ and $y = y'$.

Lemma 7.15. *Let θ be an antimorphic involution on Σ^*, and $w \in \Sigma^+$. A θ-pk-border v of w is not primitive if and only if there exist words x, y, x', y' satisfying $v = xy = x'y'$, $yx = y'x'$, and $x \neq x'$.*

Proof. **(Only if)** Under the assumption, $v = xy = z^n$ for some primitive word $z \in \Sigma^+$ and $n \geq 2$. Then $x = z^iz_p$ and $y = z_sz^{n-i-1}$ for some $i \geq 0$, $z_p, z_s \in \Sigma^*$ such that $z = z_pz_s$. Let $x' = z^jz_p$ and $y' = z_sz^{n-j-1}$ for some $j \neq i$. Since $n \geq 2$, such j exists. Clearly $xy = x'y'$ and we can easily check $y'x' = yx$. **(If)** We can represent w as both $w = xy\alpha = \beta\theta(yx)$ and $w = x'y'\alpha = \beta\theta(y'x')$. Without loss of generality, we can assume $|x'| < |x|$, and then this implies that $\theta(x) = \theta(x')q$ and $\theta(y') = q\theta(y)$ for some $q \in \Sigma^+$. Therefore, $x = \theta(q)x'$ and $y' = y\theta(q)$. Substituting these into $xy = x'y'$, we obtain $xy = \theta(q)x'y = x'y\theta(q)$. Then Theorem 7.1 implies that v is not primitive. □

The next proposition now follows from Theorem 7.14 and Lemma 7.15.

Proposition 7.16. *Let θ be an antimorphic involution on Σ^* and u be a θ-pk-unbordered word. If w is a nonempty θ-pk-border of u^2, then the factorization of w into x and y such that $u^2 = xy\alpha = \beta\theta(yx)$ for some $\alpha, \beta \in \Sigma^*$ is unique.*

7.5 Discussion

In this paper, we proposed the notion of a θ-pseudoknot-unbordered word, where θ is a morphic or antimorphic involution. This concept models DNA (or RNA) single strands that do not form some pseudoknot-like secondary structures. This formulation is general enough to handle intermolecular structures similar to pseudoknots. In addition, this notion is a proper generalization of the notion of θ-unbordered word, and thus of the classical notion of unbordered word. After obtaining some basic properties of θ-bordered and θ-unbordered words, we investigated the question of whether or not all powers of a θ-unbordered words remain θ-unbordered. The question was answered in the negative by providing counterexamples. We also showed that, for a θ-unbordered word u, the fact that u is not primitive is a sufficient condition for u^k to be θ-pseudoknot-unbordered for all $k \geq 1$. This is the first step towards obtaining a condition that a language L of θ-pseudoknot-unbordered words would have to satisfy in order for L^+ to have the same property. Another direction of research is to consider more realistic pseudoknot structures, i.e., to remove the restriction $v_1 = v_2 = v_4 = v_5 = \lambda$ in the general definition of the pseudoknot as a word of the form $v_1 x v_2 y v_3 \theta(x) v_4 \theta(y) v_5$. In particular, the conditions $v_2 = \lambda$ and $v_4 = \lambda$ should be weakened, because pseudoknots occurring in real RNAs rarely satisfy these conditions due to steric effects.

Bibliography

[1] L. M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187):1021–1024, November 1994.

[2] M. Andronescu, D. Dees, L. Slaybaugh, Y. Zhao, A. Condon, B. Cohen, and S. Skiena. Algorithms for testing that sets of DNA words concatenate without secondary structure. In M. Hagiya and A. Ohuchi, editors, *DNA Based Computers 8*, volume 2568 of *Lecture Notes in Computer Science*, pages 182–195. Springer, 2003.

[3] A. Condon. Problems on RNA secondary structure prediction and design. In *ICALP 2003*, volume 2719, pages 22–32. Springer, 2003.

[4] M. Daley and L. Kari. DNA computing: Models and implementations. *Comments on Theoretical Biology*, 7(3):177–198, 2002.

[5] A. Ehrenfeucht and D. Silberger. Periodicity and unbordered segments of words. *Discrete Mathematics*, 26(2):101–109, 1979.

[6] S. G.-Jones, S. Moxon, M. Marshall, A. Khanna, S. R. Eddy, and A. Bateman. Rfam: Annotating non-coding RNAs in complete genomes. *Nucleic Acids Research*, 33:D121–D124, 2005.

[7] N. Jonoska, K. Mahalingam, and J. Chen. Involution codes: With application to DNA coded languages. *Natural Computing*, 4(2):141–162, 2005.

[8] L. Kari, R. Kitto, and G. Thierrin. Codes, involutions and dna encoding. In W. Brauer, H. Ehrig, J. Karhum#aki, and A. Salomaa, editors, *Formal and Natural Computing*, volume 2300 of *Lecture Notes in Computer Science*, pages 376–393. Springer-Verlag, Berlin, 2002.

[9] L. Kari, S. Konstantinidis, E. Losseva, P. Sosík, and G. Thierrin. A formal language analysis of DNA hairpin structures. *Fundamenta Informaticae*, 71(4):453–475, 2006.

[10] L. Kari, S. Konstantinidis, E. Losseva, and G. Wozniak. Sticky-free and overhang-free DNA languages. *Acta Informatica*, 40:119–157, 2003.

[11] L. Kari and K. Mahalingam. Involutively bordered words. *International Journal of Foundations of Computer Science*, 18(5):1089–1106, 2007.

[12] L. Kari and K. Mahalingam. Watson-Crick conjugate and commutative words. In *Proc. of DNA 13*, volume 4848 of *Lecture Notes in Computer Science*, pages 273–283, 2008.

[13] L. Kari, K. Mahalingam, and G. Thierrin. The syntactic monoid of hairpin-free languages. *Acta Informatica*, 44:153–166, 2007.

[14] L. Kari and S. Seki. On pseudoknot-bordered words and their properties. *Journal of Computer and System Sciences*, 75:113–121, 2009.

[15] L. Kari and S. Seki. Towards the sequence design preventing pseudoknot formation. In *Prpc. 2nd International Workshop on Natural Computing (IWNC)*, pages 101–110, 2009.

[16] S. Kobayashi. Testing structure freeness of regular sets of biomolecular sequences (extended abstract). In C. Ferreti, G. Mauri, and C. Zandron, editors, *DNA 10*, volume 3384, pages 192–201. Springer, 2005.

[17] A. Lentin and M. P. Schützenberger. A combinatorial problem in the theory of free monoids. In *Combinatorial Mathematics and its Applications*, pages 128–144, 1967.

[18] H. Matsui, K. Sato, and Y. Sakakibara. Pair stochastic tree adjoining grammars for aligning and predicting pseudoknot RNA structures. In *2004 IEEE Computational Systems Bioinformatics Conference*, pages 1–11, 2004.

[19] J. S. McCaskill. The equilibrium partition function and base pair binding probability for rna secondary structure. *Biopolymers*, 29:1105–1119, 1990.

[20] E. Rivas and S. R. Eddy. The language of RNA: A formal grammar that includes pseudoknots. *Bioinformatics*, 16(4):334–340, 2000.

[21] G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages*, volume 1. Springer-Verlag, Berlin Heidelberg, 1997.

[22] H. J. Shyr. *Free Monoids and Languages.* Hon Min book company, Taichung, Taiwan, 3 edition, 2001.

[23] H. J. Shyr, G. Thierrin, and S. S. Yu. Monogenic e-closed languages and dipolar words. *Discrete Mathematics*, 126:339–348, 1994.

[24] Y. Uemura, A. Hasegawa, S. Kobayashi, and T. Yokomori. Tree adjoining grammars for RNA structure prediction. *Theoretical Computer Science*, 210:277–303, 1999.

[25] S. S. Yu. *Languages and Codes.* Tsang Hai Book Company Co., Taichung, Taiwan, 2005.

Chapter 8

Duplication on DNA sequences

This chapter presents results on a model of DNA duplication process, investigated by the author, Masami Ito, Lila Kari, and Zachary Kincaid. These results were summarized into the following paper[1]:

Masami Ito, Lila Kari, Zachary Kincaid, and Shinnosuke Seki.

Duplication in DNA Sequences.

In A. Condon, D. Harel, J. N. Kok, A. Salomaa, and E. Winfree, editors, *Algorithmic Bioprocesses*, *Natural Computing Series*, pages 43-61, Springer, 2009.

Its conference version was presented at the 12th International Conference on Developments in Language Theory (DLT 2008):

Masami Ito, Lila Kari, Zachary Kincaid, and Shinnosuke Seki.

Duplication in DNA Sequences.

[1]A version of this chapter has been published.

In M. Ito and M. Toyama, editors, *DLT 2008*, volume 5257 of *Lecture Notes in Computer Science*, pages 419-430, Springer, 2008.

The contents of this chapter are well-organized and more-deeply discussed in the undergraduate thesis of Zachary Kincaid with other interesting topics:

Zachary Kincaid.

Duplication and Decomposition of Formal Languages.

Undergraduate thesis, the University of Western Ontario, 2008.

Summary: The duplication and repeat-deletion operations are the basis of a formal language theoretic model of errors that can occur during DNA replication. During DNA replication, subsequences of a strand of DNA may be copied several times (resulting in duplications) or skipped (resulting in repeat-deletions). As formal language operations, iterated duplication and repeat-deletion of words and languages have been well-studied in the literature. However, little is known about single-step duplications and repeat-deletions. In this paper, we investigate several properties of these operations, including closure properties of language families in the Chomsky hierarchy and equations involving these operations. We also make progress towards a characterization of regular languages that are generated by duplicating a regular language.

Duplication in DNA sequences

Masami Ito[1], Lila Kari[2], Zachary Kincaid[3], and Shinnosuke Seki[2]

[1] Department of Mathematics, Faculty of Science, Kyoto Sangyo University, Kyoto, Japan, 603-8555.
[2] Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.
[3] Department of Computer Science, University of Toronto, Toronto, Ontario, M5S 2E4, Canada.

8.1 Introduction

Duplication grammars and duplication languages have recently received a great deal of attention in the formal language theory community. Duplication grammars, defined in [16], model duplication using string rewriting systems. Several properties of languages generated by duplication grammars were investigated in [16, 17]. Another prevalent model for duplication is a unary operation on words [2, 3, 8, 10, 12, 13]. The research on duplication is motivated by errors that occur during DNA[2] replication. Duplication and repeat-deletion (also called repeat expansion and repeat contraction, i.e., insertions and deletions of tandem repeating sequence) are biologically significant because they are among the most common errors that occur during DNA replication. In general insertions and deletions have been linked to cancer and more than 15 hereditary diseases [1]. They can also have positive consequences such as a contribution to the genetic functional compensation [5]. Interestingly, the mechanisms that cause insertions and deletions are not all well understood by geneticists

[2]A DNA single strand is a string over the DNA alphabet of bases $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. Due to the Watson-Crick complementarity property of bases, wherein $\mathtt{A}$ is complement to $\mathtt{T}$ and $\mathtt{C}$ is complement to $\mathtt{G}$, two DNA single strands of opposite orientation and exact complementary sequences can bind to each other to form a double DNA strand. This process is called base-pairing.

[4]. For example, the strand slippages at tandem repeats and interspersed repeats are well understood but the repeat expansion and contraction in tri-nucleotide repeat diseases remain unexplained.

Strand slippage is a prevalent explanation for the occurence of repeat expansions and repeat contractions during DNA replication. DNA replication is the process by which the DNA polymerase enzyme creates a new “nascent DNA strand” that is the complement of a given single strand of DNA referred to as the “template strand”. The replication process begins by mixing together the template DNA strand, the DNA polymerase enzyme, a special short DNA single strand called a “primer”, and sufficient individual bases that will be used as building blocks. The primer is specially designed to base-pair with the template and thus make it double-stranded for the length of the primer. The DNA polymerase will use the primer-template double-strand subsequence as a toe-hold, and will start adding complementary bases to the template strand, one by one, in one direction only, until the entire template strand becomes double-stranded. It has been observed that errors can happen during this process, the most common of them being insertions and deletions of bases. The current explanation is that these repeat expansions and repeat contractions are caused by misalignments between the template and nascent strand during replication [4]. DNA polymerase is not known to have any “memory” to remember which base on the template has been just copied onto the nascent strand, and hence the template and nascent strands can *slip*. As such, the DNA polymerase may copy a part of the template twice (resulting in an insertion) or forget to copy it (deletion). Repeat expansions and contractions occur most frequently on repeated sequences, so they are appropriately modelled by the rewriting rules $u \to uu$ and $uu \to u$, respectively.

The rule $u \to uu$ is a natural model for duplication, and the rule $uu \to u$ models the dual of duplication, which we call *repeat-deletion*. Since strand slippage is responsible for both these operations, it is natural to study both duplication and repeat-deletion. Repeat-deletion has already been extensively studied, e.g., in [11]. However, the existing literature addresses mainly the iterated application of both repeat-deletion and duplication. This paper investigates the effects of a *single* duplication or repeat-deletion. This restriction introduces subtle new complexities into languages that can be obtained as a duplication or repeat-deletion of a language.

This paper is organized as follows: in Section 8.2, we define terminology and notations to be used throughout the paper. Section 8.3 is dedicated to the closure properties of the language families of the Chomsky hierarchy under duplication and repeat-deletion. In Section 8.4, we present and solve language equations based on these operations, and give constructive solutions of the equation in the case involving duplication operation and regular languages. In Section 8.5, we introduce a generalization of duplication, namely controlled duplication. Section 8.6 investigates a characterization of the regular languages that can be obtained as a duplication of a regular language. When complete, such a characterization would constructively solve the language equation involving repeat-deletion and regular languages, for a certain class of languages. Lastly, in Section 8.7 we present some results on the relationship between duplication, repeat-deletion, and primitive words.

The conference version of this paper was published in [9].

8.2 Preliminaries

We now provide definitions for terms and notations to be used throughout the paper. For basic concepts in formal language theory, we refer the reader to [6, 7, 20, 22]. For a relation R, we denote by R^* the reflexive, transitive closure of R. Σ denotes a finite alphabet, Σ^* denotes the set of words over Σ, and Σ^+ denotes the set of words over Σ excluding the empty word λ. For a non-negative integer $n \geq 0$, Σ^n denotes the set of words of length n over Σ, and let $\Sigma^{\leq n} = \bigcup_{i=0}^{n} \Sigma^i$. The length of a word $w \in \Sigma^*$ is denoted by $|w|$. A language over Σ is a subset of Σ^*. For a language $L \subseteq \Sigma^*$, the set of all (internal) factors (resp. prefixes, suffixes) of L, are denoted by $\mathrm{Inf}(L)$ (resp. $\mathrm{Pref}(L)$, $\mathrm{Suff}(L)$). The complement of a language $L \subseteq \Sigma^*$, denoted by L^c, is defined as $L^c = \Sigma^* \setminus L$. We denote by FIN the family of all finite languages, by REG the family of all regular languages, by CFL the family of all context-free languages, and by CSL the family of all context-sensitive languages. We note that $\mathrm{FIN} \subsetneq \mathrm{REG} \subsetneq \mathrm{CFL} \subsetneq \mathrm{CSL}$.

For a finite automaton $A = (Q, \Sigma, \delta, s, F)$ (where Q is a state set, Σ is an alphabet, $\delta : Q \times \Sigma \to 2^Q$ is a transition function, $s \in Q$ is the start state, and

$F \subseteq Q$ is a set of final states), let $\mathcal{L}(A)$ denote the language accepted by A. We extend δ to $\hat{\delta} : Q \times \Sigma^* \to 2^Q$ as follows: (1) $\hat{\delta}(q, \lambda) = \{q\}$ for $q \in Q$ and (2) $\hat{\delta}(q, wa) = \cup_{p \in \hat{\delta}(q,w)} \delta(p, a)$ for $q \in Q$, $w \in \Sigma^*$, and $a \in \Sigma$. For $P_1, P_2 \subseteq Q$, we define an automaton $A_{(P_1,P_2)} = (Q \cup s_0, \Sigma, \delta', s_0, P_2)$, where $s_0 \notin Q$ is a new start state and $\delta' = \delta \cup (s_0, \lambda, P_1)$. Thus,

$$\mathcal{L}(A_{(P_1,P_2)}) = \{w \mid \hat{\delta}(p_1, w) \cap P_2 \neq \emptyset \text{ for some } p_1 \in P_1\}$$

If P_i is the singleton set $\{p_i\}$, then we may simply write p_i for $i \in \{1, 2\}$.

In this paper, we investigate two operations that are defined on words and extended to languages: *duplication* and *repeat-deletion*. We employ the duplication operation $\heartsuit$ described in [2], which is defined as follows:

$$u^{\heartsuit} = \{v \mid u = xyz, v = xyyz \text{ for some } x, y \in \Sigma^*, y \in \Sigma^+\}.$$

In the canonical way, the duplication operation is extended to a language $L \subseteq \Sigma^*$:

$$L^{\heartsuit} = \bigcup_{u \in L} u^{\heartsuit}.$$

We also define another unary operation based on the dual of the $\heartsuit$ operation. We term this operation *repeat-deletion* and denote it by $\spadesuit$. Note that while biologists refer to this process simply as deletion, in formal language theory, the term deletion typically refers to removing arbitrary (rather than repeated) factors of word.

Definition 2. For a word $v \in \Sigma^*$, the language generated by repeat-deletion of v is defined

$$v^{\spadesuit} = \{u \mid v = xyyz, u = xyz \text{ for some } x, z \in \Sigma^*, y \in \Sigma^+\}.$$

Again, the repeat-deletion operation is extended to languages: for a given language $L \subseteq \Sigma^*$,

$$L^{\spadesuit} = \bigcup_{v \in L} v^{\spadesuit}$$

We avoid inverse notation because $\heartsuit$ and $\spadesuit$ are not inverses when considered as operations on languages. That is, for a language $L \subseteq \Sigma^*$, $L \subseteq (L^{\heartsuit})^{\spadesuit}$ but it is not always the case that $L = (L^{\heartsuit})^{\spadesuit}$.

Example 31. Let $L = a^*bb$. Then $abb \in L \Rightarrow aabb \in L^{\heartsuit}$. Therefore $aab \in (L^{\heartsuit})^{\spadesuit}$, but $aab \notin L$.

Previous work focussed on the reflexive transitive closure of the duplication operation, which we will refer to as duplication closure. All occurrences of $\heartsuit$, duplication, $\spadesuit$, and repeat-deletion refer to the *single step* variations of the operations.

8.3 Closure Properties

Much of the work on duplication has been concerned with determining which of the families of languages on the Chomsky hierarchy are closed under duplication closure. It is known that, on a binary alphabet, the family of regular languages is closed under duplication closure. In contrast, on a larger alphabet, REG is still closed under n-bounded duplication closure for $n \leq 2$, but REG is not closed under n-bounded operation closure for any $n \geq 4$. The family of context-free languages is closed under (uniformly) bounded duplication closure. The readers are refered to [8] for these results.

It is a natural first step to determine these closure properties under (single step) duplication. In this section, we show that the family of regular languages is closed under repeat-deletion but not duplication, the family of context-free languages is not closed under either operation, and the family of context-sensitive languages is closed under both operations.

The following two propositions are due to [21] (without proofs):

Proposition 8.1. *The family of regular languages is not closed under duplication.*

Proof. Let $L = ab^*$ and suppose that $L^\heartsuit$ is regular. Since the family of regular languages is closed under intersection, $L' = L^\heartsuit \cap ab^*ab^*$ is regular. But L' is exactly the language $\{ab^iab^j : i \leq j\}$, which is clearly not regular. So by contradiction, $L^\heartsuit$ is not regular, and the family of regular languages is not closed under duplication. $\square$

Note that the proof of the preceding proposition requires that the alphabet contain at least two letters. As we shall see in Section 8.6, this bound is tight: the family of regular languages over a unary alphabet is closed under duplication.

Proposition 8.2. *The family of context-free languages is not closed under duplication.*

Proof. Let $L = \{a^i b^i \mid i \geq 1\}$, a context-free language. Suppose $L^\heartsuit$ is context-free. Since the family of context-free languages is closed under intersection with regular languages, $D = L^\heartsuit \cap \{a^* b^* a^* b^*\}$ is context free.

Let p be the pumping-lemma constant of the language D. Consider the word $z = a^p b^p a^p b^p \in D$. We can decompose z as $z = uvwxy$ such that vx is a pumped part. Let $z_i = uv^i wx^i y$. Firstly, v must not contain both a and b; otherwise pumping v results in a word with more than two repetitions of $a^i b^j$ for some $i, j \geq 1$. This also applies to x. Secondly, vx must be within the central $b^p a^p$ part; otherwise, the pumped vx causes a difference between the number of first as and the number of last bs. Now we know that vwx is within the central $b^p a^p$ part of z, and $v = b^i$ and $x = a^j$ for some $0 \leq i, j \leq p$ (with i, j not both zero). Then $z_2 = a^p b^{p+i} a^{p+j} b^p$, which can not be generated by duplication of a word in L. Thus we conclude that $L^\heartsuit$ is not context-free. □

Proposition 8.3. *The family of context-sensitive languages is closed under duplication.*

Proof. Let L be a context-sensitive language, and A_L be a linear bounded automaton for L. Now we construct a Turing machine $A_\heartsuit$ for $L^\heartsuit$ and show that $A_\heartsuit$ is a linear bounded automaton. Indeed, for a given input $w \in \Sigma^*$, $A_\heartsuit$ nondeterministically choose $w' \in \text{Inf}(w)$ (let $w = xw'z$ for some $x, z \in \Sigma^*$) and checks whether $w' = yy$ for some $y \in \Sigma^*$. If not, it turns down this choice. Otherwise, it deletes one of y so that the input tape has xyz. Now $A_\heartsuit$ simulates A_L on this tape, and if A_L accepts the given input, xyz, then $A_\heartsuit$ accepts $w = xyyz$. Therefore, $A_\heartsuit$ accepts w if and only if there exists a nondeterministic choice of the infix with respect to which the simulated A_L accepts the given input. Thus, $\mathcal{L}(A_\heartsuit) = L^\heartsuit$.

This construction has four steps; the choice of an infix of an input, check of whether the infix is repetitive, deletion, and the simulation of A_L. The first three steps require the workspace linear-proportional to the length of an input. In the fourth step, A_L receives an input which is shorter than the original input to $A_\heartsuit$ and A_L is a linear bounded automaton. As a result, $A_\heartsuit$ is also a linear bounded automaton. $\square$

In the following, we consider the closure properties of the language families in the Chomsky hierarchy under repeat-deletion. Our first goal is to prove that the family of regular languages is closed under repeat-deletion. For this purpose, we define the following binary operation $\natural$ on languages $L, R \subseteq \Sigma^*$:

$$L \natural R = \{xyz \mid xy \in L, yz \in R, y \neq \lambda\}.$$

Proposition 8.4 (Due to Z. Ésik). *The family of regular languages is closed under* $\natural$.

Proof. Let $L_1, L_2 \subseteq \Sigma^+$ be regular languages. Let $\#$ be a new letter (not in Σ) and let h be homomorphism defined by $h(a) = a$ for $a \in \Sigma^*$ and $h(\#) = \lambda$. Let $L_1' = L_1 \leftarrow \{\#\} = \{u\#v \mid uv \in L_1\}$ ($\leftarrow$ denotes the insertion operation) and $L_2' = L_2 \leftarrow \{\#\}$. Moreover, let $\overline{L_1} = L_1'\#\Sigma^*$ and let $\overline{L_2} = \Sigma^*\#L_2'$. Then $L_1 \natural L_2 = h(\overline{L_1} \cap \overline{L_2})$. Since the family of regular languages is closed under insertion, concatenation, intersection, and homomorphism, $L_1 \natural L_2$ is regular. $\square$

Let L be a regular language. We can construct a finite automaton $A = (Q, \Sigma, \delta, s, F)$ such that $\mathcal{L}(A) = L$. Recall that for any state $q \in Q$, $\mathcal{L}(A_{(s,q)}) = \{w : sw \vdash_A^* q\}$ and $\mathcal{L}(A_{(q,F)}) = \{w : \exists f \in F \text{ such that } qw \vdash_A^* f\}$. Intuitively, $\mathcal{L}(A_{(s,q)})$ is the set of words accepted "up to q", and $\mathcal{L}(A_{(q,F)})$ is the set of words accepted "after q" so that $\mathcal{L}(A_{(s,q)})\mathcal{L}(A_{(q,F)}) \subseteq L$ is the set of words in L that have a derivation that passes through state q.

Lemma 8.5. *Let L be a regular language and $A = (Q, \Sigma, \delta, s, F)$ be a finite automaton accepting L. Then $L^\spadesuit = \bigcup_{q \in Q} \mathcal{L}(A_{(s,q)}) \natural \mathcal{L}(A_{(q,F)})$.*

Proof. Let $L' = \bigcup_{q \in Q} \mathcal{L}(A_{(s,q)}) \natural \mathcal{L}(A_{(q,F)})$. First we prove that $L^\spadesuit \subseteq L'$. Let $\alpha \in L^\spadesuit$. Then there exists a decomposition $\alpha = xyz$ for some $x, y, z \in \Sigma^*$ such that $xyyz \in L$ and $y \neq \lambda$. Since A accepts $xyyz$, there exists some $q \in Q$ such that $sxyyz \vdash^* qyz$ and $qyz \vdash^* f$ for some $f \in F$. By construction, $xy \in \mathcal{L}(A_{(s,q)})$ and $yz \in \mathcal{L}(A_{(q,F)})$. This implies that $xyz \in \mathcal{L}(A_{(s,q)}) \natural \mathcal{L}(A_{(q,F)})$, from which we have $L^\spadesuit \subseteq L'$.

Conversely, if $\alpha \in L'$, then there exists $q \in Q$ such that $\alpha \in \mathcal{L}(A_{(s,q)}) \natural \mathcal{L}(A_{(q,F)})$.

We can decompose α into xyz for some $x, y, z \in \Sigma^*$ such that $xy \in \mathcal{L}(A_{(s,q)})$, $yz \in \mathcal{L}(A_{(q,F)})$, and $y \neq \lambda$. Since $\mathcal{L}(A_{(s,q)})\mathcal{L}(A_{(q,F)}) \subseteq L$, we have that $xyyz$ belongs to L. It follows that $\alpha = xyz \in L^{\spadesuit}$ and $L' \subseteq L^{\spadesuit}$. We conclude that $L' = L^{\spadesuit}$. $\square$

Proposition 8.6. *The family of regular languages is closed under repeat-deletion.*

Proof. Since the family of regular languages is closed under finite union and the $\natural$ operation, it is closed under repeat-deletion (due to Lemma 8.5). $\square$

Proposition 8.7. *The family of context-free languages is closed under $\natural$ with regular languages.*

Proof. Repeat the argument in the proof for Proposition 8.4. Since the family of context-free languages is closed under insertion, concatenation with regular languages, intersection with regular languages, and homomorphism, the family of context-free languages is closed under $\natural$ with regular languages. $\square$

Lemma 8.8. *The family of context-free languages is not closed under $\natural$.*

Proof. Let $L_1 = \{a^i\#b^i\$ \mid i \geq 0\}$ and $L_2 = \{\#b^j\$c^j \mid j \geq 0\}$. Although L_1 and L_2 are CFLs, $L_1 \natural L_2 = \{a^i\#b^i\$c^i \mid i \geq 0\}$, which is not context-free. $\square$

Proposition 8.9. *The family of context-free languages is not closed under repeat-deletion.*

Proof. Let $L = \{a^i\#b^i\#b^jc^j \mid i, j \geq 0\}$, which is context-free. Then $L^{\spadesuit} \cap a^*\#b^*c^* = \{a^i\#b^jc^j \mid i, j \geq 0, i \leq j\}$, which is not context free. Since the family of context-free languages is closed under intersection with regular languages, and since $L^{\spadesuit} \cap a^*\#b^*c^*$ is not context-free, we may conclude that $L^{\spadesuit}$ is not context free. Thus, the family of context-free languages is not closed under repeat-deletion. $\square$

However, there do exist context-free (and non-regular) languages whose image under repeat deletion remains context-free. An example is shown below.

Example 32. Let $L = \{a^n b^n \mid n \geq 0\}$; this is a context-free language. Then $L^{\spadesuit} = \{a^n b^m \mid 1 \leq m < n \leq 2m\} \cup \{a^n b^m \mid 1 \leq n < m \leq 2n\}$. This $L^{\spadesuit}$ is generated by the following context-free grammar, and hence in CFL. Let $G = (\{a, b\}, \{S, X, Y, X_f, Y_f\}, P, S)$, where the set of production rules P is given by

$$\begin{aligned}
S &\rightarrow X \mid Y, \\
X &\rightarrow aXb \mid aaX_f b, \\
Y &\rightarrow aYb \mid aY_f bb, \\
X_f &\rightarrow aX_f b \mid aaX_f b \mid \lambda, \\
Y_f &\rightarrow aY_f b \mid aY_f bb \mid \lambda,
\end{aligned}$$

Proposition 8.10. *The family of context-sensitive languages is closed under repeat-deletion.*

Proof. Let L and A_L be defined as we did in Proposition 8.3. As $A_{\heartsuit}$ in the proposition, we construct a linear bounded automaton $A_{\spadesuit}$ for $L^{\spadesuit}$ which simulates A_L. In contrast to $A_{\heartsuit}$, $A_{\spadesuit}$ nondeterministically copies an infix of a given input w. Formally

	$\heartsuit$	$\spadesuit$	$\natural$	$\natural$ with regular
FIN	Y	Y	Y	N
REG	N	Y	Y	Y
CFL	N	N	N	Y
CSL	Y	Y	Y	Y

Table 8.1: Closure properties of several language classes under duplication, repeat-deletion, and the $\natural$ operation

speaking, w is regarded as a catenation of x, y, z and y is duplicated so as to result in $xyyz$ on the input tape. Then $A_\spadesuit$ runs A_L on the tape. If A_L accepts $xyyz$, then $A_\spadesuit$ accepts $w = xyz$. As shown in Proposition 8.3, $A_\spadesuit$ is a linear bounded automaton. $\square$

In summary, the following closure properties related to duplication, repeat-deletion, and the $\natural$ operation hold:

8.4 Language Equations

We now consider the language equation problem posed by the duplication operation: for a given language $L \subseteq \Sigma^*$, can we find a language $X \subseteq \Sigma^*$ such that $X^\heartsuit = L$? In the following, we show that, if L is a regular language and there exists a solution to $X^\heartsuit = L$, then we can compute a maximal solution. We note that the solution to the language equation is not unique in general.

Example 33. $\{aaa, aaaa, aaaaa\}^\heartsuit = \{aaa, aaaaa\}^\heartsuit = \{a^i : 4 \leq i \leq 10\}$

In view of the fact that a language equation may have multiple solutions, we define an equivalence relation $\sim_{\heartsuit}$ on languages as follows:

$$X \sim_{\heartsuit} Y \Leftrightarrow X^{\heartsuit} = Y^{\heartsuit}.$$

For the same reason, we define an equivalence relation $\sim_{\spadesuit}$ as follows:

$$X \sim_{\spadesuit} Y \Leftrightarrow X^{\spadesuit} = Y^{\spadesuit}.$$

Lemma 8.11. *If $[X] \in 2^{\Sigma^*}/\sim_{\heartsuit}$ and if $\Xi \subseteq [X]$ ($\Xi \neq \emptyset$), then $\bigcup_{L\in\Xi} L \in [X]$.*

Proof. Let $[X] \in 2^{\Sigma^*}/\sim_{\heartsuit}$ and $\Xi \subseteq [X]$ ($\Xi \neq \emptyset$). Prove that $L_{\Xi} = \bigcup_{L\in\Xi} L \in [X]$.

Let $Y \in \Xi$. Clearly, $Y \subseteq L_{\Xi}$ and so $Y^{\heartsuit} \subseteq {L_{\Xi}}^{\heartsuit}$. Now let $w \in {L_{\Xi}}^{\heartsuit}$. Then $\exists x, z \in \Sigma^*, y \in \Sigma^+, v \in L_{\Xi}$ such that $w = xyyz$ and $v = xyz$. Then there exists $Z \in \Xi$ such that $v \in Z$. Since $Y, Z \in \Xi$, $v^{\heartsuit} \subseteq Z^{\heartsuit} = Y^{\heartsuit}$. Then $w \in v^{\heartsuit}$ implies $w \in Y^{\heartsuit}$. Thus, ${L_{\Xi}}^{\heartsuit} \subseteq Y^{\heartsuit}$. We conclude that $Y^{\heartsuit} = {L_{\Xi}}^{\heartsuit}$ and $L_{\Xi} \in [X]$. □

Corollary 8.12. *For an equivalence class $[X] \in 2^{\Sigma^*}/\sim_{\heartsuit}$, there exists a unique maximal element $X_{\max}$ with respect to the set inclusion partial order defined as follows:*

$$X_{\max} = \bigcup_{L\in[X]} L.$$

We provide a way to construct the maximum element of a given equivalence class. First, we prove a more general result.

Proposition 8.13. *Let $L \subseteq \Sigma^*$, and let $f, g : \Sigma^* \to 2^{\Sigma^*}$ be any functions such that $u \in g(v) \Leftrightarrow v \in f(u)$ for all $u, v \in \Sigma^*$. If a solution to the language equation $\bigcup_{x \in X} f(x) = L$ exists, then the maximum solution (with respect to the set inclusion partial order) is given by $X_{\max} = \left(\bigcup_{y \in L^c} g(y)\right)^c$.*

Proof. For two languages $X, Y \subseteq \Sigma^*$ such that $\bigcup_{x \in X} f(x) = L$ and $\bigcup_{y \in Y} f(y) = L$, $\bigcup_{z \in X \cup Y} f(z) = L$ holds. Hence the assumption implies the existence of $X_{\max}$.

($\subseteq$) Suppose $\exists w \in g(v) \cap X_{\max}$ for some $v \in L^c$. This means that $v \in f(w)$. However, $f(w) \subseteq \bigcup_{x \in X_{\max}} f(x) = L$, and hence $v \in L$, a contradiction. ($\supseteq$) Suppose that $\exists w \in X_{\max}^c \cap (\bigcup_{y \in L^c} g(y))^c$. If $f(w) \subseteq L$, then $w \in X_{\max}$ (by the maximality of $X_{\max}$). Otherwise, $\exists v \in f(w) \cap L^c$. This implies that $w \in g(v) \subseteq \bigcup_{y \in L^c} g(y)$. In both cases, we have a contradiction. Therefore, we have $X_{\max}^c = \bigcup_{y \in L^c} g(y)$, i.e., $X_{\max} = (\bigcup_{y \in L^c} g(y))^c$. □

Lemma 8.14. *Let $u, v \in \Sigma^*$. Then $u \in v^{\heartsuit}$ if and only if $v \in u^{\spadesuit}$.*

Proof. ($\Rightarrow$) If $u \in v^{\heartsuit}$, then there exist $x, z \in \Sigma^*$ and $y \in \Sigma^+$ such that $v = xyz$ and $u = xyyz$. Then $u^{\spadesuit}$ contains $xyz = v$. ($\Leftarrow$) If $v \in u^{\spadesuit}$, then there exist $x', z' \in \Sigma^*$ and $y' \in \Sigma^+$ such that $v = x'y'z'$ and $u = x'y'y'z'$. Then $x'y'y'z' = u \in v^{\heartsuit}$. □

Proposition 8.13 and Lemma 8.14 imply the following corollaries.

Corollary 8.15. *Let $L \subseteq \Sigma^*$. If there exists a language $X \subseteq \Sigma^*$ such that $X^{\spadesuit} = L$, then the maximum element $X_{\max}$ of $[X]_{\sim_{\spadesuit}}$ is given by $((L^c)^{\heartsuit})^c$.*

Corollary 8.16. *Let $L \subseteq \Sigma^*$. If there exists a language $X \subseteq \Sigma^*$ such that $X^{\heartsuit} = L$, then the maximum element $X_{\max}$ of $[X]_{\sim_{\heartsuit}}$ is given by $((L^c)^{\spadesuit})^c$.*

Proposition 8.17. *Let L, X be regular languages satisfying $X^{\heartsuit} = L$. Then it is decidable whether X is the maximal solution for this language equation.*

Proof. Since L is regular and REG is closed under repeat-deletion and complement, the maximum solution of $X^{\heartsuit} = L$ given in Corollary 8.16, $((L^c)^{\spadesuit})^c$, is regular. Since the equivalence problem for regular languages is decidable, it is decidable whether a given solution to the duplication language equation is maximal. $\square$

Due to the fact that REG is not closed under duplication, we cannot obtain a similar decidability result for the $X^{\spadesuit} = L$ language equation. This motivates our investigation in the next two sections of necessary and sufficient conditions for the duplication of a regular language to be regular. Indeed, in the cases when the duplication language $(L^c)^{\heartsuit}$ is regular, the solution to language equations $X^{\spadesuit} = L$, $L \in$ REG, can be constructed as described in Corollary 8.15.

8.5 Controlled Duplication

In Section 8.4 we showed that for a given language $L \subseteq \Sigma^*$, the maximal solution of the repeat-deletion language equation $X^{\spadesuit} = L$ is given by $((L^c)^{\heartsuit})^c$. However, unlike the duplication language equation, we do not have an efficient algorithm to compute this language due to the fact that the family of regular languages is not closed under duplication. This motivates "controlling" the duplication in such a manner that duplications can occur only for some specific words.

Let L, C be languages over Σ. We define the duplication of L using the control set C as follows:

$$L^{\heartsuit(C)} = \{xyyz \mid xyz \in L, y \in C\}.$$

Note that this generalization of the duplication operation can express two variants of duplication that appear in previous literature, namely uniform and length-bounded duplication ([12, 13]). Indeed, using the notation in [13], we have

$$D^1_{\{n\}}(L) = L^{\heartsuit(\Sigma^n)} \text{ and } D^1_{\{0,1,\ldots,n\}}(L) = L^{\heartsuit(\Sigma^{\leq n})}.$$

This section presents basic properties of controlled duplications, some of which will turn out to be useful in Section 8.6. For symmetry, we also investigate properties of controlled repeat-deletion.

Lemma 8.18. *Let $L \subseteq \Sigma^*$ be a language and $C_1, C_2 \subseteq \Sigma^*$ be control sets. If $C_1 \subseteq C_2$, then $L^{\heartsuit(C_1)} \subseteq L^{\heartsuit(C_2)}$.*

Lemma 8.19. *Let $L \subseteq \Sigma^*$ be a language and $C_1, C_2 \subseteq \Sigma^*$ be control sets. Then $L^{\heartsuit(C_1 \cup C_2)} = L^{\heartsuit(C_1)} \cup L^{\heartsuit(C_2)}$.*

Let $L \subseteq \Sigma^*$ be a language, $C \subseteq \Sigma^*$ be a control set, and $w \in C$. Then w is said to be *useful with respect to* L if $w \in \mathrm{Inf}(L)$; otherwise, it is called *useless* with respect to L. The control set C is said to *contain an infinite number of useful words with respect to* L if and only if $|\mathrm{Inf}(L) \cap C| = \infty$.

Lemma 8.20. *Let $L \subseteq \Sigma^*$ be a language, $C \subseteq \Sigma^*$ be a control set, and C' be the set of all useless words in C with respect to L. Then $L^{\heartsuit(C)} = L^{\heartsuit(C \setminus C')}$.*

Proof. Lemma 8.19 implies $L^{\heartsuit(C)} = L^{\heartsuit(C \setminus C')} \cup L^{\heartsuit(C')}$. Since $L^{\heartsuit(C')} = \emptyset$, $L^{\heartsuit(C)} = L^{\heartsuit(C \setminus C')}$ □

Proposition 8.21. *For a regular language $L \subseteq \Sigma^*$ and a regular control set $C \subseteq \Sigma^*$, it is decidable whether C contains an infinite number of useful words with respect to L.*

Proof. Since L and C are regular, $\mathrm{Inf}(L)$ and hence $\mathrm{Inf}(L) \cap C$ are also regular. Since finiteness of a regular language is decidable, it is decidable whether or not a regular control set C contains an infinite number of useful words with respect to a language L. □

Note that if $L \subseteq \Sigma^*$, $C \subseteq \Sigma^*$ is a control set, and C contains at most a finite number of useful words with respect to L, then $C' = C \cap \mathrm{Inf}(L)$ is a finite language and satisfies $L^{\heartsuit(C)} = L^{\heartsuit(C')}$. In particular, for any finite language L and any control set C, there exists a finite control set $C' \subseteq C$ satisfying $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.

We now extend our previous results on the closure properties of language families so as to accommodate the controlled duplication. Since $\heartsuit = \heartsuit_{\Sigma^*}$, we trivially have the following:

- The family of regular languages is not closed under controlled duplication.
- The family of context-free languages is not closed under controlled duplication, repeat-deletion, or $\natural$.

We conclude this section with definions of repeat-deletion and the $\natural$ operation using control sets, and by providing a few results of them.

Let $L, L_1, L_2, C \subseteq \Sigma^*$. Then

$$\begin{aligned} L^{\spadesuit(C)} &= \{xyz \mid xyyz \in L, y \in C\}, \\ L_1 \natural_C L_2 &= \{xyz \mid xy \in L_1, yz \in L_2, y \in C\}. \end{aligned}$$

It is straightforward to prove that the family of regular languages is closed under $\natural_C$ for any regular language C. Let L_1, L_2 be regular languages and form $\overline{L_1}$ and $\overline{L_2}$ as defined in the proof of Proposition 8.4. We see that $L_1 \natural_C L_2 = h(\overline{L_1} \cap \overline{L_2} \cap \Sigma^* \# C \# \Sigma^*)$. Furthermore, by repeating the argument in the proof of Proposition 8.6, we have that the family of regular languages is closed under $\spadesuit_C$ for any regular control set C.

It is simple to check that if each word in L contains a subword that is in C, $\heartsuit_C$

and $\spadesuit_C$ satisfy the requirements of Proposition 8.13, so that we have a procedure to find X such that $X^{\heartsuit(C)} = L$ if such an X exists.

Proposition 8.22. *Let $L \subseteq \Sigma^*$ be a context-free language and let $C \subseteq \Sigma^+$ be a finite control set. Then $L^{\spadesuit(C)}$ is context-free.*

Proof. Let h be the homomorphism defined by $h(a) = h(\overline{a}) = a$ for $a \in \Sigma, \overline{a} \in \overline{\Sigma}$. Then $L' = h^{-1}(L)$ is context-free. Consider $L'' = L' \cap (\Sigma^*\{u\overline{u} \mid u \in C\}\Sigma^*)$. Then L'' is context-free. Now let θ be the homomorphism defined by $\theta(a) = a$ and $\theta(\overline{a}) = \lambda$ for $a \in \Sigma$. Then $\theta(L'') = L^{\spadesuit(C)}$ and hence $L^{\spadesuit(C)}$ is context-free. $\square$

8.6 Conditions for $L^{\heartsuit(C)}$ to be Regular

For a regular language L and a control set C, we now investigate a necessary and sufficient condition for $L^{\heartsuit(C)}$ to be regular. As suggested in the following example, even for a "simple" language L and a control set C, $L^{\heartsuit(C)}$ can be non-regular.

Example 34. Let $\Sigma = \{a, b\}$ and $L = \{w \in \Sigma^* \mid |w| = 0 \pmod 3\}$ and $C = \Sigma^*$. Then $L^{\heartsuit(C)} \notin \text{REG}$.

Given a regular language L, a sufficient condition for $L^{\heartsuit(C)}$ to be regular is a corollary of the following result in [3]. A family of languages is called a *trio* if it is closed under λ-free homomorphism, inverse homomorphism, and intersection with regular languages. Note that both the families of regular languages and of context-free languages are trio.

Theorem 8.23 ([3]). *Any trio is closed under duplication with a finite control set.*

Corollary 8.24. *Let $L \subseteq \Sigma^*$ be a regular language and $C \subseteq \Sigma^*$. If there exists a finite control set $C' \subseteq \Sigma^*$ such that $L^{\heartsuit(C)} = L^{\heartsuit(C')}$, then $L^{\heartsuit(C)}$ is regular.*

Given a regular language L, we now investigate necessary conditions for $L^{\heartsuit(C)}$ to be regular. Results in [19] stating that infinite repetitive languages cannot be even context-free indicate that the converse of Corollary 8.24 may also be true. Hence, in the remainder of this section we shall investigate the following claim:

Claim 8.25. *Let $L \subseteq \Sigma^*$ be a regular language and $C \subseteq \Sigma^*$ be a control set. If $L^{\heartsuit(C)}$ is regular then there exist a finite control set $C' \subseteq \Sigma^*$ such that $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.*

As shown in the following example, this claim generally does not hold.

Example 35. Let $\Sigma = \{a, b\}$, $L = ba^+b$, and $C = ba^+ \cup a^+b$. We can duplicate a prefix ba^i of a word $ba^jb \in L$ $(i \leq j)$ to obtain a word $ba^iba^jb \in L^{\heartsuit(C)}$. In the same way, the duplication of a suffix $a^\ell b$ of a word ba^kb $(k \geq \ell)$ results in a word $ba^kba^\ell b \in L^{\heartsuit(C)}$. Thus $L^{\heartsuit(C)} = ba^+ba^+b$. Note that L and $L^{\heartsuit(C)}$ are regular. However there exists no finite control set C' satisfying $L^{\heartsuit(C)} = L^{\heartsuit(C')}$. This is because ba^+ba^+b can have arbitrary long repetitions of a's, and hence arbitrary long control factors are required to generate it.

Nevertheless this claim holds for several interesting cases: the case where L is finite or C contains at most a finite number of useful words with respect to L, the case of a unary alphabet $\Sigma = \{a\}$, the case $L = \Sigma^*$, and the case where the control set is "marked", i.e. there exists $a \in \Sigma$ such that $C \subseteq a(\Sigma \setminus \{a\})^*a$. Moreover, it turned out that the proof technique we employ for this fourth case can be utilized to prove that the claim holds for the case where C is nonoverlapping and an infix code, which is more general than the fourth case. In the following, we prove the direct implication of the claim for these cases (the reverse one is clear from Corollary 8.24).

In the case where L is finite, $L^{\heartsuit(C)}$ is finite and hence regular. Since $\mathrm{Inf}(L)$ is finite, by letting $C' = C \cap \mathrm{Inf}(L)$, we have $L^{\heartsuit(C)} = L^{\heartsuit(C')}$. Thus the claim holds for this case. Moreover, even for an infinite L, we can say that if C contains at most a finite number of useful words with respect to L, then the claim holds because C', defined in the same manner as above, is finite. Therefore in the following we assume that L is infinite and C contains an infinite number of useful words with respect to L.

Next, we show that the claim holds in the case of a unary alphabet. We employ the following known result for this purpose.

Proposition 8.26 ([6]). *Let $\Sigma = \{a\}$ be a unary alphabet, and L be a language over Σ. L is regular if and only if there exists a finite set $\mathcal{N}$ of pairs of integers such that $L = \bigcup_{k \geq 0, (n,m) \in \mathcal{N}} a^{kn+m}$.*

Proposition 8.27. *Let Σ be a unary alphabet, say $\Sigma = \{a\}$, $L \subseteq \Sigma^*$ be a regular language, and $C \subseteq \Sigma^*$ be an arbitrary language. Then $L^{\heartsuit(C)}$ is regular, and there* *exists a finite context* $C' \in \mathrm{FIN}$ *such that* $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.

Proof. L being regular, there exists a finite set of pairs of integers $\mathcal{N} = \{(p_i, q_i) \mid p_i, q_i \in \mathbb{N}_0, 1 \leq i \leq n\}$ for some $n \in \mathbb{N}$ such that $L = \bigcup_{x \geq 0, (p_i, q_i) \in \mathcal{N}} a^{p_i x + q_i}$.

Let $L_i = \bigcup_{x \geq 0} a^{p_i x + q_i}$, and consider a word $a^k \in C$, where $k \in \mathbb{N}$. For some $x \geq 0$, we can apply the duplication with respect to a^k to $a^{p_i x + q_i}$ if and only if $p_i x + q_i \geq k$. The application generates $a^{p_i x + q_i + k} \in L^{\heartsuit(C)}$. Note that for $x_1, x_2 \in \mathbb{N}_0$, $p_i x_1 + q_i + k = p_i x_2 + q_i + k \pmod{p_i}$. We define a function $\psi_i : C \mapsto \{0, 1, \ldots, p_i - 1\}$ such that for $a^k \in C$, $\psi_i(a^k) = q_i + k \pmod{p_i}$. Hence, we can partition C into p_i disjoint sets depending on ψ_i. Formally speaking, $C = \bigcup_{0 \leq m < p_i} C_{i,m}$, where $C_{i,m} = \{w \in C \mid \psi_i(w) = m\}$. Now the necessary and sufficient condition mentioned above as to the applicability implies that for $a^j, a^k \in C_{i,m}$, if $j \leq k$, then $L_i^{\heartsuit(\{a^j\})} \supseteq L_i^{\heartsuit(\{a^k\})}$. Let $w_{i,m}$ be the shortest word in $C_{i,m}$. Then $L_i^{\heartsuit(\{w_{i,m}\})} = L_i^{\heartsuit(C_{i,m})}$ holds. Thus, by letting $C' = \{w_{i,m} \mid 1 \leq i \leq n, 0 \leq m < p_i\}$, we have $L^{\heartsuit(C)} = L^{\heartsuit(C')}$. Clearly C' is finite, and hence $L^{\heartsuit(C')}$ is regular. $\square$

By letting $C = \Sigma^*$, Proposition 8.27 implies that the family of regular languages is closed under duplication when Σ is unary.

Next we show that the claim holds for the case when $L = \Sigma^*$ (Corollary 8.32). This requires the following known two lemmata. A word $w \in \Sigma^+$ is said to be *primitive* if $w = v^n$ implies that $n = 1$, i.e., $w = v$. A word $v \in \Sigma^+$ is called a *conjugate* of w if $v = xy$ and $w = yx$ for some $x, y \in \Sigma^*$.

Lemma 8.28 ([14]). *For a primitive word p, any conjugate of p is primitive.*

Lemma 8.29 ([15]). *Let p and q be primitive words with $p \neq q$ and let $i, j \geq 2$. Then $p^i q^j$ is primitive.*

For a language $C \subseteq \Sigma^*$, we define $\mathrm{Dup}(C) = \{ww \mid w \in C\}$.

Proposition 8.30. *Let $C \subseteq \Sigma^*$. Then $\Sigma^*\mathrm{Dup}(C)\Sigma^*$ is regular if and only if there exists a finite language C' such that $\Sigma^*\mathrm{Dup}(C')\Sigma^* = \Sigma^*\mathrm{Dup}(C)\Sigma^*$.*

Proof. The proof of 'if'-part is obvious since $\Sigma^*\mathrm{Dup}(C')\Sigma^*$ is regular. Now consider the proof of 'only if'-part. Assume $L = \Sigma^*\mathrm{Dup}(C)\Sigma^*$ is regular and consider the regular language $L \cap (\Sigma^* \setminus L\Sigma^+) \cap (\Sigma^* \setminus \Sigma^+ L)$. All words in this language have a representation ww for some $w \in C$. Hence there exists $C' \subseteq C$ such that $\mathrm{Dup}(C') = L \cap (\Sigma^* \setminus L\Sigma^+) \cap (\Sigma^* \setminus \Sigma^+ L)$. Notice that for any $w \in C$ there exist $w' \in C'$ and $x, y \in \Sigma^*$ such that $ww = xw'w'y$. Therefore, $\Sigma^*\mathrm{Dup}(C)\Sigma^* = \Sigma^*\mathrm{Dup}(C')\Sigma^*$.

Suppose C' is infinite. Then there exists a word $uu \in \mathrm{Dup}(C')$ with length twice that of the pumping lemma constant for $\mathrm{Dup}(C')$. So by the pumping lemma, there exists a decomposition $uu = u_1u_2u_3u_1u_2u_3$, of uu such that $u_1, u_3 \in \Sigma^*$, $u_2 \in \Sigma^+$ and $u_1u_2^iu_3u_1u_2u_3 \in \mathrm{Dup}(C')$ for any $i \in \mathbb{N}$. Notice that for any $i \in \mathbb{N}$, $u_1u_2^iu_3u_1u_2u_3$ is not primitive because it is in $\mathrm{Dup}(C')$. Consider the case $i \geq 3$. By Lemma 8.28, $u_2^{i-1}(u_2u_3u_1)^2$ is not primitive. Then Lemma 8.29 implies that u_2 and $u_2u_3u_1$ share a primitive root, say $p \in \Sigma^+$. We may now write $u_2 = p^n$ and $u_2u_3u_1 = p^m$ for some $n, m \geq 1$. Hence $u_2^{i-1}(u_2u_3u_1)^2 = p^{n(i-1)+2m}$. From Lemma 8.28, it follows that

$u_1u_2^iu_3u_1u_2u_3 = q^{n(i-1)+2m}$, where q is a conjugate word of p. Now we have that $u_1u_2^iu_3u_1u_2u_3 = q^{n(i-1)+2m}$ is a proper prefix (and suffix) of $u_1u_2^{i+1}u_3u_1u_2u_3 = q^{ni+2m}$, which contradicts with the definition of $\mathrm{Dup}(C')$. Thus C' must be finite. $\square$

Lemma 8.31. *Let $C \subseteq \Sigma^*$. Then $(\Sigma^*)^{\heartsuit(C)} = \Sigma^*\mathrm{Dup}(C)\Sigma^*$.*

Proof. Let $w \in (\Sigma^*)^{\heartsuit(C)}$. Then there exist $x, y, z \in \Sigma^*$ such that $y \in C$ and $w = xyyz$. Thus, $w \in \Sigma^*\mathrm{Dup}(C)\Sigma^*$. Conversely, let $v \in \Sigma^*\mathrm{Dup}(C)\Sigma^*$. Then v is of the form $xyyz$ such that $x, z \in \Sigma^*$ and $yy \in \mathrm{Dup}(C)$ (i.e., $y \in C$). The duplication of y in $xyz \in \Sigma^*$ results in $xyyz = v$, and hence $v \in (\Sigma^*)^{\heartsuit(C)}$. $\square$

The following corollary derives from Lemma 8.31 and Proposition 8.30. In fact, this corollary asserts the claim in the case when $L = \Sigma^*$.

Corollary 8.32. *Let $C \subseteq \Sigma^*$. Then $(\Sigma^*)^{\heartsuit(C)}$ is regular if and only if there exists a finite subset $C' \subseteq C$ such that $(\Sigma^*)^{\heartsuit(C')} = (\Sigma^*)^{\heartsuit(C)}$.*

The last case we consider is that of marked duplication, where given a word w in $L^{\heartsuit(C)}$, we can deduce or at least guess the factor whose duplication generates w from a word in L, according to some mark of a control set C. Here we consider a mark which shows the beginning and end of a word in C, that is, $C \subseteq \#(\Sigma\setminus\{\#\})^*\#$ for some character $\#$. For a strongly-marked duplication, where $\# \notin \Sigma$ and $L \subseteq \Sigma^*\#\Sigma^*\#\Sigma^*$, we can easily show that the existence of a finite control set provided $L^{\heartsuit(C)}$ is regular, using the pumping lemma for the regular language. Hence we consider the case when the mark itself is a character in Σ, say $\# = a$ for some $a \in \Sigma$.

It turned out that we could employ the proof of the claim in the case of the marked duplication for the more general case when C is a nonoverlapping and an infix code. A language L is called *non-overlapping* if $vx, yv \in L$ implies $x = y = \lambda$, and L is called *infix-code* if $L \cap (\Sigma^* L \Sigma^+ \cup \Sigma^+ L \Sigma^*) = \emptyset$. That is, any elements of the language which is non-overlapping and an infix-code do not overlap each other. In the following, we prove the claim for this case.

We introduce several notions and notations used in the proof. For a word $w \in L^{\heartsuit(C)}$, we call a tuple (x, y, z) a *dup-factorization of w with respect to L and C* if $w = xyyz$, $xyz \in L$, and $y \in C$. When L and C are clear from the context, we simply say that (x, y, z) is a dup-factorization of w. Let $\delta(w)$ be the number of dup-factorizations of w with respect to L and C. For $y \in C$, if there are $x, z \in \Sigma^*$ such that (x, y, z) is a dup-factorization of w, then we call y a *dup-factor* of w. Let $F_d(w)$ be the set of all dup-factors of w. Note that $|F_d(w)| \leq \delta(w)$ but the inequality may be strict.

Proposition 8.33. *Let L be a regular language and C be a control set which is non-overlapping and an infix-code. Then the regularity of $L^{\heartsuit(C)}$ implies the existence of a finite control set C' such that $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.*

Proof. Let $\equiv_L$ and $\equiv_{\heartsuit}$ be the syntactic congruences of L and $L^{\heartsuit(C)}$, respectively, and we define $\equiv \, = \, \equiv_L \cap \equiv_\heartsuit$. Since both L and $L^{\heartsuit(C)}$ are regular, $C/\equiv$ is finite. Let $\Gamma_2 = \{[c] \in C/\equiv \text{ s.t. } |[c]| \leq 2\}$. Using induction on the number of dup-factorizations, we prove that (i) $\Gamma_2 \neq \emptyset$, and (ii) any word in $L^{\heartsuit(C)}$ has a dup-factor which is in an equivalence class in Γ_2.

Firstly, we consider a word w in $L^{\heartsuit(C)}$ which has the smallest number of dup-factorizations among the elements of $L^{\heartsuit(C)}$. Suppose that no dup-factor of w is in equivalence classes in Γ_2. Let (x, y, z) be a dup-factorization of w. Then there exists $y' \in C$ such that $y' \equiv y$, $y' \neq y$, and $y' \notin \mathrm{Suff}(x)$. Let $w' = xy'yz$. This is in $L^{\heartsuit(C)}$, and hence w' must have a dup-factorization, say (α, β, γ) for some $\alpha, \beta, \gamma \in \Sigma^*$. Due to the non-overlapping and infix-code properties of C, β^2 is an infix of either x or yz. Here we assume that it is in x, and let $x = \alpha\beta^2 v$, $\gamma = vy'yz$ for some $v \in \Sigma^*$. Then

$$\begin{aligned} w' = \alpha\beta^2\gamma \in L^{\heartsuit(C)} \quad &\Rightarrow \quad \alpha\beta v y' y z \in L \\ &\Rightarrow \quad \alpha\beta v y y z \in L \\ &\Rightarrow \quad \alpha\beta^2 v y y z = w \in L^{\heartsuit(C)}. \end{aligned}$$

Thus, $(\alpha, \beta, vyyz)$ is a dup-factorization of w. Generally speaking, for a dup-factorization (α, β, γ) of w', w has a corresponding dup-factorization (α', β, γ) if y' is an infix of α, or (α, β, γ') otherwise (i.e., y' is an infix of γ). Indeed, this means that $\delta(w') < \delta(w)$ and $F_d(w') \subseteq F_d(w)$. The first consequence is a contradiction

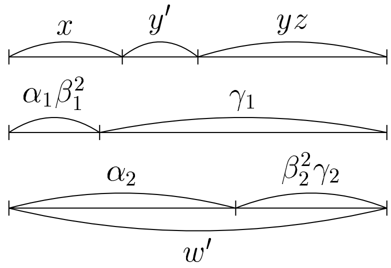

Figure 8.1: The comparison between two dup-factorizations, $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$, of w'.

while the second one is of importance in the induction step. The second is clear from the above discussion. In order to show the first, it is enough to prove that there do not exist two distinct dup-factorizations of w' which correspond to the same dup-factorization of w, and there exists no dup-factorization of w' which corresponds to (x, y, z).

Let $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ be two distinct dup-factorizations of w', and consider dup-factorizations of w which correspond to them respectively (either $(\alpha_i', \beta_i, \gamma_i)$ or $(\alpha_i, \beta_i, \gamma_i')$ for each $i = 1, 2$). Firstly we prove that $(\alpha_1, \beta_1, \gamma_1') \neq (\alpha_2, \beta_2, \gamma_2')$. Suppose not, then since $w' = \alpha_1\beta_1^2\gamma_1 = \alpha_2\beta_2^2\gamma_2$, we have $\gamma_1 = \gamma_2$, a contradiction. Next we compare $(\alpha_1, \beta_1, \gamma_1')$ and $(\alpha_2', \beta_2, \gamma_2)$ (see Fig. 8.1). Their construction shown above implies that γ_1 and α_2 must contain y' as their infix. Hence $|\alpha_1\beta_1^2| + |y'| \leq |\alpha_2|$. Since α_2' is generated by replacing y' in α_2 with y and $\beta \neq \lambda$, we have $|\alpha_1| < |\alpha_2|$. Thus, $(\alpha_1, \beta_1, \gamma_1') \neq (\alpha_2', \beta_2, \gamma_2)$. Using the same way, we can easily check that $(\alpha_i', \beta_i, \gamma_i), (\alpha_i, \beta_i, \gamma_i') \neq (x, y, z)$.

Now we assume that for all words in $L^{\heartsuit(C)}$ which have at most n dup-factorizations have a dup-factor which is in the equivalence class in Γ_2. Suppose that there were $v \in L^{\heartsuit(C)}$ with $n+1$ dup-factorizations and without any dup-factor which is in the equivalence class of size at most 2. Then we can construct a word v' as above which satisfies $\delta(v') \leq n$ and $F_d(v') \subseteq F_d(v)$, which contradict with the induction assumption. $\square$

Corollary 8.34. *Let L be a regular language and C be a control set. If there exists a finite set $C_1 \subset C$ such that $C \setminus C_1$ is non-overlapping and an infix-code, then the regularity of $L^{\heartsuit(C)}$ implies the existence of a finite control set C' such that $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.*

Proof. Note that $L^{\heartsuit(C)} = L^{\heartsuit(C_1)} \cup L^{\heartsuit(C \setminus C_1)}$. Proposition 8.33 implies the existence of a finite control set C_2 such that $L^{\heartsuit(C \setminus C_1)} = L^{\heartsuit(C_2)}$. Then by letting $C' = C_1 \cup C_2$, which is finite, we have $L^{\heartsuit(C)} = L^{\heartsuit(C')}$. $\square$

Indeed, we can prove that $\Gamma_1 = \{[c] \in C/\equiv \text{ s.t. } |[c]| = 1\}$ is enough to generate $L^{\heartsuit(C)}$, that is, for a finite control set $C' = \{c \mid [c] \in \Gamma_1\}$, $L^{\heartsuit(C)} = L^{\heartsuit(C')}$.

Proposition 8.35. *Let L be a regular language and $C \subseteq \Sigma^*$ be a nonoverlapping and an infix code. If $L^{\heartsuit(C)}$ is regular, then $L^{\heartsuit(C)} = L^{\heartsuit(C')}$, where $C' = \{c \mid [c] \in \Gamma_1\}$.*

Proof. All we have to prove is that for $w \in L^{\heartsuit(C)}$, unless w has a dup-factor which is in C', there exists $w' \in L^{\heartsuit(C)}$ such that $\delta(w') < \delta(w)$ and $F_d(w') \subseteq F_d(w)$.

Let (x, y, z) be a dup-factorization of w, and let $y' \in C$ such that $y \neq y'$ but $y \equiv y'$. Then let $w_0 = xy'yz$, which is in $L^{\heartsuit(C)}$. The proof of Proposition 8.33 implies that if either (1) $y' \notin \mathrm{Suff}(x)$ or (2) $x = x_1 y'$ for some $x_1 \in \Sigma^*$ but (x_1, y', yz) is not a dup-factorization of w_0, then $\delta(w_0) < \delta(w)$. Even otherwise ($w_0 = x_1 y'y'yz$), $\delta(w_0) \leq \delta(w)$. If this holds with equality, consider $w_1 = x_1 yy'yz \in L^{\heartsuit(C)}$. If either (1) $y \notin \mathrm{Suff}(x_1)$ or (2) $x_1 = x_2 y$ for some $x_2 \in \Sigma^*$ but $(x_2, y, y'yz)$ is not a dup-factorization of w_1, then $\delta(w_1) < \delta(w_0) = \delta(w)$. Otherwise, let $w_2 = x_2 y'yy'yz$. Note that x_k is getting strictly shorter. Hence repeating this process, we eventually reach an integer $i \geq 0$ such that either (1) or (2) holds for w_i. We can check that $\delta(w_i) < \delta(w_{i-1}) \leq \cdots \leq \delta(w_0) \leq \delta(w)$ and $F_d(w_i) \subseteq F_d(w)$ as follows: Let $w_i = x_i (y'y)^{i/2+1} z \in L^{\heartsuit(C)}$ (for even i; the odd case is essentially same and hence omitted). Let $w_i = \alpha\beta^2\gamma$, where (α, β, γ) is a dup-factorization of w_i. Since either (1) or (2) holds, β^2 is an infix of x_i or that of yz. Assume the former and let $x_i = \alpha\beta^2\gamma'$ and $\gamma = \gamma'(y'y)^{i/2+1}z$. Then $\alpha\beta\gamma'(y'y)^{i/2+1}z \in L$. Using $y \equiv y'$, we can say that $\alpha\beta\gamma'(yy')^{i/2}yyz \in L$, and hence $\alpha\beta^2\gamma'(yy')^{i/2}yyz \in L^{\heartsuit(C)}$. The lefthand side is $x_i(yy')^{i/2}yyz = x_{i-1}y'(yy')^{i/2-1}yyz = \cdots = xyyz = w$. $\square$

Consequently, we can say that if we let $m = |C/\!\equiv\!|$, then the size of finite control set C' is at most $m - 1$ because at least one equivalence class in $C/\equiv$ must have infinite cardinality.

8.7 Duplication and Primitivity

Recall that a word $w \in \Sigma^*$ is primitive if there exists no $u \in \Sigma^*$ such that $w = u^k$ for some $k \geq 2$. We denote by Q the set of all primitive words over the alphabet Σ. There is evidently a connection between duplication, repeat-deletion, and primitive words, but the nature of this relationship is unclear. The following section elucidates some of the properties of this relationship.

Proposition 8.36 (see, for instance, [18]). *Let $u, v \in \Sigma^+$ such that uv is primitive. Then both $u(uv)^n$ and $v(uv)^n$ are primitive for any $n \geq 2$.*

Proposition 8.37. *Let $w \in \Sigma^*$ be a non-primitive word. If we duplicate an infix of w which is strictly shorter than the primitive root of w, then the resulting word is primitive.*

Proof. Let $w = f^n$ for $f \in Q$ and $n \geq 2$. We also denote $w = xyz$ for $x, y, z \in \Sigma^*$, where y is the infix we duplicate so that the resulting word is $xyyz$. Since $w = f^n = xyz$, there exist $f_s \in \mathrm{Suff}(f)$ and $f'_p \in \mathrm{Pref}(f)$ satisfying $y = f_s f'_p$. Then yzx, a conjugate of xyz, is written as $yzx = (f_s f_p)^n$, where $f_p \in \mathrm{Pref}(f)$ satisfying $f = f_p f_s$. Let $g = f_s f_p$. Clearly $g \in Q$. Now we prove that $yyzx$ is primitive, and hence $xyyz$ is also primitive.

We have $yyzx = f_s f'_p yzx = f_s f'_p g^n$. Since $|y| < |f|$, there exists a word $v \in \Sigma^+$ such that $f_p = f'_p v$. Then $yyzx = y(yv)^n$ and Proposition 8.36 implies that $yyzx$ is primitive. $\square$

Proposition 8.38. *Let $x, y, z \in \Sigma^*$. If xyz is primitive and $xyyz$ is not primitive, then xz is primitive.*

Proof. Let f be the primitive root of y, i.e., $y = f^m$ for some $m \geq 1$. Since $xyyz \notin Q$, its conjugate $zxyy$ is also not primitive. Suppose zx were not primitive, i.e., $zx = g^n$ for some $n \geq 2$ and $g \in Q$. If $g \neq f$, then $zxyy = g^n f^{2m}$. Lemma 8.29 implies that $zxyy \in Q$, a contradiction. If $g = f$, then $y = g^m$ and hence $zxy = g^{n+m} \notin Q$. Thus, $xyz \notin Q$, a contradiction. As a result, $zx \in Q$, that is, $xz \in Q$. □

8.8 Discussion

In this paper, we studied duplication and repeat-deletion, two formal language theoretic models of insertion and deletion errors occurring during DNA replication. Specifically, we obtained the closure properties of the families of languages in the Chomsky hierarchy under these operations, the language equations of the form $X^{\heartsuit} = L$ and $X^{\spadesuit} = L$ for a given language L, and the operation of controlled duplication. In addition, we made steps towards finding a necessary and sufficient condition for a controlled duplication of a regular language to be regular.

Two problems for further investigation are: the problem of how to decide for a given language L whether the language equation $X^{\heartsuit} = L$ has a solution, and the problem of finding a necessary condition for the controlled duplication of a regular language to be regular, in the general case.

Acknowledgements

We wish to express our gratitude to Dr. Zoltán Ésik for the concise proof of Proposition 8.4. We would also like to thank Dr. Helmut Jürgensen for our discussion on the claim and Dr. Kathleen Hill for extended discussions on the biological motivation for duplication and repeat-deletion.

Bibliography

[1] M. Bichara, J. Wagner, and I. B. Lambert. Mechanisms of tandem repeat instability in bacteria. *Mutation Research*, 598(1-2):144–163, 2006.

[2] J. Dassow, V. Mitrana, and Gh. Păun. On the regularity of duplication closure. *Bulletin of the EATCS*, 69:133–136, 1999.

[3] J. Dassow, V. Mitrana, and A. Salomaa. Operations and language generating devices suggested by the genome evolution. *Theoretical Computer Science*, 270:701–738, 2002.

[4] M. Garcia-Diaz and T. A. Kunkel. Mechanism of a genetic glissando: Structural biology of indel mutations. *Trends in Biochemical Sciences*, 31(4):206–214, 2006.

[5] Z. Gu, L. M. Steinmetz, X. Gu, G. Scharfe, R. W. Davis, and W-H. Li. Role of duplicate genes in genetic robustness against null mutations. *Nature*, 421:63–66, 2003.

[6] M. A. Harrison. *Introduction to Formal Language Theory.* Addison-Wesley, 1978.

[7] M. Ito. *Algebraic Theory of Automata and Languages.* World Scientific, 2004.

[8] M. Ito, P. Leupold, and K. S-Tsuji. Closure of language classes under bounded duplication. In O. H. Ibarra and Z. Dang, editors, *DLT 2006*, volume 4036 of *Lecture Notes in Computer Science*, pages 238–247. Springer, 2006.

[9] Masami Ito, Lila Kari, Zachary Kincaid, and Shinnosuke Seki. Duplication in DNA sequences. In M. Ito and M. Toyama, editors, *DLT 2008*, volume 5257 of *Lecture Notes in Computer Science*, pages 419–430, 2008.

[10] P. Leupold. *Languages Generated by Iterated Idempotencies and the Special Case of Duplication.* PhD thesis, Universitat Rovira i Virgili, Facultat de Lletres, Department de Filologies Romàniques, Tarragona, Spain, 2006.

[11] P. Leupold. Duplication roots. In T. Harju, J. Karhumäki, and A. Lepistö, editors, *DLT 2007*, volume 4588 of *Lecture Notes in Computer Science*, pages 290–299. Springer, 2007.

[12] P. Leupold, C. Martin-Vide, and V. Mitrana. Uniformly bounded duplication languages. *Discrete Applied Mathematics*, 146(3):301–310, 2005.

[13] P. Leupold, V. Mitrana, and J. Sempere. Formal languages arising from gene repeated duplication. In *Aspects of Molecular Computing*, volume 2950 of *Lecture Notes in Computer Science*, pages 297–308. Springer, 2004.

[14] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and its Applications.* Addison-Wesley, 1983.

[15] R. C. Lyndon and M. P. Schützenberger. The equation $a^m = b^n c^p$ in a free group. *Michigan Mathematical Journal*, 9:289–298, 1962.

[16] C. Martin-Vide and Gh. Păun. Duplication grammars. *Acta Cybernetica*, 14:151–164, 1999.

[17] V. Mitrana and G. Rozenberg. Some properties of duplication grammars. *Acta Cybernetica*, 14:165–177, 1999.

[18] C. M. Reis and H. J. Shyr. Some properties of disjunctive languages on a free monoid. *Information and Control*, 37:334–344, 1978.

[19] R. Ross and K. Winklmann. Repetitive strings are not context-free. *R.A.I.R.O. informatique théorieque / Theoretical Informatics*, 16(3):191–199, 1982.

[20] G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages*, volume 1. Springer-Verlag, Berlin Heidelberg, 1997.

[21] D. B. Searls. The computational linguistics of biological sequences. In L. Hunter, editor, *Artificial Intelligence and Molecular Biology*, pages 47–120. AAAI Press, 1993.

[22] S. S. Yu. *Languages and Codes.* Tsang Hai Book Company Co., Taichung, Taiwan, 2005.

Chapter 9

Schema for parallel insertion and deletion

The contents of this chapter are taken from "Schema for parallel insertion and deletion"[1], which will be present at the 14th International Conference on Developments in Language Theory (DLT 2010).

Summary: We propose a general framework for parallel insertion/deletion operations based on p-schemata. A p-schema is a set of tuples of words. When being used for parallel insertion of a language into a word, an element of a p-schema specifies how to split the given word into factors between which the insertion of the language will take place. Parallel deletion based on a p-schema is defined as an "inverse" operation of parallel insertion based on the p-schema. Several well-known language operations are particular cases of p-schema-based insertions or deletions: catenation, Kleene star, reverse catenation, sequential insertion, parallel insertion,

[1] A version of this chapter has been published.

insertion next to a given letter, contextual insertion, right and left quotient, sequential deletion, parallel deletion. Additional operations that can be defined using p-schemata include contextual parallel insertion, as well as parallel insertion (deletion) of exactly n words, at most n words, an arbitrary number of words. We also consider the decidability and undecidability of existence of solutions of language equations involving p-schema-based parallel insertion/deletion.

Schema for parallel insertion and deletion

Lila Kari and Shinnosuke Seki

Department of Computer Science, The University of Western Ontario, London, Ontario, N6A 5B7, Canada.

9.1 Introduction

Since Adleman's success [1] in solving the Directed Hamiltonian Path Problem purely by biological means, which threw new light on fundamental research on operations in formal language theory, various bio-operations have been intensively investigated. Examples include hairpin inversion [11], circular insertion/deletion [18], excisions of loop, hairpin, and double-loop [12], and contextual insertion/deletion [16], to name a few.

The fact that one can experimentally implement in the laboratory some variants of insertions and deletions into/from DNA sequences [7], and use these as the sole primitives for DNA computation, gives practical significance to the research on insertion and deletion. Contextual insertion and deletion are also of theoretical interest because they have been proved to be Turing-universal [16]. In this paper, we will parallelize contextual insertion and deletion. For words x and y, the (x, y)-contextual insertion of a language L into a word w [16] results in the language

$$\bigcup_{w_1, w_2 \text{ with } w = w_1 x y w_2} w_1 x L y w_2.$$

In other words, one considers all the possibilities of cutting w into two segments, such that the first segment ends with x and the second segment begins with y, and for each such possibility L is inserted between these segments. This operation suggests that for any positive integer n, an n-tuple $(w_1, w_2, \ldots, w_n)$ of words may be used to control the parallel insertion of $n-1$ instances of L into $w = w_1 w_2 \cdots w_n$ to generate the language $w_1 L w_2 L \cdots L w_{n-1} L w_n$. A set of such tuples is called a *parallel operation schema* or *p-schema* for short, and we call the parallel insertion thus determined *parallel insertion based on the p-schema.* A p-schema can be used to control not only parallel insertion but parallel deletion as well. Parallel deletion of L from a word w based on a given n-tuple $(u_1, u_2, \ldots, u_n)$ deletes $n-1$ non-overlapping elements of L from w so as to leave this n-tuple, and concatenates them to generate the word $u = u_1 u_2 \cdots u_n$. As we shall see in Section 9.3, various well-known sequential as well as parallel operations (catenation, Kleene star, reverse catenation, sequential insertion, parallel insertion, insertion next to a given letter, contextual insertion, right and left quotient, sequential deletion, parallel deletion) are special instances of parallel operations based on p-schemata. Additional operations that can be defined using p-schemata are contextual parallel insertion, as well as parallel insertion (deletion) of *exactly n words, at most n words, an arbitrary number of words.*

Besides being proper generalizations of existing language operations, parallel operations based on p-schemata lead to some interesting results when studied in the context of language equations. Equations of the form $X_1 \diamond X_2 = X_3$ have been intensely studied in the literature, where $\diamond$ is a binary operation on languages, and some of X_1, X_2, X_3 are fixed languages, while the others are unknowns (see, e.g., [2, 4, 5, 6, 8, 10, 14, 15, 16]). In this paper, we focus on such language equations with $\diamond$ being p-schema-based insertion or deletion. Since these two operations are parameterized by p-schemata, we can also consider the problem of deciding whether $L_1 \diamond_X L_2 = L_3$ has a solution, i.e., whether there exists a p-schema F such that parallelly inserting L_2 into (deleting from) L_1 based on F results in L_3.

In general, procedures do not exist for solving such equations when they involve a context-free language. Therefore, we focus on solving equations of the form (1) $X \leftarrowtail_F R_2 = R_3$, (2) $R_1 \leftarrowtail_X R_2 = R_3$, (3) $R_1 \leftarrowtail_F X = R_3$, and their p-schema-based deletion variants, where all of R_1, R_2, R_3, F are regular[2]. Among these equations, the equations of the first or second form can be solved using the technique of [15]. The application of this technique presumes the property that the union of all the solutions to the given equation is the unique maximal solution. As we shall see, the third-type equations do not have this property, that is, they may have multiple maximal solutions. Algorithms to solve these equations are one of the main contributions of this paper. Our algorithms work not only as a procedure to decide the existence of solutions, but as a procedure to enumerate all maximal solutions (Theorems 9.14 and 9.17). Moreover, combining these algorithms with the algorithms

[2] by catenating words in a tuple of words via a special symbol #, we can naturally associate a set of tuples of words with a language, and as such we can establish a Chomsky-hierarchy for the sets of tuples of words.

to solve the equations of the first or second form (outlined in Section 9.5) enables us to solve two-variables equations of the form $X \leftarrowtail_F Y = R_3$ (Theorem 9.20), $R_1 \leftarrowtail_X Y = R_3$ (Theorem 9.21), and $R_1 \rightarrowtail_X Y = R_3$ (Theorem 9.22). The proposed algorithms can be modified to also solve inequality (set inclusion) variants of the above-mentioned equations with maximality condition on variables.

9.2 Preliminaries

By Σ we denote a finite alphabet, and the set of words over Σ is denoted by Σ^* which includes the *empty word* λ. For a given word w, its length is denoted by $|w|$, and its reversal is denoted by w^R. For an integer $n \geq 0$, Σ^n, $\Sigma^{\leq n}$, and $\Sigma^{\geq n}$ denote the sets of all words of length *exactly* n, *at most* n, and *at least* n, respectively. A word u is called a *factor* (*prefix*, *suffix*) of a word w if $w = xuy$ (resp. $w = uy$, $w = xu$) for some words x, y. Let us denote the set of all prefixes (suffixes) of w by $\mathrm{Pref}(w)$ (resp. $\mathrm{Suff}(w)$). For a language $L \subseteq \Sigma^*$, $L^c = \Sigma^* \setminus L$.

Regular languages are specified by (non-deterministic) finite automata (NFA) $A = (Q, \Sigma, \delta, s, F)$, where Q is a finite set of states, $s \in Q$ is the start state, $F \subseteq Q$ is a set of final states, and δ is a map from $Q \times \Sigma$ to 2^Q. For notational convenience, we employ the notation NFA also to denote a language accepted by an NFA (we use this slight abuse of notation for other kinds of acceptors). The family of languages accepted by NFAs is denoted by REG. An NFA is said to be *deterministic* if δ is a function. The deterministic property of a machine is stated explicitly by using the capital letter D. A language is said to be *effectively regular* if there exists an algorithm to construct an NFA which accepts this language.

A characterization of languages can be given in terms of *syntactic semigroups*. For a language $L \subseteq \Sigma^*$, there exists a maximal congruence $\equiv_L$ which saturates L (i.e., L is a union of equivalence classes). This is called the *syntactic congruence* of L, which is formally defined as follows: for $u, v \in \Sigma^*$,

$$u \equiv_L v \iff \text{for any } x, y \in \Sigma^*,\ xuy \in L \text{ if and only if } xvy \in L.$$

For a word $w \in \Sigma^*$, a set $[w]_{\equiv_L} = \{u \in \Sigma^* \mid w \equiv_L u\}$ is called an *equivalence class* with w as its representative. The number of equivalence classes is called the *index* of $\equiv_L$.

Theorem 9.1 ([19]). *Let $L \subseteq \Sigma^*$ be a language. The index of $\equiv_L$ is finite if and only if L is regular.*

For technical reasons, we define a function called *saturator with respect to a language* L_1. Let σ_{L_1} be a function from a word w into the equivalence class $[w]_{\equiv_{L_1}}$. The saturator w.r.t. L_1 is its extension defined as $\sigma_{L_1}(L) = \bigcup_{w \in L} [w]_{\equiv_{L_1}}$.

We can choose an arbitrary word in $[w]_{\equiv_L}$ as a representative of this class. By taking a representative from every class, we can construct a subset of Σ^* called a *complete system of representatives* of $\Sigma^*/\equiv_L$. In particular, for a regular language

R, there exists a complete system of representatives which is computable. Let $A = (Q, \Sigma, \delta, s, F)$ be the (unique) minimal-DFA for R. Then $u \equiv_R v$ if and only if $\delta(q, u) = \delta(q, v)$ for any $q \in Q$. Hence, the index of $\equiv_L$ is at most $|Q|^{|Q|}$.

Theorem 9.2. *Let R be a regular language and $A = (Q, \Sigma, \delta, s, F)$ be the min-DFA for R. Each equivalence class in $\Sigma^*/\equiv_R$ is regular, and contains a word of length at most $|Q|^{|Q|}$.*

Corollary 9.3. *For a regular language R, there exists a computable complete system of representatives of $\Sigma^*/\equiv_L$.*

9.3 Parallel insertion and deletion schema

Imagine that we will insert a language L into a word u in parallel. Let $\prod_{i=1}^{n} \Sigma^*$ be the Cartesian product of Σ^* with itself n times; that is to say, the set of all n-tuples of words. Let $\mathfrak{F} = \bigcup_{n \geq 1} \underbrace{\Sigma^* \times \Sigma^* \times \cdots \times \Sigma^*}_{n \text{ times}}$. A subset F of $\mathfrak{F}$ can be used to control the parallel insertion of a language L in a sense that if $(u_1, u_2, \ldots, u_n) \in F$, then the word $u = u_1 u_2 \cdots u_n$ is split in the manner dictated by the n-tuple in F, and L is inserted between u_i and u_{i+1} for all $1 \leq i < n$ to generate the language $u_1 L u_2 L \cdots u_{n-1} L u_n$. The set can be also used to control a parallel deletion. For this intended end-usage, we call a subset of $\mathfrak{F}$ a *parallel schema*, or shortly *p-schema*, *over* Σ.

As abstracted above, a p-schema F enables us to define the *(parallel) insertion*

$\leftarrowtail_F$ as: for a word $u \in \Sigma^*$ and a language $L \subseteq \Sigma^*$,

$$u \leftarrowtail_F L = \bigcup_{n \geq 1, u = u_1 \cdots u_n, (u_1, \ldots, u_n) \in F} u_1 L u_2 L \cdots u_{n-1} L u_n.$$

Note that an n-tuple in F parallel-inserts $n-1$ words from L into u. Similarly, we define the *(parallel) deletion $\rightarrowtail_G$ based on a p-schema G* as: for a word $w \in \Sigma^*$ and a language $L \subseteq \Sigma^*$,

$$\begin{aligned} w \rightarrowtail_G L &= \{u_1 \cdots u_n \mid n \geq 1, x_1, \ldots, x_{n-1} \in L, \\ &\qquad (u_1, \ldots, u_n) \in G, w = u_1 x_1 u_2 x_2 \cdots u_{n-1} x_{n-1} u_n\}, \end{aligned}$$

These operations are extended to languages in a conventional manner: for a language L_1, $L_1 \leftarrowtail_F L = \bigcup_{u \in L} u \leftarrowtail_F L$ and $L_1 \rightarrowtail_G L = \bigcup_{w \in L} w \rightarrowtail_G L$.

Many of the well-known operations are particular cases of p-schema-based operations. We list instances of p-schema-based insertion:

catenation	$F_{\text{cat}} = \Sigma^* \times \lambda$,
reverse catenation	$F_{\text{rcat}} = \lambda \times \Sigma^*$,
(sequential) insertion	$F_{\text{sins}} = \Sigma^* \times \Sigma^*$,
parallel insertion	$F_{\text{pins}} = \bigcup_{n \geq 0} (\lambda \times \prod_{i=1}^{n} \Sigma \times \lambda)$.

Deletions based on F_{cat}, F_{rcat}, F_{sins}, and F_{pins} correspond to right and left quotient, (sequential) deletion, and parallel deletion, respectively.

Parallel insertion (deletion) of *exactly n words*, *at most n words*, or *arbitrary number of words* are important instances of insertion (deletion) based on:

$$F_{\mathrm{pins}(n)} = \prod_{i=1}^{n+1} \Sigma^*, \qquad F_{\mathrm{pins}(\leq n)} = \bigcup_{i=0}^{n} F_{\mathrm{pins}(i)}, \qquad F_* = \bigcup_{i=0}^{\infty} F_{\mathrm{pins}(i)},$$

respectively. Using for instance F_*, one can implement Kleene-star, the most well-studied unary operation in formal language theory, as $L^* = \lambda \longleftarrow_{F_*} L$.

The p-schemata introduced so far are "syntactic" in a sense, while many of semantic (letter-sensitive) operations are known. For a letter $b \in \Sigma$, *parallel insertion next to b* [14] is the insertion based on $F_{\mathrm{pinsb}} = \{(u_1, u_2, \ldots, u_n) \mid n \geq 1, u_1, \ldots, u_n \in (\Sigma \setminus \{b\})^* b\}$. For a context $C \subseteq \Sigma^* \times \Sigma^*$, *$C$-contextual (sequential) insertion* [16] is the insertion based on $F_{\mathrm{scins}}(C) = \bigcup_{(x,y) \in C} \Sigma^* x \times y \Sigma^*$. This operation is naturally parallelized as *C-contextual parallel insertion* with the p-schema $F_{\mathrm{pcins}}(C) = \{(u_1, \ldots, u_n) \mid n \geq 1, \forall 1 \leq i < n, \big(\mathrm{Suff}(u_i) \times \mathrm{Pref}(u_{i+1})\big) \cap C \neq \emptyset\}$.

It may be worth noting that the descriptional powers of our framework and of I-shuffle proposed by Domaratzki, Rozenberg, and Salomaa [9] (a generalization of semantic shuffle proposed by Domaratzki [8]) are incomparable. Indeed, only I-shuffle can specify contexts not only on the left operand but also on the right operand, while p-schema-based operations can insert/delete multiple copies of right operand. Thus, insertion/deletion based on a p-schema which contains 2-tuples and/or 1-tuples is a special instance of I-shuffle.

9.4 Hierarchy of p-schemata and closure properties

In this section, we investigate closure properties of abstract families of acceptors augmented with reversal-bounded counters under the p-schema-based operations. Such an acceptor was proposed by Ibarra [13] as the *counter machine*. For $k \geq 0$, let NCM(k) be the class of NFAs augmented with k reversal-bounded counters, and NCM be the union of such classes over all k's. By augmenting an NCM with an unrestricted pushdown stack, we obtain a *non-deterministic pushdown counter machine* (NPCM). For $k \geq 0$, let NPCM(k) be an NPCM with k reversal-bounded counters. DCM(k), DPCM(k), DCM, and DPCM are the deterministic analogs of NCM(k), NPCM(k), NCM, and NPCM. A desirable property specific to these deterministic classes is proved by Ibarra [13] as follows:

Theorem 9.4. *For $L_1 \in \mathrm{DCM}$ and $L_2 \in \mathrm{DPCM}$, it is decidable whether $L_1 = L_2$.*

It is natural to encode a tuple $(u_1, u_2, \ldots, u_n)$ as a word $u_1\#u_2\#\cdots\#u_n$ using a special symbol #. Denoting this (one-to-one) encoding by ψ, we can encode a p-schema F as $\psi(F) = \{\psi(f) \mid f \in F\}$. Furthermore, we say that a p-schema F is *in a language class* $\mathcal{L}$ if $\psi(F) \in \mathcal{L}$. For instance, F is *regular* if $\psi(F)$ is a regular language over $\Sigma \cup \{\#\}$.

First of all, we prove that REG is closed under insertion/deletion based on a regular p-schema as Corollary 9.6. Actually, the following stronger result holds, though the rest of this paper does not require more than Corollary 9.6.

Proposition 9.5. *Let $L_1 \in \mathrm{NCM}(k_1)$, $L_2 \in \mathrm{REG}$, and F be a p-schema in $\mathrm{NCM}(k_\psi)$. Then both $L_1 \leftarrowtail_F L_2$ and $L_1 \rightarrowtail_F L_2$ are in $\mathrm{NCM}(k_1 + k_\psi)$.*

Proof. We show only a construction of an NCM M for $L_1 \leftarrowtail_F L_2$, and omit the construction of an NCM for $L_1 \rightarrowtail_F L_2$.

Let M_2 be a finite automaton for L_2, and M_1, M_ψ be respective NCMs with k_1, k_ψ counters for $L_1, \psi(F)$. The NCM M expects its input to be of the form $u_1 x_1 u_2 x_2 \cdots x_{n-1} x_n$ for some integer $n \geq 1$, $u_1 u_2 \cdots u_n \in L_1$, $x_1, x_2, \ldots, x_{n-1} \in L_2$, and $u_1 \# u_2 \# \cdots \# u_n \in \psi(F)$. M simulates M_1 and M_ψ simultaneously. Guessing non-deterministically that the prefix $u_1 x_1 \cdots x_{i-1} u_i$ has been read, M pauses the simulation of both M_1 and M_ψ and instead activates the simulation of M_2 on x_i after having M_ψ make a #-transition. When M_2 is in one of its accepting states, M non-deterministically resumes the simulation of M_1 and M_ψ on the suffix $u_{i+1} x_{i+1} \cdots x_{n-1} u_n$ of the input. The simulation of M_2 is initialized every time it is invoked. $\square$

Corollary 9.6. *For regular languages R_1, R_2 and a regular p-schema F, both $R_1 \leftarrowtail_F R_2$ and $R_1 \rightarrowtail_F R_2$ are effectively regular.*

We can prove an analogous result of Proposition 9.5 for NPCM. By enlarging some of the respective language classes which L_1 and F belong to up to NPCM and a class which L_2 belongs to up to CFL, we can ask whether or not $L_1 \leftarrowtail_F L_2$ or $L_1 \rightarrowtail_F L_2$ are in NPCM. In the following we only address some non-closure properties of DPCM with implications to language equation solvability in the next section.

Let us define the *balanced language* L_b over $\Sigma = \{a, \$\}$ as follows:

$$
\begin{aligned}
L_b \;\; = \;\; & \{a^{i_1}\$a^{i_2}\$\cdots\$a^{i_k}\$a^{i_{k+1}}\$\cdots\$a^{i_n} \mid n \geq 2,\, i_1, \ldots, i_n \geq 0 \text{ and} \\
& \exists 1 \leq k < n \text{ such that } i_1 + i_2 + \cdots + i_k = i_{k+1} + \cdots + i_n\}.
\end{aligned}
$$

In other words, a word in L_b has a central marker \$ so that the number of a's to the left of this marker is equal to the number of a's to its right. For $L_1 = \{a^n\$a^n \mid n \geq 1\}$, we obtain $L_1 \leftarrowtail_{F_*} \$ = L_b$. Recall the definition of F_*; in this case it scatters an arbitrary number of \$'s into any word in L_1. We can generate L_b also by deletion. Let $L_1 = \bigcup_{n \geq 0}(a^n \sqcup\!\sqcup \$^*)\$\#(a^n \sqcup\!\sqcup \$^*)$ and $F = \{a, \$\}^* \times \{a, \$\}^*$, where $\sqcup\!\sqcup$ denotes shuffle operation. Then $L_b = L_1 \rightarrowtail_F \#$. These L_1's are DCM(1). L_b is clearly in NCM(1) because the non-determinism makes it possible for the reversal-bounded counter to guess when it should transit into its decrementing mode. In contrast, L_b is proved not to be DPCM (see, e.g., [3]). Consequently we have the following non-closure property.

Proposition 9.7. *There exist $L_1 \in \mathrm{DCM}(1)$, a regular p-schema F, and a singleton language L_2 such that $L_1 \leftarrowtail_F L_2 \notin \mathrm{DPCM}$.*

Proposition 9.8. *There exist $L_1 \in \mathrm{DCM}(1)$, a regular p-schema F, and a singleton language L_2 such that $L_1 \rightarrowtail_F L_2 \notin \mathrm{DPCM}$.*

By swapping the roles of L_1 and F in the above example, we can also obtain the following non-closure property.

Proposition 9.9. *There exist a regular language R_1, a singleton language L_2, and a $\mathrm{DCM}(1)$ p-schema F such that $R_1 \rightarrowtail_F L_2 \notin \mathrm{DPCM}$.*

9.5 Language equations with p-schemata-based operations

In this section, we consider language equations involving p-schema-based operations. The simplest equations to be studied are one-variable equations of the form $X \leftarrowtail_F L_2 = L_3$, $L_1 \leftarrowtail_X L_2 = L_3$, $L_1 \leftarrowtail_F X = L_3$, and their deletion variants. Such equations with special instances of p-schema-based operations (catenation, insertion, etc.) as well as incomparable operations (shuffle, etc.) have been intensively studied for the last decades [4, 5, 8, 10, 14, 15, 17]. These papers mainly dealt with language equations with the property that the union of all their solutions (if any) is also their solution (maximum solution). For instance, if $XL = R$ and $YL = R$, then

$(X \cup Y)L = R$. For such equations, we can employ a technique established in [15]; assuming a given equation has a solution, firstly construct the candidate of its maximum solution, and then substitute it into the equation to check whether it is actually a solution. Since $X \leftarrowtail_F L_2 = L_3$, $L_1 \leftarrowtail_X L_2 = L_3$, and their deletion variants have this property, this technique can solve these equations. We will now see how to construct the candidate for each.

In [5], Cui, Kari, and Seki defined the left-l-inverse relation between operations as: the operation $\bullet$ is *left-l-inverse* of the operation $\circ$ if for any words $u, w \in \Sigma^*$ and any language $L \subseteq \Sigma^*$, $w \in u \circ L \iff u \in w \bullet L$. This is a symmetric relation. By definition, insertion and deletion based on the same p-schema are left-l-inverse to each other. There they proved that for operations $\circ, \bullet$ which are left-l-inverse to each other, if $X \circ L_2 = L_3$ has a solution, then $(L_3^c \bullet L_2)^c$ is its maximum solution.

Theorem 9.10. *For regular languages R_2, R_3 and a regular p-schema F, the existence of a solution to both $X \leftarrowtail_F R_2 = R_3$ and $X \rightarrowtail_F R_2 = R_3$ is decidable.*

Proof. Both $(R_3^c \rightarrowtail_F R_2)^c \leftarrowtail_F R_2$ and $(R_3^c \leftarrowtail_F R_2)^c \rightarrowtail_F R_2$ are regular according to Corollary 9.6 and the fact that REG is closed under complement. Now it suffices to employ Theorem 9.4 for testing the equality. □

For $L_1 \leftarrowtail_X L_2 = L_3$, the candidate is $F_{\max} = \{f \in \mathfrak{F} \mid L_1 \leftarrowtail_f L_2 \subseteq L_3\}$. For $L_1 \rightarrowtail_X L_2 = L_3$, $F_{\max}$ should be rather $\{f \in \mathfrak{F} \mid L_1 \rightarrowtail_f L_2 \subseteq L_3\}$. When L_1, L_2, L_3 are all regular, we can construct an NFA for $\psi(\mathfrak{F} \setminus F_{\max})$, which is equal to $(\Sigma \cup \#)^* \setminus \psi(F_{\max})$. A similar problem was studied in [10], and our construction originates from theirs. As such, the proof of next result is omitted.

Theorem 9.11. *For regular languages R_1, R_2, R_3, the existence of a solution to both $R_1 \leftarrowtail_X R_2 = R_3$ and $R_1 \rightarrowtail_X R_2 = R_3$ is decidable.*

9.5.1 Solving $L_1 \leftarrowtail_F X = L_3$

In contrast, the equations $L_1 \leftarrowtail_F X = L_3$ and $L_1 \rightarrowtail_F X = L_3$ may not have a maximum solution. For example, let $L_1 = L_3 = \{a^{2n} \mid n \geq 1\}$, and $F = F_{\text{pins}(2)} \cup F_{\text{pins}(0)}$. Both $L_{\text{even}} = \{a^{2m} \mid m \geq 0\}$ and $L_{\text{odd}} = \{a^{2m+1} \mid m \geq 0\}$ are (maximal) solutions to $L_1 \leftarrowtail_F X = L_3$. On the other hand, $L_1 \leftarrowtail_F (L_{\text{even}} \cup L_{\text{odd}})$ can generate a^3, which is not in L_3. For deletion, let $F = \{(\lambda, aba), (\lambda, \lambda, \lambda), (aba, \lambda)\}$, and $L_1 = \{ababa\}$. Then $L_1 \rightarrowtail_F \{ab\} = L_1 \rightarrowtail_F \{ba\} = \{aba\}$, but $L_1 \rightarrowtail_F \{ab, ba\} = \{aba, a\}$. These examplify that we cannot apply the previously-mentioned approach to solving language equations with the second operand being unknown.

We propose an alternative approach based on an idea from Conway (Chapter 6 of [4]) to solve $f(\Sigma \cup \{X_1, X_2, \ldots\}) \subseteq R$, where f is a regular function over Σ and variables $X_1, X_2, \ldots$, and R is a regular language. The idea shall be briefly explained in terms of p-schema-based operations in order to step into more general cases than the case when all the involved languages are regular.

Lemma 9.12. *Let L, L_1 be languages. Then $(L_1 \leftarrowtail_F (L_2 \cup w)) \cap L \neq \emptyset$ if and only* *if* $(L_1 \leftarrow\!\!\prec_F (L_2 \cup [w]_{\equiv_L})) \cap L \neq \emptyset$ *for any word* w *and language* L_2.

By replacing L in this lemma with L_3^c, we can see that if $L_1 \leftarrow\!\!\prec_F (L_2 \cup w) \subseteq L_3$, then $L_1 \leftarrow\!\!\prec_F (L_2 \cup [w]_{\equiv_{L_3}}) \subseteq L_3$. Thus, it makes sense to introduce the notion of a syntactic solution. For a language L, we say that a solution to a one-variable language equation is *syntactic with respect to* L if it is a union of equivalence classes in $\Sigma^*/\equiv_L$.

Proposition 9.13. *For languages* L_1, L_3*, the equation* $L_1 \leftarrow\!\!\prec X = L_3$ *has a solution if and only if it has a syntactic solution with respect to* L_3*.*

Thus, in order to determine whether $L_1 \leftarrow\!\!\prec_F X = L_3$ has a solution, it suffices to test whether it has a syntactic solution. On condition that this test can be executed, this problem becomes decidable. If L_3 is regular, then the number of candidates of syntactic solution is finite (Theorem 9.1), and they are regular (Theorem 9.2). Let ß $= \{\sigma_{R_3}(L) \mid L \subseteq \Sigma^*\}$, the set of all candidates of syntactic solution. A pseudocode to solve $L_1 \leftarrow\!\!\prec_F X = R_3$ is given below:

Algorithm to solve $L_1 \leftarrow\!\!\prec_F X = R_3$

1. Order the elements of ß in some way (let us denote the i-th element of ß by ß$[i]$).
2. for each $1 \leq i \leq |$ß$|$, test whether $L_1 \leftarrow\!\!\prec_F$ ß$[i]$ is equal to R_3.

With the further condition that L_1 and F are chosen so that any language obtained by substituting a candidate into $L_1 \leftarrowtail_F X$ is comparable with R_3 for equality, this algorithm becomes executable. One such condition of significance is that both L_1 and F are regular. In this case, the algorithm, Theorem 9.4, and Corollary 9.6 lead us to the next theorem, which is stronger than decidability. It should be noted that maximal solutions are syntactic.

Theorem 9.14. *For regular languages R_1, R_3 and a regular p-schema F, the set of all syntactic solutions to $R_1 \leftarrowtail_F X = R_3$ is computable.*

The regularity of R_3 is necessary for the algorithm to work, whereas such condition is not imposed on L_1. If a condition on L_1, F under which $L_1 \leftarrowtail_F \text{ß}[i] \in \text{DPCM}$ for any $1 \leq i \leq |\text{ß}|$ were found, we could solve $L_1 \leftarrowtail_F X = R_3$ under it using Theorem 9.4. This is an unsettled question, but as suggested in Proposition 9.7, weakening the condition on L_1 slightly can make $L_1 \leftarrowtail_F X$ non-DPCM. It is probably more promising to broaden the class of F.

9.5.2 Solving $L_1 \rightarrowtail_F X = L_3$

Let us continue the investigation on the existence of right operand by changing the operation to p-schema-based deletion.

Lemma 9.15. *Let L_1 be a language. Then $L_1 \rightarrowtail_F (\{w\} \cup L_2) = L_1 \rightarrowtail_F ([w]_{\equiv_{L_1}} \cup L_2)$ for any word w, language L_2, and a p-schema F.*

Proof. Let $u \in L_1 \rightarrowtail_F ([w]_{\equiv_{L_1}} \cup L_2)$; that is, there exist $v \in L_1$, $n \geq 0$, $(u_1, u_2, \ldots, u_{n+1}) \in F$, and $x_1, \ldots, x_n \in [w]_{\equiv_{L_1}} \cup L_2$ such that $u = u_1 u_2 \cdots u_{n+1}$ and $v = u_1 x_1 u_2 x_2 \cdots u_n x_n u_{n+1}$. Now on v if $x_i \in [w]_{\equiv_{L_1}}$, then we replace x_i with w, and this process converts v into a word v'. Note that this replacement process guarantees that $v' \in L_1$ because the replaced factors are equal to w with respect to the syntactic congruence of L_1. Moreover, $u \in v' \rightarrowtail_F (\{w\} \cup L_2)$. Thus, $L_1 \rightarrowtail_F (\{w\} \cup L_2) \supseteq L_1 \rightarrowtail_F ([w]_{\equiv_{L_1}} \cup L_2)$. $\square$

This lemma provides us with two approaches to determine whether a given equation with p-schema-based deletion has a solution. The first approach is based on syntactic solutions. Given a language L_2, Lemma 9.15 implies that $L_1 \rightarrowtail_F L_2 = L_1 \rightarrowtail_F \sigma_{L_1}(L_2)$. Therefore, as in the case of insertion, the existence of a solution to $L_1 \rightarrowtail_F X = L_3$ is reduced to that of its syntactic solutions, but with respect to L_1 (not L_3). Moreover, maximal solutions are syntactic.

Proposition 9.16. *For languages L_1, L_3 and a p-schema F, the equation $L_1 \rightarrowtail_F X = L_3$ has a solution if and only if it has a syntactic solution with respect to L_1. Furthermore, its maximal solution (if any) is syntactic.*

With a straightforward modification, the algorithm presented in Sect. 9.5.1 can be used to output all syntactic solutions to $R_1 \rightarrowtail_F X = L_3$ with F being a regular p-schema. Thus, we have the following result, analogous to Theorem 9.14.

Theorem 9.17. *For a regular language R_1, $L_3 \in$ DPCM, and a regular p-schema F, the set of all syntactic solutions to $R_1 \rightarrowtail_F X = L_3$ is computable.*

Note that even if L_3 is DPCM, the equation above is solvable due to Corollary 9.6 and Theorem 9.4.

The existence of the second approach provided by Lemma 9.15 is due to the essential difference between Lemma 9.15 and its analog for insertion (Lemma 9.12). A word obtained by deleting some words in L_2 from a word in L_1 can be also obtained by deleting their representatives in a complete system of representatives with respect to L_1 from the word in L_2 based on the same schema; this is not true for insertion. Since its choice is arbitrary, we fix $\mathfrak{R}(L_1)$ to be the set of smallest words according to the lexicographical order in each equivalence class. We say that a solution to $L_1 \rightarrowtail_F X = L_3$ is *representative* if it is a subset of $\mathfrak{R}(L_1)$.

Proposition 9.18. *For languages L_1, L_3 and a p-schema F, the equation $L_1 \rightarrowtail_F X = L_3$ has a solution if and only if it has a representative solution.*

If L_1 is regular, then $\mathfrak{R}(L_1)$ is a finite computable set due to Theorem 9.1 and Corollary 9.3, and hence, our argument based on representative solution amounts to the second approach.

Theorem 9.19. *For a regular language R_1, $L_3 \in$ DPCM, and a regular p-schema F, the set of all representative solutions of $R_1 \rightarrowtail_F X = L_3$ is computable.*

With Theorem 9.1, Lemma 9.15 also leads us to a corollary about the number of distinct languages obtained by p-schema-based deletion from a regular language. Namely, given a regular language R_1 and a p-schema F, there exist at most a finite number of languages which can be represented in the form $R_1 \rightarrowtail_F L_2$ for some language L_2. This result is known for sequential deletion [14].

9.5.3 Solving two-variables language equations and inequalities

There is one thing which deserves explicit emphasis: the set of all candidates of syntactic solutions is solely determined by only one of L_3, L_1, and does not depend on the other or F at all. This property paves the way to solving two-variables language equations of the form $X \leftarrowtail_F Y = L_3$, $L_1 \leftarrowtail_X Y = L_3$, and $L_1 \rightarrowtail_X Y = L_3$. The first equation with $F = F_{\text{cat}}$ (catenation) has been investigated under the name of *decomposition of regular languages* and proved to be decidable [17, 20].

Let us assume that (L_1, L_2) is a solution of $X \leftarrowtail_F Y = L_3$. Then $\sigma_{L_3}(L_2)$ is a solution of $L_1 \leftarrowtail_F Y = L_3$, and hence, $(L_1, \sigma_{L_3}(L_2))$ is also a solution of $X \leftarrowtail_F Y = L_3$. This means that if the equation has a solution (pair of languages), then it also has a solution whose second element is a sum of equivalence classes in $\Sigma^*/\equiv_{L_3}$. Therefore, solving $X \leftarrowtail_F \text{ß}[i] = L_3$ for all $1 \leq i \leq |\text{ß}|$ using Theorem 9.10 amounts to solving the two-variables equation. For a regular language R_3 and a regular p-schema F, the above method works effectively to solve $X \leftarrowtail_F Y = R_3$.

Theorem 9.20. *It is decidable whether the equation $X \leftarrowtail_F Y = R_3$ has a solution or not if both R_3 and F are regular.*

Undertaking the same "two-staged" strategy but using Theorem 9.11 instead, we can solve the equations of second and third forms.

Theorem 9.21. *For regular languages R_1, R_3, it is decidable whether the equation $R_1 \leftarrowtail_X Y = R_3$ has a solution or not.*

Theorem 9.22. *For regular languages R_1, R_3, it is decidable whether the equation $R_1 \rightarrowtail_X Y = R_3$ has a solution.*

Unlike p-schema-based insertion, this strategy does not work to solve the equation of the form $X \rightarrowtail_F Y = L_3$. This is because in this case it is not L_3 but L_1 that determines the syntactic solutions of $L_1 \rightarrowtail_F Y = L_3$.

The usage of the proposed algorithm is not exclusive to solving language *equations*. By replacing the equality test in Step 2 with the following inclusion test "for each $1 \leq i \leq |\text{ß}|$, test whether $L_1 \leftarrowtail_F \text{ß}[i]$ is a subset of R_3", the proposed algorithm can answer the problem of finding maximal solutions to the language inequality $L_1 \leftarrowtail_F X \subseteq R_3$, and with the two-staged strategy, this further enables us to solve $X \leftarrowtail_F Y \subseteq R_3$ and $L_1 \leftarrowtail_X Y \subseteq R_3$. Now it should be trivial how to approach $R_1 \rightarrowtail_F X \subseteq L_3$ and $R_1 \rightarrowtail_X Y \subseteq L_3$.

9.5.4 Undecidability

We conclude this section and this paper by complementing the decidability results obtained so far with some undecidability results for one-variable equations. Usually, the existence of solutions to a language equation of this type is decidable if all known languages are regular, and undecidable if at least one of the known languages is context-free. The results of this section bring down, for several cases, the limit for undecidability of existence of solutions of such language equations from the class of context-free languages to NCM(1). The equation $L_1 \leftarrow\!\!\prec_F X = L_3$ is solvable in the case of L_1, F, L_3 being regular, i.e., NCM(0). Actually, we shall prove that once one of them becomes NCM(1), then this problem immediately turns into undecidable.

Proposition 9.23. *For languages L_1, L_3 and a p-schema F, if one of L_1, L_3, F is in* NCM(1) *and the others are regular, it is undecidable whether $L_1 \leftarrow\!\!\prec_F X = L_3$ has a solution or not.*

Proof. We employ the reduction of universe problem (whether a given NCM(1) is Σ^*) into these problems. The universe problem is known to be undecidable for the class NCM(1) [13]. Because of space limitations, we can consider here only the case when F is an NCM(1) p-schema.

Let $\natural, \$$ be special symbols not included in Σ. Based on a given $L \in$ NCM(1), we define a p-schema $F = \lambda \times \$L$, which is in NCM(1), too. Then for regular languages $\$\Sigma^*$ and $\natural\$\Sigma^*$, we claim that $\$\Sigma^* \leftarrow\!\!\prec_F X = \natural\$\Sigma^*$ has a solution $\iff$ $L = \Sigma^*$. Indeed, the left-hand side of the above equation is $X\$L$ so that its only one possible solution is $X = \natural$. Thus, the existence of the solution leads us immediately to that L is universe. $\square$ $\square$

For the equation $L_1 \rightarrowtail_F X = L_3$, the similar undecidability result holds.

Proposition 9.24. *For languages L_1, L_3 and a p-schema F, if one of L_1, L_3, F is in* NCM(1) *and the others are regular, it is undecidable whether $L_1 \rightarrowtail_F X = L_3$ has a solution or not.*

Bibliography

[1] L. M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187):1021–1024, November 1994.

[2] M. Anselmo and A. Restivo. On languages factorizing the free monoid. *International Journal of Algebra and Computations*, 6:413–427, 1996.

[3] E. Chiniforooshan, M. Daley, O. H. Ibarra, L. Kari, and S. Seki. Reversal-bounded counter machines and multihead automata: Revisited. in preparation, 2010.

[4] J. H. Conway. *Regular Algebra and Finite Machines*. Chapman & Hall, London, 1971.

[5] B. Cui, L. Kari, and S. Seki. Block insertion and deletion on trajectories. In preparation, 2009.

[6] M. Daley, O. Ibarra, and L. Kari. Closure and decidability properties of some language classes with respect to ciliate bio-operations. *Theoretical Computer Science*, 306:19–38, 2003.

[7] C. W. Dieffenbach and G. S. Dveksler, editors. *PCR Primer: A Laboratory Manual*. Cold Spring Harbor Laboratory Press, 2003.

[8] M. Domaratzki. Semantic shuffle on and deletion along trajectories. In C. S. Calude, E. Claude, and M. J. Dinneen, editors, *DLT 2004*, volume 3340 of *Lecture Notes in Computer Science*, pages 163–174. Springer, 2004.

[9] M. Domaratzki, G. Rozenberg, and K. Salomaa. Interpreted trajectories. *Fundamenta Informaticae*, 73:81–97, 2006.

[10] M. Domaratzki and K. Salomaa. Decidability of trajectory-based equations. *Theoretical Computer Science*, 345:304–330, 2005.

[11] A. Ehrenfeucht, T. Harju, I. Petre, and G. Rozenberg. Patterns of micronuclear genes in ciliates. In N. Jonoska and N. C. Seeman, editors, *DNA 7*, volume 2340 of *Lecture Notes in Computer Science*, pages 279–289. Springer, 2002.

[12] R. Freund, C. Martín-Vide, and V. Mitrana. On some operations on strings suggested by gene assembly in ciliates. *New Generation Computing*, 20:279–293, 2002.

[13] O. H. Ibarra. Reversal-bounded multicounter machines and their decision problems. *Journal of the ACM*, 25:116–133, 1978.

[14] L. Kari. *On Insertion and Deletion in Formal Languages*. PhD thesis, University of Turku, Department of Mathematics, SF-20500 Turku, Finland, 1991.

[15] L. Kari. On language equations with invertible operations. *Theoretical Computer Science*, 132:129–150, 1994.

[16] L. Kari and G. Thierrin. Contextual insertions/deletions and computability. *Information and Computation*, 131:47–61, 1996.

[17] Lila Kari and Gabriel Thierrin. Maximal and minimal solutions to language equations. *Journal of Computer and System Sciences*, 53:487–496, 1996.

[18] L. F. Landweber and L. Kari. The evolution of cellular computing: Nature's solution to a computational problem. In L. Kari, H. Rubin, and D. Wood, editors, *Proc. DNA-Based Computers IV*, pages 3–13, 1999.

[19] M. O. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3:114–125, 1959.

[20] A. Salomaa and S. Yu. On the decomposition of finite languages. In G. Rozenberg and W. Thomas, editors, *Developments in Language Theory*, pages 22–31, 1999.

Part IV

Discussion

Chapter 10

Discussion

The implications of taking into consideration particularities of biomolecular information encoding and processing have been studied in this thesis. In particular, we presented the contributions obtained in the two research directions:

1. extensions of the notions and results in combinatorics on words being inspired by biological information encoding;
2. mathematical modeling of bio-operations.

Primary contributions, in the first direction, are the extended notions of power and primitivity, namely, θ-power and θ-primitivity. These extended notions, in turn, enabled us to consider the Fine and Wilf's theorem and Lyndon-Schützenberger equation in more general settings. We proved the extended Fine and Wilf's theorem with a bound and its improvement, and showed the goodness of the latter. Furthermore, the optimality of the latter was analyzed. As for the extended Lyndon-Schützenberger equation, we characterized its parameters over which the equation is solved positively and the ones over which it is solved negatively. Throughout the process to reach these goals, we proved various properties of θ-powers and θ-primitive words. Among the most important are the characterizations of languages equations over θ-primitive word(s). As illustrated in the encoding set design problem, a DNA molecule and its WK-complement are often required to be handled in a unified manner. Our primary contributions and their proofs would enable us to directly approach to or at least give some hint to the problem of how to handle the complementary molecules uniformly.

Along the second direction, we modeled three bio-operations: pseudoknot formation and two phenomena, duplication and insertion/deletion. The topics we investigated on these operations include closure properties of language classes under these operations and language equations involving these operations (decidability of the existence of the solutions, and algorithms to solve the solvable equations).

Let us conclude this thesis by enumerating questions which have been left open in this thesis and future directions. In Chapter 5, we completely characterized all pairs of integers p, q for which the improved bound $b'(p, q)$ is optimal, but this characterization exemplified that the bound is NOT strongly optimal. Thus, the strongly optimal bound should be formulated mathematically. The extended Fine and Wilf's theorem can be generalized further, for examples, as follows:

Problem 10.1. Let θ be an antimorphic involution θ and $k \geq 2$. Find a function $f : \mathbb{N}^k \to \mathbb{N}$ such that for given θ-primitive words $v_1, v_2, \ldots, v_k$, if θ-powers of $v_1, v_2, \ldots, v_n$ share a common prefix of length $f(|v_1|, \ldots, |v_k|)$, then $\rho_\theta(v_1) = \rho_\theta(v_2) = \cdots = \rho_\theta(v_n)$.

The extended Lyndon-Schützenberger equation has not been solved yet on parameters $(3, n, m)$ for $n, m \geq 3$.

Problem 10.2. For an antimorphic involution θ, θ-primitive words u, v, w, and integers $n, m \geq 3$, let $v_1, \ldots, v_n \in \{v, \theta(v)\}$ and $w_1, \ldots, w_m \in \{w, \theta(w)\}$. Determine whether the equation

$$u_1 u_2 u_3 = v_1 \cdots v_n w_1 \cdots w_m$$

implies that $u, v, w \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$ or not.

The extended Lyndon-Schützenberger equation can also be generalized further as follows: Given an antimorphic involution θ, $k \geq 2$, θ-primitive words $u, v_1, v_2, \ldots, v_k$, and positive integers $n, n_1, n_2, \ldots, n_k \geq 1$, we can consider the following equation:

$$u_1 u_2 \cdots u_n = v_{1,1} \cdots v_{1,n_1} v_{2,1} \cdots v_{2,n_2} \cdots v_{k,1} \cdots v_{k,n_k}, \tag{10.1}$$

where $u_1, \ldots, u_n \in \{u, \theta(u)\}$ and $v_{i,1}, \ldots, v_{i,n_i} \in \{v_i, \theta(v_i)\}$ for $1 \leq i \leq k$.

Problem 10.3. Find a necessary and sufficient condition on $n, n_1, n_2, \ldots, n_k$ such that for any antimorphic involution θ and arbitrary θ-primitive words $u, v_1, \ldots, v_k$, Eq. (10.1) implies that $u, v_1, \ldots, v_k \in \{t, \theta(t)\}^*$ for some $t \in \Sigma^+$.

One can state that our results in Chapters 3-6 center around the language equation or the system of language equations which forces some of the involved words to be in the set $\{t, \theta(t)\}^*$ for some word $t \in \Sigma^+$. We can say that such a word equation or a system of equations has a *defect effect with respect to* θ, and can call results about this effect *defect theorems with respect to* θ. Other word equations as well as equation systems on words deserve further investigation of whether they have defect effects with respect to θ.

In Chapter 7, H-type pseudoknot formation was modeled by the notion of pseudoknot-bordered words, which are of the form $w = xy\alpha = \beta\theta(x)\theta(y)$. Although this model enabled us to investigate the words of the form $xy\gamma\theta(x)\theta(y)$, this is only a special case of H-type pseudoknots, which should be modeled as $x\alpha y\beta\theta(x)\gamma\theta(y)$ with α, γ not always being empty. Therefore, we need a more appropriate model for H-type pseudoknots formation. In addition, there exist non-H-type pseudoknots, though H-type ones are most typical among all the pseudoknots. It is an interesting topic to establish some general framework to formalize pseudoknots.

Controlled duplication was considered in Chapter 8, and several interesting sufficient conditions were found on the control set C under which $L^{\heartsuit(C)}$ is regular for any regular language L. Our claim that $L^{\heartsuit(C)}$ is regular for any regular language L if and only if there exists a finite control set C' such that $L^{\heartsuit(C)} = L^{\heartsuit(C')}$ was negatively solved by an example. Thus, the following problem remains open.

Problem 10.4. Find a necessary and sufficient condition under which $L^{\heartsuit(C)}$ is regular for any regular language L.

One significant and challenging problem, which was left open in Chapter 9, is to solve the two-variables language equation with p-schema-based parallel deletion of the following form:

$$X \rightarrowtail_F Y = L_3$$

for regular language L_3, regular p-schema F, and two variables X, Y. To our knowledge, even with the specific case when $F = F_{\text{cat}} = \Sigma^* \times \lambda$, i.e., the case when $\rightarrowtail_F$ is right quotient, it is not known whether the existence of the solution to this equation is decidable or not.

As done for insertion/deletion in Chapter 9, one can extend duplication as a parallel operation called *parallel duplication*, which maps a word $u_1x_1u_2x_2\cdots u_{k-1}x_{k-1}u_k$ into $u_1x_1^2u_2x_2^2\cdots u_{k-1}x_{k-1}^2u_k$. Without any modification, p-schema makes it possible for us to define this parallelized operation as: for a p-schema F and a language L,

$$L^{\heartsuit_F} = \{u_1x_1^2u_2x_2^2\cdots u_{k-1}x_{k-1}^2u_k \mid u_1x_1u_2x_2\cdots u_k \in L, (u_1, u_2, \ldots, u_k) \in F\}.$$

We can further strengthen parallel duplication with control set.

Unquestionably, the most essential concept throughout this thesis was the antimorphic involution θ as a formal model of Watson-Crick complementarity. WK-complementarity refers to the bonds A-T and C-G, but *in vivo* other bonds such as G-T are thermodynamically-favored and hence found often. Once taking G-T into consideration, a proper model of bonds between nucleotides will be not a function but a relation. Any problems investigated in all the chapters but Chapters 8 and 9 in this thesis can be generalized naturally by replacing the antimorphic involution θ with the relation. In particular, the equivalence between a word and its complement is generalized as the equivalence among words which are in the relation.

Appendices

Copyright releases

The contents of Chapters 3 and 7 were published by Elsevier. According to the following page of Elsevier,

- http://www.elsevier.com/wps/find/supportfaq.cws_home/rightsasanauthor

the authors retain the right to include the article in their dissertations.

The contents of Chapter 5 were published by IOS press. According to the following page of IOS press,

- http://www.iospress.nl/authco/copyright.html

copyright remains the author's.

The contents of Chapters 4, 8, and 9 were published by Springer. I sent them a request for their permission for me to use these contents for this dissertation and I got the following kind permission of Springer Science + Business Media.

```
Dear Sir,

Congratulations.

Please proceed as requested.

Best regards,

Nel van der Werf (Ms)
Rights and Permissions/Springer

Van Godewijckstraat 30 | P.O. Box 17
3300 AA Dordrecht | The Netherlands
tel? +31 (0) 78 6576 298???
```

fax +31 (0)78 65 76-377

Nel.vanderwerf?@springer.com
www.springer.com

-----Original Message-----
Subject: Permission for the reuse of my papers for my thesis (ASAP)

Dear Dr. Straalen,

Good evening, my name is Shinnosuke Seki.
I hope this message finds you well.

Today, I successfully defended my Ph.D. thesis
at the University of Western Ontario,
which includes the fulltexts of the following three my publications
in Springer Verlag.

(1)
M. Ito, L. Kari, Z. Kincaid, S. Seki,
Duplication in DNA sequences,
In: A. Condon, D. Harel, J. N. Kok, A. Salomaa, E. Winfree (Eds.)
Algorithmic Bioprocesses, Natural Computing Series, Springer (2009) 43-61.
ISBN 978-3-540-88868-0

(2)
L. Kari, S. Seki,
Schema for parallel insertion and deletion,
In: Proceedings of DLT 2010 London, Canada, August 17-20, 2010.
Lecture Notes in Computer Science (LNCS) 6224, Springer, 267-278, 2010.

(3)
E. Czeizler, E. Czeizler, L. Kari, S. Seki,
An extension of the Lyndon Schutzenberger result to pseudoperiodic words,
In: Proceedings of DLT 2009 Stuttgart, Germany, June 30-July 3, 2009
LNCS 5583, Springer, 183-194, 2009.
ISBN: 978-3-642-02736-9

I would like to print out 4 copies of my thesis
for the university (2 copies), for the department, and for my parents.

Can I have your permission to use these publications' full texts
for this purpose?
I have already got a permission from the all the coauthors.
I have to submit the final version of my thesis on August 13th (Fri) 2010.
I really appreciate your quick answer to this message.

Yours respectfully,
Shinnosuke Seki

Curriculum Vitae

Academic Activities

Education

Ph.D.	Computer Science, The University of Western Ontario, Canada, September 2006 - Expected graduation August 2010 Supervisor: Lila Kari
M.Eng.	Computer Science, The University of Electro-Communications, Japan, 2006 Thesis: "A Grammatical Approach to the Alignment of RNA Secondary Structures Including Pseudoknots" Supervisor: Satoshi Kobayashi
B.Eng.	Computer Science, The University of Electro-Communications, Japan, 2004 Thesis: "Efficient Learning of k-reversible Context-free Grammars from Positive Structural Examples" Supervisor: Satoshi Kobayashi

Awards

- **Presentation award at UWORCS, the annual workshop at Department of Computer Science, UWO**, 2008, 2009.

Leadership

- **Organizing Committee Member** for "Developments in Language Theory (DLT 2010)", London, Canada, August 17-20, 2010

- **Lead TA** for "Computer Science 1032a - Information Systems and Design", January 2008 - Now

  Throughout my Ph.D. studies at UWO, I have been a teaching assistant (TA) of CS1032A "Information Systems and Design". This is a first year undergrad course to teach the basics of computer science such as how to write html and how to use Microsoft Office software. Since the 2008 winter term (January-April, 2008), I have been appointed as the organizer of the course for four consecutive terms till now. This course has about 600 students with 25 TAs. My job as organizer includes office hours as well as the management of the TAs (TA office hour schedule control, workload allocation).

Academic Grades in the doctoral program

Term	Course Title	Score
2006 Fall	COMPSCI 663A Computational Biology	94
2007 Winter	COMPSCI 662B DNA Computing	100
	COMPSCI 830B Coding Theory	98
2007 Fall	COMPSCI 834A Synthetic Biology	95
2008 Winter	COMPSCI 623B Information Theory	95
2009 Winter	COMPSCI 9620 Advanced Automata Theory	98

Invited lectures

- Schema for parallel insertion and deletion,
  Talk at Workshop on Formal Languages and Automata III, Kyoto Sangyo University, December 12th, 2009

- An extension of fundamental notions in combinatorics on words inspired by DNA information encoding,
  Universität Potsdam, July 6th, 2009.

- Encoding information for DNA computing,
  Department of Computer Science, University of Western Ontario, CS 9562b/4462b, February 13th, 2009.

Paper reviews

I have reviewed papers submitted to the following journals: Information and Computation, International Journal of Computer Mathematics, International Journal on Artificial Intelligence Tools, Journal of Automata, Languages, and Combinatorics, Journal of Combinatorial Optimization, Journal of Computational Biology, Journal of Royal Society Interface, Journal of Universal Computer Science, Natural Computing, New Generation Computing, Theoretical Computer Science, and Transactions on Parallel and Distributed Systems.

I have reviewed papers submitted to the following conferences: Automata and Formal Languages (AFL), Descriptional Complexity of Formal Systems (DCFS), Developments in Language Theory (DLT), DNA, International Conference on implementation and application of automata (CIAA), and Unconventional Computation (UC).

Research

Research Interests

Theory of Computation
DNA computing, DNA tile self-assembly.

Discrete Mathematics in relation to Computer Science
Language operations, Combinatorics on words

Research Projects

Theory of Computation, Canada — 2006-Now
DNA tile self-assembly
DNA computing

Formal Language Theory, Canada — 2006-Now
Combinatorics on words
Language operations and language equations

RNA Secondary Structure Analysis, Japan — 2004-06
Efficient alignment of RNA secondary structures
Pseudoknots
Multiple alignment

Computational Learning Theory, Japan — 2003-04
Learning from positive structural examples

Referred book chapters

- L. Kari, S. Seki, P. Sosík,
  DNA Computing: Foundations and Implications,
  in: G. Rozenberg, T. Bäck, J. Kok (Eds.), Handbook of Natural Computing, to appear.

Editorial works

- Y. Gao, H. Lu, S. Seki, S. Yu (Eds.),
  Proceedings of 14th International Conference on Developments in Language Theory (DLT 2010) London, Ontario, Canada, August 17-20,
  Lecture Notes in Computer Science, vol. 6224, Springer-Verlag, 2010.

Refereed journal publications

1. L. Kari, S. Seki. An improved bound for an extension of Fine and Wilf's theorem. *Fundamenta Informaticae* 101(3) (2010) 215-236.

2. E. Czeizler, L. Kari, S. Seki. On a special class of primitive words. *Theoretical Computer Science* 411(3) (2010) 617-630.

3. M. Ito, L. Kari, Z. Kincaid, S. Seki. Duplication in DNA sequences. in: A. Condon, D. Harel, J. N. Kok, A. Salomaa, E. Winfree (Eds.) *Algorithmic Bioprocesses*, Natural Computing Series (2009) 43-61.

4. L. Kari, K. Mahalingam, S. Seki. Twin-roots of words and their properties. *Theoretical Computer Science* 410 (2009) 2393-2400.

5. L. Kari, S. Seki. On pseudoknot words and their properties. *Journal of Computer and System Sciences* 75 (2009) 113-121.

6. S. Seki, S. Kobayashi. A Grammatical Approach to the Alignment of Structure-Annotated Strings. *IEICE (Institute of Electronics, Information and Communication Engineers) Transactions on Information and Systems* E88-D (12) (2005) 2727-2737.

Refereed conference proceedings publications

1. L. Kari and S. Seki. Schema for parallel insertion and deletion. In: Proceedings of *14th International Conference on Developments in Language Theory (DLT 2010)*, London, Canada, August 17-20th, 2010. Y. Gao, H. Lu, S. Seki, and S. Yu (Eds.), LNCS 6224 (2010), Springer, pages 267-278.

2. E. Chiniforooshan, D. Doty, L. Kari, and S. Seki. Scalable, time-responsive, digital, energy-efficient molecular circuits using DNA strand displacement. In: Preliminary Proceedings of *The 16th International Meeting on DNA Computing and Molecular Programming (DNA 16)*, HongKong, China, June 14-17th, 2010, pages 162-173.

3. L. Kari, S. Seki, and Z. Xu. Triangular tile self-assembly systems. In: Preliminary Proceedings of *The 16th International Meeting on DNA Computing and Molecular Programming (DNA 16)*, HongKong, China, June 14-17th, 2010, pages 173-182.

4. B. Cui, L. Kari, and S. Seki. On the reversibility of parallel insertion, and its relation to comma codes. In: Proceedings of *3rd International Conference on Algebraic Informatics (CAI 2009)*, Thessaloniki, Greece, May 19-22, 2009.

S. Bozapalidis and G. Rahonis (Eds.), LNCS 5725 (2009), Springer, pages 204-219.

5. E. Czeizler, E. Czeizler, L. Kari, and S. Seki. An extension of the Lyndon Schutzenberger result to pseudoperiodic words. In: Proceedings of *13th International Conference on Developments in Language Theory (DLT 2009)*, Stuttgart, Germany, June 30-July 3, 2009. V. Diekert and D. Nowotka (Eds.), LNCS 5583 (2009), Springer, pages 183-194.

6. M. Ito, L. Kari, Z. Kincaid, and S. Seki. Duplication in DNA sequences. In: Proceedings of *12th International Conference on Developments in Language Theory (DLT 2008)*, Kyoto, Japan, September 16-19, 2008. M. Ito and M. Toyama (Eds.), LNCS 5257 (2008), Springer, pages 419-430.

7. E. Czeizler, L. Kari, and S. Seki. On a special class of primitive words. In: Proceedings of *Mathematical Foundations of Computer Science (MFCS 2008)*, Torun, Poland, August 25-29, 2008. G. Goos, J. Hartmanis, and J. v. Leeuwen (Eds.), LNCS 5162 (2008), Springer, pages 265-277.

8. L. Kari and S. Seki. Towards the sequence design preventing pseudoknot formation. In: Proceedings of *2nd International Workshop on Natural Computing (IWNC 2007)*, Nagoya, Japan, December 10-12, 2007. PICT1 (2009), Springer, pages 101-110.

9. S. Kobayashi and S. Seki. An efficient multiple alignment method for RNA secondary structures including pseudoknots. In: Proceedings of *2nd International Workshop on Natural Computing (IWNC 2007)*, Nagoya, Japan, December 10-12, 2007. PICT1 (2009), Springer, pages 179-188.

10. T. Takakura, H. Asakawa, S. Seki, and S. Kobayashi. Efficient tree grammatical modeling of RNA secondary structures from alignment data. Poster *Recomb2005* (2005), pages 339-340.

11. S. Seki, A. Kijima, S. Kobayashi, and G. Sanpei. Hierarchical Alignment of RNA Secondary Structures Including Pseudoknots. In: Proceedings of *15th International Conference on Genome Informatics (GIW-2004)*, Yokohama, Japan, December 13-15, 2004, pages 117-1 - 117-2.

12. S. Seki and S. Kobayashi. Efficient Learning of k-reversible Context-Free Grammars from Positive Structural Examples. In: Proceedings of *7th International Colloquium on Grammatical Inference (ICGI-2004)*, NCSR "Demokritos", Athens, Greece, October 11-13, 2004. G. Paliouras and Y. Sakakibara (Eds.), LNAI 3264 (2004), Springer, pages 285-287.

Other publications (non-refereed)

1. E. Czeizler, E. Czeizler, L. Kari, S. Seki.
   An extension of Lyndon-Schützenberger's result to pseudoperiodic words. ISBN: 978-0-7714-2699-5. UWO Technical Report TR-722, 2009.

2. S. Seki. Natural Computing. In Wikipedia (English) with Prof. Lila Kari.

Under Review

1. L. Kari, B. Masson, S. Seki. Properties of pseudo-primitive words and their applications.

2. E. Czeizler, E. Czeizler, L. Kari, S. Seki. An extension of the Lyndon Schutzenberger result to pseudoperiodic words.

3. B. Cui, L. Kari, S. Seki. Block insertion and deletion on trajectories.

4. M. Daley, L. Kari, S. Seki, P. Sosìk, Orthogonal shuffle on trajectories.

5. N. Bryans, E. Chiniforooshan, D. Doty, L. Kari, S. Seki, The power of nondeterminism in self-assembly.