Tomi Kärki

Similarity Relations on Words: Relational Codes and Periods

Turku Centre *for* Computer Science

TUCS Dissertations
No 98, January 2008

Similarity Relations on Words: Relational Codes and Periods

Tomi Kärki

To be presented, with the permission of the Faculty of Mathematics and Natural Sciences of the University of Turku, for public criticism in Auditorium XXI on February 22nd, 2008, at 12 noon.

University of Turku
Department of Mathematics
FIN-20014 Turku, Finland

2008

Supervisors

Professor Tero Harju
Department of Mathematics
University of Turku
FIN-20014 Turku
Finland

Doctor Vesa Halava
Department of Mathematics
University of Turku
FIN-20014 Turku
Finland

Reviewers

Professor Lucian Ilie
Department of Computer Science
University of Western Ontario
Middlesex College 368
London, Ontario, N6A 5B7
Canada

Professor Wojciech Rytter
Institute of Informatics
University of Warsaw
ul. Banacha 2
02-097, Warsaw
Poland

Opponent

Professor Michel Rigo
Institut de Mathématiques
Université de Liège
Grande Traverse 12 (B.37)
B-4000, Liège
Belgium

ISBN 978-952-12-2027-2
ISSN 1239-1883

To my family: Anniina and Jaakko

Abstract

In this thesis we introduce the notion of a similarity relation on words induced by a reflexive and symmetric relation on letters. As a motivation for the research, we propose several applications arising from computer science and molecular biology. In the first part of the thesis, the theory of variable length codes is revisited by generalizing codes to (R, S)-codes, where R and S are arbitrary similarity relations. In particular, we study the relationally free monoids and defect theorems. The second part of the thesis is devoted to interaction properties of periods with respect to similarity relations. We define global, external and local relational periods and prove several relational variations of the famous theorem of Fine and Wilf.

Keywords: combinatorics on words, similarity relation, code, free hull, defect theorem, period, the theorem of Fine and Wilf, partial word.

Acknowledgements

I wish to express my deepest gratitude to my supervisors, Professor Tero Harju and Doctor Vesa Halava. Without their guidance and broad knowledge this thesis would not have been possible. They have always had time for me when I have needed help or opinions. Their personality and sense of humor have made our academic teamwork very much enjoyable. I also want to thank Professor Juhani Karhumäki, the leader and promoter of our research group, for introducing me the research area and the international research community of combinatorics on words and for helping me in many ways during my carrier.

Special thanks are due to Professor Lucian Ilie from University of Western Ontario, Canada and Professor Wojciech Rytter from University of Warsaw, Poland for kindly accepting to review my thesis and for their useful remarks. I would also like to thank Doctor Paul Bell for carefully reading the thesis manuscript and correcting my language. It is a great honour for me that Professor Michel Rigo from Université de Liège, Belgium accepted to act as the opponent for the public defence of this disputation. I would also like to thank Doctor Julien Cassaigne from Institut de Mathématiques de Luminy, France and Professor Luca Q. Zamboni from University of North Texas, USA for giving me an opportunity to learn about scientific culture abroad and for the successfully started collaboration.

The Department of Mathematics and Turku Centre for Computer Science have provided outstanding working conditions. I thank the staff, all my friends and colleagues, for creating an excellent atmosphere for the research. In particular, I owe my gratitude to my two office mates, Doctor Ari Renvall and Ph.D. Student Riku Klén for their invaluable support and unselfish devotion to solving any problems I have had. I also want to thank the department sports club Luiskaotsat and our badminton team for offering challenges besides work.

Finally, I thank my entire family, my parents, grandmother and my family-in-law for always supporting and believing in me. Especially, I want to thank my wife Anniina for her endless encouragement, patience and love.

Turku, January 2008 Tomi Kärki

Notation

ε	the empty word	5
$\mathcal{A}^*$	finite words over $\mathcal{A}$	5
$\mathcal{A}^+$	nonempty finite words over $\mathcal{A}$	5
$\mathcal{A}^\omega$	infinite words over $\mathcal{A}$	5
$\lvert w\rvert$	length of the word w	6
$\overline{w}$	reversal of w	6
$\mathrm{Alph}(w)$	letters occurring in w	6
$F_n(w)$	factors of length n of w	6
$\mathrm{pref}_k(v)$	prefix of length k of v	6
$\mathrm{Pref}(v)$	prefixes of v	6
$\mathrm{suf}_k(v)$	suffix of length k of v	6
$\mathrm{Suf}(v)$	suffixes of v	6
$u^{-1}w$	left quotient of w by u	6
wu^{-1}	right quotient of w by u	6
$\mathbb{Z}_+$	positive integers	6
$x\,R\,y$	x is R-related to y	7
R_Y	restriction of R on Y	7
2^X	power set of X	7
$R(Y)$	elements R-compatible with elements of Y	7
$\mathbb{Z}$	integers	7
ι	identity relation	8
Ω	universal similarity relation	8
$\mathrm{Base}(M)$	base $(M \setminus \{\varepsilon\}) \setminus (M \setminus \{\varepsilon\})^2$ of the monoid M	9
$\pi(w)$	the minimal period of w	11
$\gcd(p, q)$	greatest common divisor of p and q	11
v^ω	infinite power of v	11
$p_w(n)$	subword complexity function of w	12
$\langle r_1, \ldots, r_n\rangle$	similarity relation generated by $r_1, \ldots, r_n$	14
$D(w)$	domain of a partial word w	16
$H(w)$	holes of a partial word w	16
$w_\diamond$	companion of a partial word w	16
$x \uparrow y$	x and y are compatible	17

$\mathrm{sz}(R)$	size of a similarity relation R	31
$A_{\max}(X,S)$	maximal alteration relations	32
$F_{\min}(X,R)$	minimal fidelity relations	32
$M_R(X,S)$	maximal size of the relations in $A_{\max}(X,S)$	36
$\widehat{I}_{R,S}(X)$	the inner (R,S)-hull of X	47
$\widehat{O}_{R,S}(X)$	the outer (R,S)-hull of X	47
$\widehat{I}_R(X)$	the inner (R,R)-hull of X	47
$\widehat{O}_R(X)$	the outer (R,R)-hull of X	47
$\widehat{F}_{R,S}(X)$	the (R,S)-free hull of X	48
$\widehat{F}_R(X)$	the (R,R)-free hull of X	48
$C^i_{R,X}(Y)$	nontrivial inner R-matches for Y over X	49
$C^o_{R,X}(Y)$	nontrivial outer R-matches for Y over X	49
$D^i_{R,X}(u,Y)$	a set related to a chain of trivial inner R-matches	49
$D^o_{R,X}(u,Y)$	a set related to a chain of trivial outer R-matches	49
$G_R(X)$	graph representing relations R_X	55
R^+	transitive closure of R	55
$c(X,R)$	number of connected components of $G_R(X)$	55
$S_{\mathcal{R}}(X_1,\ldots,X_n)$	generalized Spehner graph	60
$S_R(X)$	simplified Spehner graph	62
$\pi_{R,g}(x)$	the minimal global R-period of w	72
$\pi_{R,e}(x)$	the minimal external R-period of w	72
$\pi_{R,l}(x)$	the minimal local R-period of w	72
$B_g(p,q)$	bound of global-global interaction	78
$[n]_q$	the least positive residue of an integer $n \pmod q$	80
$B_l(p,q)$	bound of global-local interaction	84
$B_e(p,q)$	bound of global-external interaction	88
$C_g(p,q)$	bound of external-global interaction (holding p)	92
$C_l(p,q)$	bound of external-local interaction (holding p)	92
$C(p,q)$	bound of external-external interaction	93
$C_e(p,q)$	bound of external-external interaction (holding p)	94
$\overline{C}(p,q)$	bound of external-external interaction (inclusive p)	98
$D_l(p,q)$	bound of local-local interaction	102
$D_e(p,q)$	bound of local-external interaction	103
$D_g(p,q)$	bound of local-global interaction	103
$FW(p,q)$	extremal relational Fine and Wilf words	106

Contents

Chapter 1

Introduction

Some hundred years after Thue's papers on nonrepetitive sequences of symbols [75, 76], often considered as the starting point of mathematical research on words, the field of combinatorics on words has established its role as an independent research area under discrete mathematics related to computer science (68R15 in the 2000 Mathematics Subject Classification). Only a few scattered papers on words appeared in the first half of the 20th century, and systematic research commenced as late as the 1950s when Schützenberger in France began developing the theory of codes [69] and Adian and Novikov in Russia were solving the Burnside problem for groups; see [3]. The first uniform representation of word combinatorics was the book published by a group of researchers under the pseudonym Lothaire in 1983 [57]. Since then, the volume of research in this area has been growing rapidly. As an example, we mention the continued works of Lothaire [58, 59], the survey of Choffrut and Karhumäki [27], the more recent tutorial with open problems by Berstel and Karhumäki [7], and the biennial international conference WORDS dedicated to this subject.

The characteristic flavour of the field originates from the noncommutativity and discreteness of the research objects, i.e., finite or infinite sequences of symbols normally regarded without any semantics. By the operation of concatenation, words can be naturally seen as algebraic objects in a free semigroup or in a free monoid generated by some alphabet. On the other hand, they are also central objects in any standard model of computing. On the border between mathematics and computer science, combinatorics of words has also many connections to other topics, such as algorithmics, mathematical game theory, discrete dynamical systems, transcendental questions in number theory as well as crystallography in physics and DNA sequences in biology.

In this thesis we consider similarity relations on words induced by compatibility relations on letters. By a compatibility relation R we mean a

reflexive and symmetric relation. Two words are R-similar if they are of the same length and their corresponding letters are pairwise R-compatible. For example, if c is related to t, then the words "cube" and "tube" are considered similar. Similarity relations generalize the compatibility notion of partial words. Partial words are sequences with a "do not know"-symbol $\diamond$ which is compatible with all other letters of the alphabet. These words were introduced by Berstel and Boasson in 1999 [6] and combinatorics on partial words has been widely studied in recent years, e.g., see references in [15]. Moreover, the first book on the subject has been recently authored by Blanchet-Sadri [14]. We study similarity relations mainly from two perspectives. On one hand we consider relations acting on a set of finite words, and on the other hand, relations on periods occurring in a single finite or infinite word.

To begin with, our aim is to generalize some constituent parts of the theory of variable length codes using similarity relations. Denote the set of letters by $\mathcal{A}$. The object of the theory is to study factorizations of words into sequences of words taken from a given set. In the monoid X^* generated by a code $X \subseteq \mathcal{A}^+$, there do not exist two distinct factorizations over X for any word. Our approach is to strengthen the coding property by requiring that two "nearly similar" words, have the same, or at least "similar," factorizations. This is attained by using similarity relations. Generalizing codes to relational codes enables us to model situations where some of the letters in a message are changed to related letters, but the message can still be factorized, in other words, decoded in a proper manner. In an (R, S)-code the similarity relation R illustrates possible changes in a message and the relation S describes the correctness of a decoded message.

The ultimate goal of the study is to prove a modified defect effect for words with similarity relations. By a defect effect, we mean the result, often considered to be folklore, that: If a set of n words satisfies a nontrivial relation then these words can be written as products of at most $n-1$ words. Actually, there exist several defect theorems depending on the restrictions that are put on the $n-1$ words; see [48]. A typical formulation of the defect effect is to say that the rank of the smallest free monoid containing a set of words X is strictly smaller than the cardinality of X if and only if X is not a code. By rank we mean the cardinality of the base of the monoid. The smallest free monoid containing X is the free hull of X. This leads us to introduce (R, S)-unique factorization extensions, (R, S)-free monoids and (R, S)-free hulls. For the generalized defect effect, we replace the notion of rank by the cardinality $c(B, R)$ of equivalence classes of the base B for the transitive closure of a similarity relation R. We succeed in showing that the cardinality $c(B, R)$ for the (R, S)-free hull of X is strictly smaller than $c(X, R)$ if and only if X is not an (R, S)-code. Moreover, we are able to prove a defect theorem for (R, S)-unique factorization hulls, which induces

a cumulative defect effect for (R, S)-free hulls.

In the second part of the thesis, we consider interaction properties of periods with respect to similarity relations. A period of a word is a positive integer p such that all letters in the word occurring in positions congruent modulo p must be equal. By the interaction property we mean that if a sufficiently long word has two periods then it also has another nontrivial derived period depending on the original periods. A basic example of a theorem of this type is the theorem of Fine and Wilf [36], one of the cornerstones in combinatorics on words: If a word w has periods p and q, and is of length at least $p + q - \gcd(p, q)$, then w has also the period $\gcd(p, q)$. Generalizing this theorem was the starting point of the study of partial words. Our aim is to give new insight to this topic by proving several variations of Fine and Wilf's theorem as an example of an interaction property between a pure period and a relational period. We introduce three types of periods, namely global, external and local relational periods and compare their properties by analyzing different interaction cases.

Basic concepts and classical theorems on word combinatorics needed for subsequent chapters are introduced in Chapter 2.

In Chapter 3 we define the main object of the thesis, i.e., the concept of a similarity relation on words. As a motivation, we discuss applications of similarity relations in two distinct disciplines, namely in theoretical computer science and in molecular biology. We show that similarity relations generalize the notion of a partial word and describe how relations can be used in modeling error correcting capabilities of variable length codes. In bioinformatics, similarity relations are connected to DNA and protein synthesis including methods of sequence alignment and DNA sequencing. Similarity relations and a short description of applications were originally presented in the joint article [44]. However, in this chapter, applications are discussed in more detail.

Chapter 4 is devoted to the theory of relational codes. First, we define (R, S)-codes for arbitrary similarity relations R and S and consider algorithmic questions on these codes. We especially consider a modification of the classical algorithm of Sardinas and Patterson. We also introduce algorithms to analyze coding properties of relational codes in more detail. Furthermore, we show that the MAXIMAL RELATION problem is NP-complete. In this problem, one is given a finite set X and a similarity relation S, and one is to determine whether X is an (R, S)-code for some compatibility relation R induced by at least k related pairs of letters. The theory of free monoids and hulls is revisited starting from (R, S)-unique factorization of elements in a submonoid of $\mathcal{A}^*$. A modified Schützenberger's stability criterion and Tilson's closure result are proved. Moreover, we show that under some restrictions there exists the smallest monoid in $\mathcal{A}^*$ where a set $X \subseteq \mathcal{A}^+$ can be factorized (R, S)-uniquely. The inner and the outer (R, S)-unique fac-

torization hulls and the (R, S)-free hull of a set of words X are defined and procedures for constructing these monoids are given. Finally, we prove a defect effect concerning (R, S)-unique factorization hulls and a cumulative defect theorem of (R, S)-free hulls is proved as a corollary. Consequently, a defect theorem of partial words follows. In the last section of the chapter generalized Spehner graphs are used for describing implementations of algorithms for finding hulls and testing (R, S)-codes. This chapter is mainly based on my article [51] and the joint works [44,45]. The generalized Spehner graphs, already introduced in my paper [53], are treated in a more general form enabling new efficient algorithms.

Properties of relational periods are discussed in Chapter 5. After introducing three types of relational periods, we consider bounds on period interactions. The properties of different relational periods are compared by proving several variations of the theorem of Fine and Wilf. In the end, we also consider relational analogues for so called extremal Fine and Wilf words, i.e., non-unary words of maximal length with two coprime periods. The material of this chapter is based on the joint works [42, 43, 46].

Conclusions and some future perspectives are given in the final chapter.

Chapter 2

Preliminaries

In this chapter we introduce the basic concepts and notation used throughout the thesis. The main objects under consideration are words and relations. Words are central elements in all standard models of computing. Our approach to these discrete structures is both combinatorial and algebraic as is usual in the mathematical context of combinatorics on words. In the following chapters we concentrate on two classical research topics of word combinatorics, namely coding properties and periodicity. Short introductions to these subjects are provided in separate sections. We confine ourselves to dealing only with binary relations on words. No deeper knowledge of the theory of relations will be needed.

For further information about the field of combinatorics on words, see the classical reference, the first book of Lothaire [57] and the more recent expositions, the tutorial by Berstel and Karhumäki [7] and the survey of Choffrut and Karhumäki [27]. A more algorithmic point of view is given in the third book of Lothaire [59]. The main reference for the theory of codes is the book by Berstel and Perrin [8]. Chapter 8 in Lothaire's second book [58] is devoted to different aspects of periodicity.

2.1 Words

An *alphabet* $\mathcal{A}$ is a nonempty finite set of symbols. These symbols are called *letters*. A *word* over $\mathcal{A}$ is a finite or infinite sequence of letters from $\mathcal{A}$. The empty sequence is called the *empty word* and is denoted by ε. The sets of all finite words, finite nonempty words and infinite words over $\mathcal{A}$ are denoted by $\mathcal{A}^*$, $\mathcal{A}^+$ and $\mathcal{A}^\omega$, respectively. Note that $\mathcal{A}^* = \mathcal{A}^+ \cup \{\varepsilon\}$.

The *concatenation* or *catenation* of finite words u and v is the word uv obtained by writing the latter word at the end of the former word. Given three words u, v and w, it is clear that this binary operation is associative, i.e., $(uv)w = u(vw)$. Hence, the parenthesis can always be omitted. How-

ever, parenthesis are occasionally used for the sake of clarity. Catenating a word w with itself k times is abbreviated by w^k. In particular, $w^0 = \varepsilon$. Furthermore, for an integer m and a word $w = w_1 \cdots w_n$, where $w_i \in \mathcal{A}$ for $1 \leq i \leq n$, the *rational power* $w^{m/n}$ is $w^q w_1 \cdots w_r$, where $m = qn + r$ for $0 \leq r < n$. The set $\mathcal{A}^+$ is a semigroup with catenation as its associative operation. Furthermore, the empty word satisfies

$$\varepsilon w = w\varepsilon = w$$

for every element $w \in \mathcal{A}^*$. Hence, $\mathcal{A}^*$ is a monoid with ε as its identity element. Since every word has a unique representation as catenation of letters, the word semigroup $\mathcal{A}^+$ is the *free* semigroup generated by $\mathcal{A}$ with respect to catenation. Similarly, the word monoid $\mathcal{A}^*$ is a free monoid.

The *length* of a word w, denoted by $|w|$, is the total number of (occurrences of) letters in w. In other words, if $w = w_1 \cdots w_n$ with $w_i \in \mathcal{A}$, $1 \leq i \leq n$, then $|w| = n$. In particular, the length of the empty word is zero. The *reversal* of a word $w = w_0 w_1 \cdots w_{n-1} w_n$ is $\overline{w} = w_n w_{n-1} \cdots w_1 w_0$. The set of letters occurring in w is denoted by $\mathrm{Alph}(w)$. A word u is a *factor* of a word v (resp. a left factor or a *prefix*, a right factor or a *suffix*), if there exist words x and y such that $v = xuy$ (resp. $v = uy, v = xu$). The set of factors of length n of a word u, is denoted by $F_n(w)$. For $0 \leq k \leq |v|$, the prefix of length k of v is denoted by $\mathrm{pref}_k(v)$, and the set of all prefixes of v is $\mathrm{Pref}(v)$. For suffixes, we define $\mathrm{suf}_k(v)$ and $\mathrm{Suf}(v)$ in a similar way. A factor (resp. prefix, suffix) u of a word v is called *proper* if $u \neq v$ and $u \neq \varepsilon$. If $w = uv$, then $u^{-1}w = v$ is the *left quotient* of w by u. If u is not a prefix of w, then $u^{-1}w$ is undefined. Similarly we define the *right quotient* wu^{-1}.

Words can also be considered as mappings. A finite word $w = w_1 \cdots w_n$ where $w_i \in \mathcal{A}$, $1 \leq i \leq n$, is a function $w\colon \{1, 2, \ldots, n\} \to \mathcal{A}$ that maps integer i to the letter in the ith position in w, i.e., $w(i) = w_i$. Similarly, infinite words $w = w_1 w_2 w_3 \cdots$ can be seen as functions from positive integers $\mathbb{Z}_+ = \{1, 2, 3, \ldots\}$ into the alphabet $\mathcal{A}$.

A *language* is a subset of $\mathcal{A}^*$. For languages $L, K \subseteq \mathcal{A}^*$ and for a word $u \in \mathcal{A}^*$, we define

$$\begin{aligned}
LK &= \{uv \mid u \in L, v \in K\}, \\
L^+ &= \bigcup_{i \geq 1} L^i = \{u_1 u_2 \cdots u_n \mid n \geq 1, u_j \in L\}, \\
L^* &= L^+ \cup \{\varepsilon\}, \\
u^{-1}K &= \{v \mid uv \in K\}, \\
Ku^{-1} &= \{v \mid vu \in K\}, \\
L^{-1}K &= \bigcup_{u \in L} u^{-1}K = \{v \mid \exists\, u \in L\colon uv \in K\}.
\end{aligned}$$

As an example, let us consider the word $w = abbabaa$. The length of the word is seven and $\mathrm{Alph}(w) = \{a, b\}$. The reversal of w is $\overline{w} = aababba$ and the set of prefixes is $\mathrm{Pref}(w) = \{\varepsilon, a, ab, abb, abba, abbab, abbaba, abbabaa\}$. For instance, we have $\mathrm{pref}_3(w) = abb$ and $\mathrm{suf}_3(w) = baa$. The square of w is $w^2 = abbabaaabbabaa$. For $u = abbab$, the left quotient of w by u is defined and $u^{-1}w = aa$. Considering w as a function, we have, for instance, $w(4) = w_4 = a$.

2.2 Relations

A k-ary *relation* over the sets $X_1, \ldots, X_k$ is a subset of the cartesian product $X_1 \times X_2 \times \cdots \times X_k$. In this thesis, we consider only binary relations R on a set X, i.e., $R \subseteq X \times X$. We often write $x\,R\,y$ instead of $(x, y) \in R$. The restriction of R on $Y \subseteq X$, i.e., $R \cap (Y \times Y)$ is denoted by R_Y. A relation R on X is called an *equivalence relation* if it is *reflexive*, *symmetric*, and *transitive*. In other words, an equivalence relation R satisfies

(E1) $\forall\, x \in X:$ $\quad x\,R\,x$,
(E2) $\forall\, x, y \in X:$ $\quad x\,R\,y \implies y\,R\,x$,
(E3) $\forall\, x, y, z \in X:$ $\quad x\,R\,y,\ y\,R\,z \implies x\,R\,z$.

If $R \subseteq X \times X$ is an equivalence relation and $x\,R\,y$, we say that x and y are *R-equivalent* or shortly *equivalent*. The set of all elements equivalent to x is called the *equivalence class* of x and it is denoted by $[x]_R$. These classes form a partition of X. The element x is called a *representative* of the equivalence class $[x]_R$.

A relation satisfying conditions (E1) and (E2) is called a *compatibility relation*. If $x\,R\,y$ and R is a compatibility relation, we say that x is *R-compatible* with y, or succinctly, that x and y are *compatible*. Let 2^X denote the *power set* of X, that is, the family of all subsets of X including the empty set $\emptyset$ and X itself. For a compatibility relation R on X, let the corresponding function $R\colon 2^X \to 2^X$ be defined by

$$R(Y) = \{x \in X \mid \exists\, y \in Y:\ y\,R\,x\}. \tag{2.1}$$

Note that since R is symmetric, we have $R(Y) = \{x \in X \mid \exists\, y \in Y:\ x\,R\,y\}$. If Y contains only one element $y \in X$, by abuse of notation, we denote $R(Y)$ by $R(y)$.

A classical example of an equivalence relation (and a compatibility relation) is the congruence of integers modulo n. As another example of a compatibility relation on the integers $\mathbb{Z}$, let us consider the relation D_2:

$$x\,D_2\,y \iff |x - y| \leq 2.$$

This relation is clearly reflexive and symmetric but it is not an equivalence relation, since it is not transitive. For instance, $-2\, D_2\, 0$ and $0\, D_2\, 1$ whereas -2 and 1 are not related. For example, $D_2(1) = \{-1, 0, 1, 2, 3\}$.

The *identity relation* on a set X is defined by $\iota_X = \{(x, x) \mid x \in X\}$. Evidently, ι_X is an equivalence relation, and therefore also a compatibility relation, on X. The *universal similarity relation* on words over $\mathcal{A}$ is defined by $\Omega = \Omega_{\mathcal{A}^*} = \{(x, y) \in \mathcal{A}^* \times \mathcal{A}^* \mid |x| = |y|\}$. Also this relation is a compatibility relation. The subscripts X and $\mathcal{A}^*$ are often omitted when they are clear from the context.

2.3 Codes

A sequence $x_1, \ldots, x_n$ is a *factorization* of a word w if $w = x_1 \cdots x_n$. If the words x_i, $1 \leq i \leq n$, belong to a set $X \subseteq \mathcal{A}^*$, then the factorization is called an *X-factorization*. The following figure illustrates graphically a word with two distinct factorizations $x_1, \ldots, x_m$ and $y_1, \ldots, y_n$.

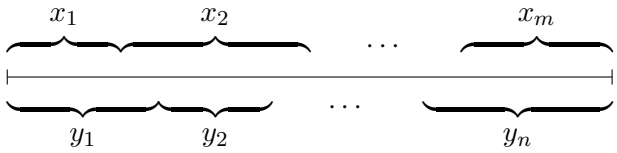


Figure 2.1: Two distinct factorizations of a word.

A set $X \subseteq \mathcal{A}^*$ is said to *generate* the subsemigroup X^+ of $\mathcal{A}^+$ (resp. the submonoid X^* of $\mathcal{A}^*$). Each word in this subsemigroup (resp. submonoid) has at least one X-factorization. If such a factorization is always unique, X is called a *code*. In other words, $X \subseteq \mathcal{A}^*$ is a code if it satisfies the following *decoding condition*: For all $n, m \geq 1$ and $x_1, \ldots, x_m, y_1, \ldots, y_n \in X$,

$$x_1 \cdots x_m = y_1 \cdots y_n \implies n = m \text{ and } x_i = y_i \text{ for } i = 1, 2, \ldots, m. \qquad (2.2)$$

More precisely, such a code is a *variable length* code as opposed to an *error correcting* code [5].

Codes are closely related to so called *free semigroups and monoids*. In the sequel we will consider only the monoid case. Corresponding definitions for semigroups are obvious. Recall that a monoid M is free if it has a subset $B \subseteq M$ such that every element of M has a unique representation as a product of elements of B. Such a set B is called a *base* of M.

Let X be a *generating set* of a monoid M, i.e., $M = X^*$. If no proper subset of X is a generating set of M, then X is called *minimal*. For word monoids $M \subseteq \mathcal{A}^*$, there exists a unique minimal generating set; see [8].

Theorem 2.1. *For a word monoid $M \subseteq \mathcal{A}^*$, the set $(M \setminus \{\varepsilon\}) \setminus (M \setminus \{\varepsilon\})^2$ is the unique minimal generating set.*

Note that the set $(M\setminus\{\varepsilon\})\setminus(M\setminus\{\varepsilon\})^2$ consists of all *indecomposable* elements of M. In other words, if $x \in (M\setminus\{\varepsilon\})\setminus(M\setminus\{\varepsilon\})^2$, then it cannot be expressed in a form $x = yz$ where y and z belong to M and they are both nonempty. The minimal generating set of a free monoid is a code and, conversely, a code is the minimal generating set of the free monoid generated by the code [8].

Theorem 2.2. *Let $X \subseteq \mathcal{A}^*$. The following conditions are equivalent:*

(i) X is a code,

(ii) X is a base of the free monoid X^*,

(iii) X^* is free and X is its minimal generating set.

For convenience, the minimal generating set of any word monoid M, even a non-free one, will be called the base of M from now on. We denote it by Base(M).

As an example, consider the set $X = \{ab, b, babb\}$ over the binary alphabet $\mathcal{A} = \{a, b\}$. Since $babb = (b)(ab)(b)$ has two different X-factorizations, X does not satisfy the decoding condition. Although X is not a code, X^* is a free monoid. Namely, Base$(X^*) = \{ab, b\}$ can be easily seen to be a code by reading any word in X^* from left to right.

2.4 Hulls

If a submonoid M of $\mathcal{A}^*$ is not free, there exists at least one element $w \in M$ which has two distinct Base(M)-factorizations. This is equivalent to saying that w has a double factorization *in* M. A *unique factorization extension* of M is a monoid $\overline{M} \subseteq \mathcal{A}^*$ such that it contains M and no element of M has a double factorization in $\overline{M}$. If every element of $\overline{M}$ also has a unique Base$(\overline{M})$-factorization, then the extension is free. Note that such extensions always exist. Namely, the word monoid $\mathcal{A}^*$ is a free (and unique factorization extension) of any of its submonoids. Hulls are the smallest among the unique factorization extensions and free extension described above. Their existence is based on the following results.

Free submonoids of $\mathcal{A}^*$ can be characterized using *stability*. A monoid $M \subseteq \mathcal{A}^*$ is called *stable* if

$$u, v, uw, wv \in M \implies w \in M. \tag{2.3}$$

This is illustrated in Figure 2.2.

The following powerful characterization of stable monoids is called *Schützenberger's criterion*; see [8].

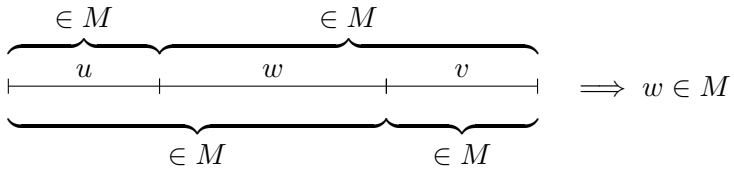


Figure 2.2: Stability of a monoid M.

Theorem 2.3 (Schützenberger's criterion)**.** *A submonoid of $\mathcal{A}^*$ is free if and only if it is stable.*

As an easy consequence we obtain the so called *Tilson's result* [78].

Corollary 2.1 (Tilson's result)**.** *Any intersection of free submonoids of $\mathcal{A}^*$ is a free monoid.*

This enables us to define the *free hull* of a monoid $M \subseteq \mathcal{A}^*$ as the intersection of all free extensions of M. It is the smallest free submonoid of $\mathcal{A}^*$ which contains M. By modifying the above stability condition, we could as well show the existence of the smallest submonoid of $\mathcal{A}^*$, where any element of M has a unique factorization; see [47]. This submonoid is called the *unique factorization hull* of M.

The famous *defect effect*, often considered to be folklore, says that if a set of n words satisfies a nontrivial relation, then these words can be expressed simultaneously as products of at most $n-1$ words. This effect is used in many different connections [9, 34, 55, 73]. Actually, there exist several defect theorems depending on the restrictions that are put to the $n-1$ words [48]. One formulation is presented using hulls as follows.

Theorem 2.4 (Defect theorem)**.** *Let $X \subseteq \mathcal{A}^+$ be a finite set and let B be the base of the free hull of X. Then $|B| \leq |X|$, and the equality holds if and only if X is a code.*

For example, consider the set $X = \{ab, abba, baab\}$ consisting of indecomposable elements of X^*. Hence, $\mathrm{Base}(X^*) = X$. A nontrivial relation $(ab)(baab) = (abba)(ab)$ shows that X^* is not free. Let $\overline{M}$ be the free hull of X^*. By Theorem 2.3, the monoid $\overline{M}$ must be stable. Therefore, substituting $u = v = ab$ and $w = ba$ in (2.3), we conclude that ba must belong to $\overline{M}$. Since the minimal generating set of $M = \{ab, abba, ba, baab\}^* \subseteq \overline{M}$ is $B = \{ab, ba\}$ and it is clearly a code, we conclude that the free hull of X^* is $M = \overline{M} = \{ab, ba\}^*$. Here $|B| = 2 < 3 = |X|$ demonstrates the defect effect.

2.5 Periods

Let $w = w_1 \cdots w_n$, $w_i \in \mathcal{A}$ for all i, be a word over the alphabet $\mathcal{A}$. An integer $p \geq 1$ is a *period* of w if, for all $i, j \in \{1, 2, \ldots, n\}$, we have

$$i \equiv j \pmod{p} \implies w_i = w_j.$$

In this case, the word w is called *p-periodic*. The smallest integer which is a period of w is called *the (minimal) period* of w and is denoted by $\pi(w)$, or shortly by π if the word w is clear from the context. Note that a word has a period p if and only if $w_i = w_{i+p}$ for each $1 \leq i \leq |w| - p$. This means that the word is a rational power of $\mathrm{pref}_p(w)$. Figure 2.3 represents a p-periodic word w.

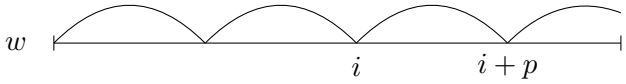


Figure 2.3: A word w with period p.

In this thesis we consider *interaction properties of periods*. More precisely, we study phenomena in which two different periods occurring simultaneously in a finite word imply the existence of a third period. The fundamental theorem of this type is the *theorem of Fine and Wilf*, originally proved in connection with real continuous functions [36]. It is well-known in combinatorics on words in the following form:

Theorem 2.5 (Theorem of Fine and Wilf)**.** *If a word x has periods p and q, and is of length at least $p + q - \gcd(p, q)$, then x also has a period $\gcd(p, q)$.*

Here $\gcd(p, q)$ denotes the greatest common divisor of integers p and q as usual. For a simple proof, see [41]. For more than two periods, this theorem was generalized in [26, 28, 50, 77].

For example, the word $w = aabaabaaabaabaa$ has periods $7, 10, 13, 14$ and, trivially, all integers $n \geq |w| = 15$. Hence, the minimal period is $\pi(w) = 7$. Note that $|w| = 15 = 7 + 10 - \gcd(7, 10) - 1$. From the theorem of Fine and Wilf it follows that there are no longer binary words with periods 7 and 10. Such words must be unary, since $\gcd(7, 10) = 1$.

An infinite word w has period p if, for all positive integers i and j, we have

$$i \equiv j \pmod{p} \implies w_i = w_j.$$

If this is the case then the word can be written in the form

$$w = v^\omega = vvv \cdots,$$

where v is a finite word of length p. An *ultimately periodic word* is a word of the form uv^ω, where the prefix u is finite and the infinite suffix v^ω is $|v|$-periodic.

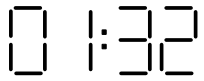
Periodicity has many applications and connections to other fields of mathematics. As an example, we mention the subword complexity $p_w(n) = |F_n(w)|$ of an infinite word w which is used in number theory. Namely, if w is an expansion of a number in some integer base b, w having a "low" subword complexity implies that the corresponding number is either rational or transcendental. The famous Morse–Hedlund theorem states that $p_w(n)$ is bounded if and only if w is ultimately periodic [61]. Clearly, ultimately periodic expansions correspond to rational numbers, whereas it was proved in [35] that numbers with Sturmian expansions, i.e., infinite words w with $p_w(n) = n + 1$ are transcendental. For more on these results, see [52] and especially the recent advances [1,2].

Chapter 3

Similarity Relations

In this thesis we investigate to what extent some classical theorems in combinatorics on words are tenable when we consider words up to a similarity relation, which is a relation on words induced by a compatibility relation on letters. In other words, we replace the identity relation ι by a similarity relation R and, instead of a word w, we consider $R(w)$, i.e., all words related to w. Our goal is to search for phenomena such that, despite identifying similar words and loosening assumptions on the initial data, we can deduce in some sense good and correct information.

As an everyday example, let us consider the display of a digital clock with at most one broken LED. The possible interpretations of the following display are 01 : 32, 81 : 32, 07 : 32 and 01 : 92.



In spite of the broken LED, we may conclude the right time quite reliably, at least with some extra information. For instance, if it is evening and the digital clock says 01 : 32, we may infer that the correct time is actually 07 : 32. In this example, the correct interpretation is similar to the displayed one. Here the similarity relation is a one LED difference in one of the numbers of the display. Note that this relation is not transitive. For example, there is a difference of one LED between the displays of numbers 5 and 6 and also 6 and 8, but you cannot confuse 5 and 8 with each other if only one malfunctioning LED is possible.

In this chapter we first formally define the concept of a similarity relation. Motivation for our research comes mainly from two distinct disciplines, namely from theoretical computer science and molecular biology. Applications of similarity relations in theoretical computer science are discussed in Section 3.2. Originally, similarity relations were introduced in [44] to generalize the notion of a partial word as presented by Berstel and Boasson in

1999; see [6]. A brief overview on partial words is given in Section 3.2.1. In Section 3.2.2 we give a description of the use of similarity relations in modeling error correcting capabilities in generalized variable length codes. These codes are considered in more detail in Chapter 5. Finally, in Section 3.3 we show how similarity relations are connected to the study of biological sequences such as DNA, RNA and proteins.

3.1 Definition

Let us begin by giving the formal definition of a similarity relation. We note that algebraic operations of the direct product are defined componentwise. Hence, the associative operation of the monoid $\mathcal{A}^* \times \mathcal{A}^*$ is naturally the componentwise catenation.

Definition 3.1. Let $\mathcal{A}$ be an alphabet. A relation R is called a *similarity relation* on words over $\mathcal{A}$ if it is a submonoid of $\mathcal{A}^* \times \mathcal{A}^*$ generated by a compatibility relation $R_\mathcal{A} \subseteq \mathcal{A} \times \mathcal{A}$ on letters. The relation $R_\mathcal{A}$ is called the *generating relation* of R. Words u and v satisfying $u\,R\,v$ are said to be *similar* or, more precisely, *R-similar*.

Hence, a similarity relation is a "letter-to-letter" compatibility relation on words of equal length. That is to say, for words $u = u_1 \cdots u_m$ and $v = v_1 \cdots v_n$, where $u_i, v_j \in \mathcal{A}$, a similarity relation R satisfies

$$u_1 \cdots u_m \, R \, v_1 \cdots v_n \iff m = n \text{ and } u_i \, R \, v_i \text{ for all } i = 1, 2, \ldots, m. \quad (3.1)$$

Since a similarity relation R is induced by its restriction $R_\mathcal{A}$ on letters, it can be presented by listing all pairs $\{a, b\}$ $(a \neq b)$ such that $(a, b) \in R_\mathcal{A}$. We use the notation

$$R = \langle r_1, \ldots, r_n \rangle,$$

where $r_i = (a_i, b_i) \in \mathcal{A} \times \mathcal{A}$ for $i = 1, 2, \ldots, n$, to denote that R is the similarity relation generated by the symmetric closure of $\iota_\mathcal{A} \cup \{r_1, \ldots, r_n\}$.

Example 3.1. Let $\mathcal{A} = \{a, b\}$ and set $R = \langle (a, b) \rangle$. Then

$$R_\mathcal{A} = \{(a, a), (b, b), (a, b), (b, a)\}.$$

Hence, the relation R makes all words with equal length similar to each other. On the other hand, let us consider the ternary alphabet $\mathcal{B} = \{a, b, c\}$ and the set $S = \langle (a, b) \rangle$. Then

$$S_\mathcal{B} = \{(a, a), (b, b), (c, c), (a, b), (b, a)\}$$

and, for example, $abba\,S\,baab$ holds but, for instance, words abc and cac are not S-similar.

A similarity relation R is clearly a compatibility relation, since the generating relation is reflexive and symmetric. Hence, it is justified to use expressions R-compatible and R-similar side by side. *From now on the relations on words considered in this thesis are supposed to be similarity relations induced by some compatibility relation on letters.*

Let us make a few observations on the basic properties of similarity relations. First of all, a similarity relation does not need to be transitive. As an example, commonly used throughout the thesis, we mention the relation $R = \langle (a,b)(b,c) \rangle$. Here the letter b is sort of a universal letter, i.e., related to all other letters. However, a is not R-compatible with c although $a\,R\,b$ and $b\,R\,c$. Secondly, from Equation (3.1) it follows that every similarity relation R satisfies the following two fundamental properties.

$$\begin{array}{lllll} \text{multiplicativity:} & u\,R\,v, u'\,R\,v' & \Longrightarrow & uu'\,R\,vv', \\ \text{simplifiability:} & uu'\,R\,vv', |u| = |v| & \Longrightarrow & u\,R\,v,\ u'Rv'. \end{array}$$

The proof of the following theorem is based on these features. Recall from Section 2.2 that $R(X) = \{w \in \mathcal{A}^* \mid \exists\, x \in X : x\,R\,w\}$ for a similarity relation R on $\mathcal{A}^*$. The function $R(X)$ is multiplicative in the following sense.

Theorem 3.1. *Let R be a similarity relation on $\mathcal{A}^*$. Then $R(X)R(Y) = R(XY)$ for all $X, Y \subseteq \mathcal{A}^*$. Especially, $R(X)^* = R(X^*)$ for all $X \subseteq \mathcal{A}^*$.*

Proof. Suppose that w belongs to $R(X)R(Y)$. Then there exist words $u \in R(X)$ and $v \in R(Y)$ such that $w = uv$. Hence, by (2.1), there exist $x \in X$ and $y \in Y$ such that $x\,R\,u$ and $y\,R\,v$. By the multiplicativity of the relation R we have $xy\,R\,uv$, and thus $w \in R(XY)$.

Conversely, let w belong to $R(XY)$. Then there exist words $x \in X$ and $y \in Y$ such that $xy\,R\,w$. By the definition of a similarity relation this means that $|w| = |x|+|y|$. Thus, w can be factored into two parts u and v satisfying $w = uv$ with $|u| = |x|$ and $|v| = |y|$. By the simplifiability of R, we have $x\,R\,u$ and $y\,R\,v$. Hence, $w = uv \in R(X)R(Y)$.

By induction, we see that $R(X)^n = R(X^n)$ for all $n \geq 1$. Furthermore, $R(X)^0 = \varepsilon = R(\varepsilon) = R(X^0)$. Thus, the second claim also follows. □

3.2 Similarity Relations in Theoretical Computer Science

It is a natural tendency in sciences to deepen knowledge by studying more and more complex systems and extensions of simplified models. As evidence of this we mention some developments in theoretical computer science in the recent decades. For example, Cobham's theorem of recognizable sets of integers was generalized for higher dimensions [64, 70]. Also the theorem of

Fine and Wilf with two periods was extended for an arbitrary number of simultaneous periods [26, 28, 50, 77]. For partial words with one, two, three or arbitrary number of holes, the Fine and Wilf theorem was generalized in a series of papers [6, 12, 22]. Finally we mention that Dejean's conjecture on repetition thresholds has been proved for larger and larger alphabets [25, 62].

Sometimes generalizations arise from a practical background. For instance, consider the so called *non-standard string matching* problem where a pattern is matched with the text not identically but up to some relation [65]. In other words, we are given a general many-to-many matching relation between symbols and a pattern p of length m. We seek those positions in a text t of length n (typically much greater than m) at which the pattern matches under the relation. Actually, this is one of the applications of similarity relations. They can be used, for instance, in *string matching with "don't cares"* and in *distance matching problems*. In string matching problem with "don't cares", introduced in [37], there exists a special symbol which matches all other symbols. In distance matching problems, a distance function d is defined between each pair of symbols. Symbols a and b match if and only if $d(a, b) \leq k$ for some specified constant k. In both these cases the relations on letters are compatibility relations inducing similarity relations. Thus, a pattern which matches the text is, using our terminology, similar to the text. We note that in [65] string matching with relations other than compatibility relations were also considered. In the following subsections we give two other applications showing how similarity relations can be used to generalize well-known concepts.

3.2.1 Partial Words

We consider partial words as introduced by Berstel and Boasson in [6]. A *partial word* of length n over an alphabet $\mathcal{A}$ is a partial function

$$w\colon \{1, 2, \ldots, n\} \to \mathcal{A}.$$

The domain $D(w)$ of w is the set of positions $p \in \{1, 2, \ldots, n\}$ such that $w(p)$ is defined. The set $H(w) = \{1, 2, \ldots, n\} \setminus D(w)$ is the set of *holes* of w. To each partial word we may associate a total word $w_\diamond$ over the extended alphabet $\mathcal{A}_\diamond = \mathcal{A} \cup \{\diamond\}$. This *companion* of w is defined by

$$w_\diamond(p) \quad = \quad \begin{cases} w(p) & \text{if } p \in D(w), \\ \diamond & \text{if } p \in H(w). \end{cases}$$

Thus, the holes are marked with the "do not know" symbol $\diamond$. Clearly, partial words are in one-to-one correspondence with words over $\mathcal{A}_\diamond$.

The *compatibility relation of partial words* is defined as follows. Let x and y be two partial words of equal length. The word y *contains* x if

$D(x) \subseteq D(y)$ and $x(k) = y(k)$ for all k in $D(x)$. Two partial words x and y are said to be *compatible* if there exists a partial word z such that z contains both x and y, in which case we write $x \uparrow y$. Intuitively, the compatibility of two words is easily perceived by placing one word above the other and comparing the letters in corresponding positions. For example, we see that the following partial words are compatible and a total word containing them both is "knowledge".

k	n	$\diamond$	w	l	$\diamond$	d	g	e
$\diamond$	n	o	w	$\diamond$	$\diamond$	d	g	$\diamond$
k	n	o	w	l	e	d	g	e

Combinatorics on partial words has been widely studied in recent years. In the early papers, the main focus was on the interaction properties of periods of partial words; see [6,10,12,19,22,38,71,72]. Other topics considered in the scientific literature are, for example, primitivity [13, 15], codes and orderings [11], the critical factorization theorem [20], conjugacy [23], equations [16] and unavoidable sets [17]. The set of periods of partial words were investigated in the style of Guibas and Odlyzko [40] in [18,21]. Lischke studied punctured languages, i.e., languages of partial words in [56].

From our viewpoint, the compatibility relation on partial words is a special case of similarity relations. We next show in detail how partial words with compatibility relation $\uparrow$ can be seen as words over the alphabet $\mathcal{A}_\diamond$ with the similarity relation

$$R_\uparrow = \langle \{(\diamond, a) \mid a \in \mathcal{A}\} \rangle. \tag{3.2}$$

Namely, consider two compatible partial words x and y. Let z be a partial word which contains both x and y. Suppose that their companions are $x_\diamond = a_1 \cdots a_n$, $y_\diamond = b_1 \cdots b_n$ and $z_\diamond = c_1 \cdots c_n$, where $a_j, b_k, c_l \in \mathcal{A}_\diamond$ for all j, k and l. According to the definition of compatible partial words, we have five possibilities for each position $i \in \{1, 2, \ldots, n\}$:

$$\begin{array}{ll}
\text{(i)} & c_i = \diamond,\ a_i = b_i = \diamond \\
\text{(ii)} & c_i \neq \diamond,\ a_i = b_i = \diamond \\
\text{(iii)} & c_i \neq \diamond,\ a_i = \diamond,\ b_i = c_i \\
\text{(iv)} & c_i \neq \diamond,\ b_i = \diamond,\ a_i = c_i \\
\text{(v)} & c_i \neq \diamond,\ a_i = b_i = c_i.
\end{array}$$

We see that in each case $a_i \, R_\uparrow \, b_i$, and thus $x_\diamond \, R_\uparrow \, y_\diamond$. On the other hand, for $R_\uparrow$-similar words $x_\diamond = a_1 \cdots a_n$ and $y_\diamond = b_1 \cdots b_n$ we can always find a word $z_\diamond = c_1 \cdots c_n$ such that the corresponding partial words x and y are contained in z and therefore $x \uparrow y$. We simply choose the letter c_i in such a way that it corresponds to one of the cases (i)–(v) above. Thus, partial words are equivalent to words over the alphabet $\mathcal{A}_\diamond$ with a specific relation $R_\uparrow$ and all results concerning similarity relations can also be applied for the compatibility relation of partial words.

3.2.2 Coding

Standard variable length codes were introduced in Section 2.3. In this section our approach is to strengthen the coding properties of such codes by requiring that two "nearly similar" words have the same, or at least "similar," factorizations. This is attained by introducing *relational codes*. The similarity of two words is described by using similarity relations. Generalizing codes to relational codes enables us to model situations where some of the letters in a message are changed to related letters, but the message can still be factorized, in other words, decoded in a proper manner. Thus, these codes possess some error correction capabilities.

The coding–decoding process of an (R, S)-code is described in Figure 3.1. A coded message is a word obtained by catenating code words. When the coded message is sent to a recipient through a channel, there may occur some errors. The similarity relation R illustrates possible changes in the message. It will be called an *alteration relation*. The decoding of the received message means factorizing the message into words which are R-similar to the words in an (R, S)-code. Note that one factor may be compatible with several code words and hence the decoded message is ambiguous. However, the factorization into R-similar code words obtained this way is S-similar with the original message. Hence, the relation S describes the correctness of the decoded message and it will be therefore called a *fidelity relation*.

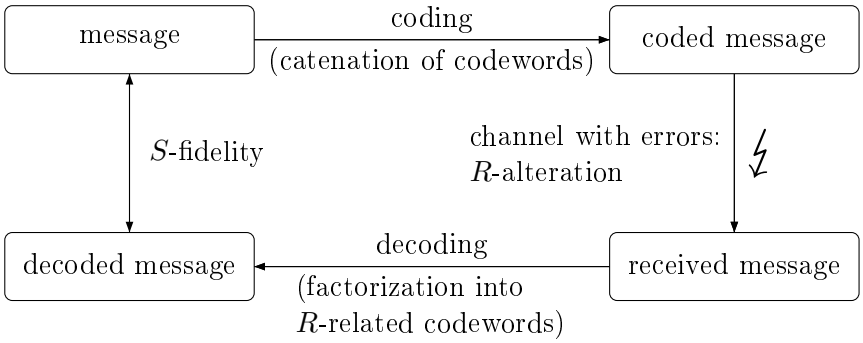


Figure 3.1: Coding and decoding of relational codes

The alteration relation can be thought of as modelling errors or differences in a coded message. The fidelity relation can be thought of as describing how well messages can be decoded. Clearly, two natural special cases occur. If S is the identity relation ι, then the decoded message must be equal with the original message. Hence, all code words are necessarily non-R-similar. Another case consists of relational codes where $S = R$. This means that the whole decoding is done up to relation R. Changes in the

channel do not affect the lengths of factors in a decoded message. If coded messages are R-similar, so are their factorizations.

We will show in detail in Section 4.1 that alteration relations and fidelity relations enable us to analyze coding properties of relational codes. For this purpose, we define the size of a similarity relation. Hence, we may study the error capacity of a relational code X by measuring maximal alteration relations. On the other hand, we may investigate the coding precision of X by finding minimal fidelity relations.

3.3 Similarity Relations in Biology

Motivation for the research of partial words comes partly from the study of biological sequences such as DNA (deoxyribonucleic acid), RNA (ribonucleic acid) and proteins [11]. The analysis of these sequences plays a central role in computational molecular biology where techniques of mathematics, statistics, computer science and biochemistry are used to solve biological problems. Since partial words are a special case of words with a similarity relation, it is evident that bioinformatics gives a suitable background for applications of similarity relations as well. Here we briefly describe the molecule biological setting and some basic operations where relations can be used as theoretical models.

3.3.1 Genes and Protein Synthesis

Cells are the building blocks of all living organisms from unicellular bacteria to plants and animals consisting of a few hundred to many trillions of cells. The structures and functions of cells are prescribed by genes via protein synthesis. A gene is a fraction of the DNA strand contained in a chromosome in a cell nucleus. This DNA strand is a polymer consisting of monomers called nucleotides which are joined together to form a single enormously long chain. For example, the longest DNA polymers among the 23 different human chromosomes are made up of hundreds of millions of nucleotides. There are four types of nucleotides, each made of a sugar molecule (deoxyribose sugar), a phosphate group and a base. The nucleotides differ only in their bases giving the nucleotides their names: A (*adenine*), C (*cytosine*), G (*guanine*) and T (*thymine*). Nucleotides form DNA strands via connections between the sugar and phosphate groups. The bases project from the side of this *sugar-phosphate backbone*. Two strands of DNA bind to each other forming *base pairs* between A and T and between C and G. This double strand twists in a helical fashion creating a so called *DNA double helix*.

Proteins are macromolecules made of amino acids playing an essential role in living organisms by participating in all central processes within cells.

Name	Abbreviation	Name	Abbreviation
Alanine	Ala	Leucine	Leu
Arginine	Arg	Lysine	Lys
Asparagine	Asn	Methionine	Met
Aspartic acid	Asp	Phenylalanine	Phe
Cysteine	Cys	Proline	Pro
Glutamine	Gln	Serine	Ser
Glutamic acid	Glu	Threonine	Thr
Glycine	Gly	Tryptophan	Trp
Histidine	His	Tyrosine	Tyr
Isoleucine	Ile	Valine	Val

Table 3.1: The 20 amino acids and their three letter abbreviations

The word protein comes from a Greek word "$\pi\rho\acute{\omega}\tau\alpha$" (prota) meaning "of primary importance." A gene prescribes the construction of proteins by specifying the sequence of amino acids in the protein that the gene encodes. The 20 amino acids and their three letter abbreviations are given in Table 3.1. The length of a gene depends upon the length of the amino acid chain it encodes. On average, the length is approximately 1000 base pairs. Individual genes are separated from one another along spacers in the DNA strand. Typically spacers, sometimes referred to as junk DNA, make up 95% of a DNA molecule. The building instructions for protein synthesis are coded in genes using triplets of nucleotides in one of the two DNA strands of the double helix. The trinucleotides are called *codons*. The protein synthesis can be divided into two main steps: transcription and translation.

In *transcription* the DNA strand serves as a template to assemble a *messenger RNA* molecule (mRNA) which is transported to the cytoplasm for translation. An RNA molecule is like a DNA molecule except that deoxyribose is replaced by a ribose sugar and the four bases of RNA are A, C, G and U, where U stands for *uracil*. The mRNA is built using *base-pairing rules*: whenever a T occurs in the DNA strand of a gene, an A is added to the mRNA molecule. Similarly, G is matched with C, and C with G. However, an A in a gene designates U in mRNA.

In the cytoplasm, the messenger RNA binds to particles called ribosomes for *translation*. Ribosomes read the coding of mRNA by moving from one codon to the next one, interpreting the present codon as one of the 20 amino acids and joining that amino acid to the growing amino acid chain (protein).

First position	Second position				Third position
	A	C	G	U	
A	Lys	Thr	Arg	Ile	A
	Asn	Thr	Ser	Ile	C
	Lys	Thr	Arg	Met	G
	Asn	Thr	Ser	Ile	U
C	Gln	Pro	Arg	Leu	A
	His	Pro	Arg	Leu	C
	Gln	Pro	Arg	Leu	G
	His	Pro	Arg	Leu	U
G	Glu	Ala	Gly	Val	A
	Asp	Ala	Gly	Val	C
	Glu	Ala	Gly	Val	G
	Asp	Ala	Gly	Val	U
U	Stop	Ser	Stop	Leu	A
	Tyr	Ser	Cys	Phe	C
	Stop	Ser	Trp	Leu	G
	Tyr	Ser	Cys	Phe	U

Table 3.2: The genetic code table (5' to 3' direction)

In the sugar-phosphate backbone a phosphate connects the number 5 carbon of one sugar molecule to the number 3 carbon of the next sugar. Hence, there are two directions to read the DNA strand. In this presentation the genetic code is always notated from the 5 carbon end to the 3 carbon end, i.e., in *5' to 3' direction*. Thus, every mRNA begins with trinucleotide AUG implying that all amino acid chains begin with methionine. Then the following amino acids of the protein are coded according to Table 3.2. Finally, at the end of the mRNA, the ribosome translates UGA, UAA or UAG as a "stop" symbol and releases the finished amino acid chain.

We have now introduced three kinds of sequences occurring in computational molecular biology:

1. DNA sequences from a few million to a few billion letters over the four letter alphabet {A,C,G,T},
2. RNA sequences with around thousand letters over the four letter alphabet {A,C,G,U} and
3. protein sequences with a few hundred letters in the 20-letter alphabet of amino acids.

For more on these sequences and gene function, see, e.g., [33].

As an example of the use of similarity relations within the framework of protein synthesis, let us consider codons as letters forming a 64-letter alphabet. We define that those codons which encode the same amino acid are similar. For instance, both GAA and GAG correspond to glutamic acid (Glu) whereas UUA, CUA and CUU are all codes for leucine (Leu). Thus, the following two codon strings stand for the same amino acid sequence (Glu-Leu-Leu) despite the mutations (underlined changes) in the genetic code and therefore the sequences are similar.

$$\underbrace{\mathrm{GAA}}_{\mathrm{Glu}}\underbrace{\mathrm{UUA}}_{\mathrm{Leu}}\underbrace{\mathrm{CUU}}_{\mathrm{Leu}} \overset{mutation}{\rightsquigarrow} \underbrace{\mathrm{GA\underline{G}}}_{\mathrm{Glu}}\underbrace{\mathrm{\underline{C}UA}}_{\mathrm{Leu}}\underbrace{\mathrm{CU\underline{A}}}_{\mathrm{Leu}}$$

3.3.2 Sequence Alignment

One of basic operations in bioinformatics, serving as a basis for more complex manipulations, is *sequence comparison* or *sequence alignment*. It is a procedure of comparing two or more sequences by searching for character patterns that are in same order in the sequences. This aligning can be done by writing the sequences into rows above each other in such a way that identical or similar characters are placed in the same column and nonidentical characters are either placed in the same column as a mismatch or opposite a gap in the other sequence. In an optimal alignment, nonidentical characters and gaps are placed to obtain as many matches as possible. In practice, biological comparison tasks require the alignment of lengthy, highly variable or extremely numerous sequences that cannot be aligned solely by human effort. Hence, a variety of computational algorithms have been applied to the sequence alignment problem. For more on these methods, see [63].

Sequence comparison is useful for discovering functional, structural or evolutionary information in biological sequences. Sequences that are very much alike may, for example, code the same protein in the case of DNA molecules or they may have the same kind of three-dimensional structure in the case of proteins. Furthermore, if two sequences from different organisms are similar, they may be descended from a common ancestor sequence during evolution. In other words, mismatches and gaps in an alignment can be interpreted as *mutations*, i.e., permanent changes in the DNA. *Point mutations* (mismatches) change only one base pair of the DNA. Thus, they affect only the amino acid coded by the mutated codon. There are two types of these base mutations. *Transition mutations* involve replacing a purine (A or G) with another purine (G or A) and its partnering pyrimidine (T or C) with another pyrimidine (C or T). If a purine is replaced by a pyrimidine or vice versa, the mutation is called a *transversion*; see Figure 3.2. In *frame shifts* one or a few bases are inserted or deleted. Hence, these mutations affect all

of the amino acids following the position of the inserted or deleted base in the protein coding.

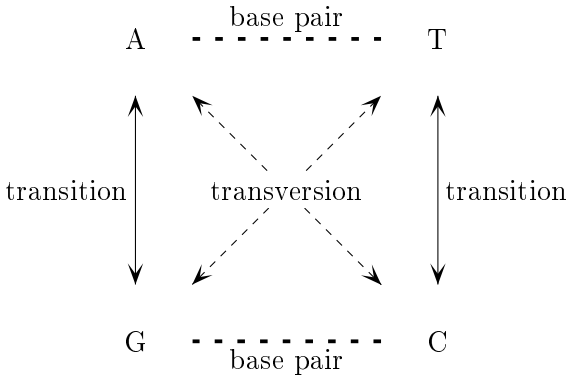


Figure 3.2: Transition and transversion mutations

It is a well-known feature of genomic sequences that transition mutations occur relatively more often than transversions. This fact has been successfully used in the *transition-constrained seed* model for sequence alignment [54]. This heuristic algorithm focuses only on those regions of the aligned sequences which share a common structure called a *seed*. For example, if a seed is represented by #@_#@, the algorithm searches for those positions of the sequences where the factors of length five have matches in the first and fourth letters, second and last letters may be equal or there is a transition mutation. Symbol _ is a "don't care" symbol meaning that the third letters do not need to match. Seeds with "don't care" symbols are called *spaced seeds*; see [60].

Set $\mathcal{A} = \{\text{A,C,G,T}\}$. Sequence alignment with the above mentioned mutations can be easily modeled using similarity relations on the extended alphabet $\mathcal{A}_\diamond$. Sequence similarity permitting transition mutations is expressed by

$$R_{\text{transit}} = \langle \{(\diamond, a) \mid a \in \mathcal{A}\}, (\text{A,G}), (\text{C,T}) \rangle.$$

As an example, let us consider the alignment of two DNA strands, TTAGATCTA and TCTGGATCA. We notice that these sequences look alike, e.g., they have a common factor GATC. Their similarity comes more evident when we align them in the following way.

T	$\diamond$	T	A	G	A	T	C	T	A
T	C	T	G	G	A	T	C	$\diamond$	A

These two words over $\mathcal{A}_\diamond$ are R_{transit}-similar. We interpret this relation as follows. The second sequence is obtained from the first one by two frame

shifts inserting a C and deleting a T and by one transition mutation replacing an A by a G. Note that the transition-constrained seed model is more specific in comparing sequences. Namely, the positions where transitions or mismatches occur are fixed.

3.3.3 DNA Sequencing

Another important method in molecular biology is *DNA sequencing*. It is used for determining the order of the nucleotide bases in a DNA strand. The genetic information obtained this way is useful in all basic research studying fundamental biological processes. In large-scale DNA sequencing the molecules to be analyzed are very long, but the sequencing methods allow only direct sequencing of relatively short, a few hundred nucleotides long, DNA fragments. In practice, a long DNA strand is cut into random pieces and multiple copies of these pieces are produced by cloning. This leads to a computational problem of assembling the pieces. In *fragment assembly* the goal is to reconstruct the DNA sequence (target) as completely and accurately as possible starting from the set of randomly selected pieces (sequenced fragments). Each pair of fragments is examined in order to determine if they share a common factor of the target, i.e., the end part of one fragment is similar to the beginning part of another. If this happens, the fragments are said to *overlap*. Then the target sequence is constructed by finding the maximal or optimal sets of consistent overlaps in order to merge the fragments.

Assembling fragments can be modeled using partial words or, more generally, using similarity relations. Like in sequence comparison, gaps are introduced for finding the best way of aligning similar bases. For example, let us try to assemble sequences TCGGA, TCCTA, GGATCC and ATTCC into a target sequence approximately of length 10. One way to do this allowing also transition mutations is the following:

T	C	G	G	A	◇	◇	◇	◇	◇	◇
◇	◇	◇	◇	◇	◇	T	C	C	T	A
◇	◇	G	G	A	T	C	C	◇	◇	◇
◇	◇	◇	◇	A	T	T	C	C	◇	◇

These R_{transit}-similar sequences imply a sequence TCGGATTCCTA, which is of length 11, close enough to the target length. Note that there is probably a sequencing error (T replaced by C) in the second to last letter of GGATCC, since T occurs in the corresponding position in two other overlapping fragments ATTCC and TCCTA. Of course, in real life applications there are several complications making the problem much harder than this small example. For more on fragment assembly, see [74].

Chapter 4

Relational Coding Properties

In this chapter we consider coding properties, unique factorizations and free monoids from the point of view of similarity relations. Our objective is to generalize the basic concepts and theorems on codes given in Sections 2.3 and 2.4. First, we formally introduce relational codes, more precisely, (R, S)-codes for arbitrary similarity relations R and S in Section 4.1. We show that relational coding properties can be algorithmically tested by the Sardinas–Patterson theorem in Section 4.1.1. These properties are further analyzed by finding minimal and maximal relations in the following subsection. In addition we show that the problem MAXIMAL RELATION is NP-complete. In this problem, we are given a finite set X and a similarity relation S, and the task is to determine whether or not X is an (R, S)-code for some compatibility relation R induced by at least k related pairs of letters.

Secondly, in Sections 4.2 and 4.3 we consider the free submonoids of $\mathcal{A}^*$ generated by (R, S)-codes. Our starting point is the (R, S)-unique factorization of words which is used for defining (R, S)-unique factorization extensions of sets $X \subseteq \mathcal{A}^*$ in Section 4.2.1. A relationally unique factorization extension of X is a monoid $M \subseteq \mathcal{A}^*$ such that all words in X have a relationally unique Base(M)-factorization. Section 4.2.2 is devoted to characterizing these extensions with stability conditions. A modified Schützenberger's criterion is proved. In Section 4.3 we show that, for a set $X \subseteq \mathcal{A}^*$, there exists under some restrictions the smallest (R, S)-unique factorization extension, the so called relational hull. The inner and the outer (R, S)-unique factorization hulls are defined. The existence of these hulls is a consequence of a generalized Tilson's result, which also implies the existence of the (R, S)-free hull of X. Procedures for finding hulls are given in Section 4.3.1. Finally, Section 4.4 is devoted to defect theorems with similarity relations. We prove a defect effect concerning inner (R, S)-unique factorizations hulls. Moreover, a cumulative defect theorem of (R, S)-free hulls is proved as a corollary and consequently a defect theorem for partial words follows.

Thirdly, we consider algorithmic aspects of the topics considered in this chapter. Our main tool is the generalized Spehner graph defined in Section 4.5. We show how it can be used for implementation of the Sardinas–Patterson algorithm as well as for the hull algorithms.

4.1 Relational Codes

In this section we generalize variable length codes using similarity relations and prove some basic properties of these relational codes.

Definition 4.1. Let R and S be two similarity relations on the monoid $\mathcal{A}^*$. A subset $X \subseteq \mathcal{A}^*$ is an (R,S)-*code* if for all $n, m \geq 1$ and $x_1, \ldots, x_m$, $y_1, \ldots, y_n \in X$, we have

$$x_1 \cdots x_m \, R \, y_1 \cdots y_n \implies n = m \text{ and } x_i \, S \, y_i \text{ for } i = 1, 2, \ldots, m. \tag{4.1}$$

The relation R will be called an *alteration relation* and the relation S will be called a *fidelity relation*.

As mentioned in Section 3.2.2, the alteration relation can be thought of as modelling errors or differences in a coded message. The fidelity relation can be thought of as describing how well messages can be decoded. If S is the identity relation ι, then an (R,S)-code is called a *strong R-code*, or shortly just an *R-code*. A strong R-code is always a set where the elements are pairwise non-similar, but the converse does not hold generally. An (R,R)-code is called a *weak R-code*. An (ι,ι)-code is simply called a *code*, since the definition coincides with the definition of a variable length code given in Section 2.3.

Consider a partial ordering of similarity relations on $\mathcal{A}^*$: $R_1 \subseteq R_2$ if, for all words u and v over $\mathcal{A}$, it follows from $u \, R_1 \, v$ that $u \, R_2 \, v$. The following theorem is illustrated in Figure 4.1.

Theorem 4.1. *(i) Let R_1, R_2 and S be similarity relations on $\mathcal{A}^*$ with $R_1 \subseteq R_2$. If X is an (R_2,S)-code, then X is an (R_1,S)-code. (ii) Let R, S_1 and S_2 be similarity relations on $\mathcal{A}^*$ and let $S_1 \subseteq S_2$. If X is an (R,S_1)-code, then X is an (R,S_2)-code.*

Proof. For (i), suppose that X is an (R_2,S)-code. Let $x_1, \ldots, x_m$ and $y_1, \ldots, y_n$ be words in X satisfying $x_1 \cdots x_m \, R_1 \, y_1 \cdots y_n$. This implies that also $x_1 \cdots x_m \, R_2 \, y_1 \cdots y_n$ by the assumption $R_1 \subseteq R_2$. Since X is an (R_2,S)-code, we have $n = m$ and $x_i \, S \, y_i$ for all $i = 1, 2, \ldots, m$. This proves the first claim.

For (ii), suppose that X is an (R,S_1)-code and $x_1, \ldots, x_m$, $y_1, \ldots, y_n \in X$ satisfy $x_1 \cdots x_m \, R \, y_1 \cdots y_n$. Then $n = m$ and $x_i \, S_1 \, y_i$ for all $i = 1, 2, \ldots, m$. Because $S_1 \subseteq S_2$, this implies that $x_i \, S_2 \, y_i$ for all $i = 1, 2, \ldots, m$ proving the second claim. □

When we consider unions and intersections of similarity relations the previous result implies the following corollary.

Corollary 4.1. *Let X be an (R_1, S_1)-code and let R_2 and S_2 be two similarity relations on $\mathcal{A}^*$. Then X is an $(R_1 \cap R_2, S_1 \cup S_2)$-code.*

Proof. Since $R_1 \cap R_2 \subseteq R_1$, X is an $(R_1 \cap R_2, S_1)$-code by Theorem 4.1(i). Since $S_1 \subseteq S_1 \cup S_2$, X is an $(R_1 \cap R_2, S_1 \cup S_2)$-code by Theorem 4.1(ii). □

For sets that are both (R, S_1)-codes and (R, S_2)-codes, the coding property can also be preserved when the fidelity relation is restricted to the intersection of S_1 and S_2 relations.

Theorem 4.2. *Let X be both an (R, S_1)-code and an (R, S_2)-code. Then it is also an $(R, S_1 \cap S_2)$-code.*

Proof. Assume that $x_1 \cdots x_m \, R \, y_1 \cdots y_n$ for $x_1, \ldots, x_m, \, y_1, \ldots, y_n \in X$. By the assumption, $n = m$ and $x_i \, S_j \, y_i$ for all $i = 1, 2, \ldots, m$ and for both $j = 1$ and $j = 2$. Thus, $x_i \, (S_1 \cap S_2) \, y_i$ for all $i = 1, 2, \ldots, m$, and, consequently, X is an $(R, S_1 \cap S_2)$-code. □

Note that X is not necessarily an $(R_1 \cup R_2, S)$-code even when it is both an (R_1, S)-code and an (R_2, S)-code.

Example 4.1. Let $X = \{ab, c\}$, $R_1 = \langle (a, c) \rangle$ and $R_2 = \langle (b, c) \rangle$. Clearly, X is both an (R_1, ι)-code and an (R_2, ι)-code. Now choose $R = R_1 \cup R_2 = \langle (a, c), (b, c) \rangle$. We have $ab \, R \, cc$, which implies that X is not an $(R_1 \cup R_2, \iota)$-code.

The next theorem gives a characterization of general (R, S)-codes in terms of weak R-codes.

Theorem 4.3. *A subset $X \subseteq \mathcal{A}^*$ is an (R, S)-code if and only if X is an (R, R)-code and $R_X \subseteq S_X$.*

Proof. Suppose first that X is an (R, S)-code. By Theorem 4.1(ii), X is an (R, Ω)-code, where Ω is the universal similarity relation. This simply means that if $x_1 \cdots x_m \, R \, y_1 \cdots y_n$ with $x_i, y_j \in X$, then $m = n$ and $|x_i| = |y_i|$ for all $i = 1, 2, \ldots, m$. Then, by the simplifiability of similarity relations, we have $x_i \, R \, y_i$ for all $i = 1, 2, \ldots, m$. Hence, X must be an (R, R)-code. Moreover, by choosing $m = n = 1$ in the definition of an (R, S)-code we see that $R_X \subseteq S_X$.

Conversely, let X be an (R, R)-code and $R_X \subseteq S_X$. Now consider words $x_1, \ldots, x_m, \, y_1, \ldots, y_n \in X$ satisfying $x_1 \cdots x_m \, R \, y_1 \cdots y_n$. Since X is an (R, R)-code, we have $n = m$ and $x_i \, R \, y_i$ for all $i = 1, 2, \ldots, m$. Hence, we also have $x_i \, S \, y_i$ for all $i = 1, 2, \ldots, m$ by the assumption $R_X \subseteq S_X$. □

Note that the roles of the relations R and S are not symmetric. Indeed, not all (R, S)-codes are (S, S)-codes. To see this, consider once again $X = \{ab, c\}$, and suppose that $R = \langle (a, c) \rangle$ and $S = \langle (a, c), (b, c) \rangle$. Now X is an (R, R)-code, but not an (S, S)-code.

As a corollary of the previous theorem, we show that the (R, S)-codes are always codes in the usual meaning. In other words, if a subset $X \subseteq \mathcal{A}^*$ is an (R, S)-code for some relations R and S, it means that the words in X^* can be uniquely factored.

Corollary 4.2. *Every (R, S)-code X is a code.*

Proof. An (R, S)-code is an (ι, S)-code by Theorem 4.1(i). Thus, it is an (ι, ι)-code by Theorem 4.3. □

We note that the converse of Corollary 4.2 does not hold in general; see Example 4.1, where the set X is clearly a code but not an $(R_1 \cup R_2, \iota)$-code. The following figure illustrates the results of the corollary and Theorem 4.1.

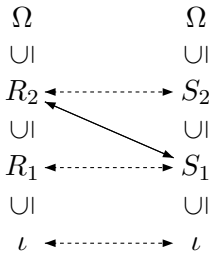


Figure 4.1: An (R_2, S_1)-code is also an (R_1, S_1)-code, an (R_2, S_2)-code and an (ι, ι)-code.

4.1.1 Sardinas–Patterson Theorem

In [68] Sardinas and Patterson gave their famous algorithm for deciding whether a given set of words is a code or not. The corresponding problem for partial words was proved to be decidable in [11]. However, the proof seems to be quite technical compared to the case of total words. It is based on a domino technique introduced in [49]. Here we give a simple solution for the more general problem of deciding whether a given set X is an (R, S)-code or not. The essential part of the procedure is to solve the problem for (R, R)-codes, i.e., for weak R-codes. We use a suitable modification of the Sardinas–Patterson theorem.

Theorem 4.4 (Modified Sardinas–Patterson). *Let R be a similarity relation on $\mathcal{A}^*$ and let $X \subseteq \mathcal{A}^+$. Set $U_1 = R(X)^{-1}X \setminus \{\varepsilon\}$, and define*

$$U_{n+1} = R(X)^{-1}U_n \ \cup \ R(U_n)^{-1}X$$

for $n \geq 1$. The set X is a weak R-code if and only if none of the sets U_n contains the empty word ε.

The proof of the previous theorem is modified from the proof of the Sardinas–Patterson theorem in [8]. First we prove the following lemma.

Lemma 4.1. *Let $X \subseteq \mathcal{A}^+$. For all $n \geq 1$ and $1 \leq k \leq n$, we have $\varepsilon \in U_n$ if and only if there exist $u \in U_k$ and integers $i, j \geq 0$ such that*

$$uX^i \cap R(X^j) \neq \emptyset \quad \textit{and} \quad i + j + k = n. \tag{4.2}$$

Proof. We prove the statement for all n by descending induction on k. Assume first that $k = n$. If $\varepsilon \in U_n$, then the condition (4.2) is satisfied with $u = \varepsilon$ and $i = j = 0$. Conversely, if the condition is satisfied, then $i = j = 0$ and $\{u\} \cap \{\varepsilon\} \neq \emptyset$. Thus, $u = \varepsilon$ and consequently $\varepsilon \in U_n$.

Now, let $n > k \geq 1$ and suppose that the claim holds for $n, n-1, \ldots, k+1$. If $\varepsilon \in U_n$, then by the induction hypothesis, there exist a word $u \in U_{k+1}$ and integers $i, j \geq 0$ such that $uX^i \cap R(X^j) \neq \emptyset$ and $i + j + (k+1) = n$. Thus, there exist words $x_1, \ldots, x_i, y_1, \ldots, y_j \in X$ such that

$$y_1 \cdots y_j \, R \, u x_1 \cdots x_i.$$

Since $u \in U_{k+1}$, there are two cases: either there exists $y \in R(X)$ such that $yu \in U_k$ or there exists $v \in R(U_k)$ such that $vu \in X$. In the first case we have $y' \, R \, y$ for some $y' \in X$ and, by the multiplicativity of R,

$$y' y_1 \cdots y_j \, R \, y u x_1 \cdots x_i.$$

Consequently, there exists a word $yu \in U_k$ such that $yuX^i \cap R(X^{j+1}) \neq \emptyset$ and $i + (j+1) + k = n$. In the second case there exists $v' \in U_k$ such that $v' \, R \, v$ and by the multiplicativity and symmetry of R we have

$$v u x_1 \cdots x_i \, R \, v' y_1 \cdots y_j.$$

Hence, there exists a word $v' \in U_k$ such that $v'X^j \cap R(X^{i+1}) \neq \emptyset$ and $j + (i+1) + k = n$.

Conversely, assume that there exist a word $u \in U_k$ and integers $i, j \geq 0$ such that $uX^i \cap R(X^j) \neq \emptyset$ and $i + j + k = n$. Then

$$y_1 \cdots y_j \, R \, u x_1 \cdots x_i$$

for some $x_1, \ldots, x_i, y_1, \ldots, y_j \in X$. If $j = 0$, then $i = 0$, $k = n$ and we are in the case considered in the beginning of this proof. Hence, let us assume that $j > 0$. We have two cases:

Case 1. Assume that $|u| \geq |y_1|$. By the simplifiability of R, we may write $u = y_1'v$, where $y_1 \, R \, y_1'$ and $v \in \mathcal{A}^*$. Then $v \in U_{k+1}$ and $y_2 \cdots y_j \, R \, v x_1 \cdots x_i$. Thus, $vX^i \cap R(X^{j-1}) \neq \emptyset$ and $i + (j-1) + (k+1) = n$. By the induction hypothesis, $\varepsilon \in U_n$.

Case 2. Assume that $|u| < |y_1|$. We write $y_1 = u'v$, where $u' \, R \, u$ and $v \in \mathcal{A}^+$. Then, by the symmetry of R, $v \in U_{k+1}$ and $x_1 \cdots x_i \, R \, v y_2 \cdots y_j$. Thus, $vX^{j-1} \cap R(X^i) \neq \emptyset$ and $(j-1) + i + (k+1) = n$. Again $\varepsilon \in U_n$ by the induction hypothesis. □

Now we are ready to prove the theorem.

Proof of Theorem 4.4. If X is not a weak R-code, then there exist positive integers m and n and words $x_1, \ldots, x_m, y_1, \ldots, y_n \in X$ such that

$$x_1 \cdots x_m \, R \, y_1 \cdots y_n \quad \text{and} \quad (x_1, y_1) \notin R.$$

By the simplifiability of R, it follows that $|x_1| \neq |y_1|$. By symmetry, we may assume that $|x_1| < |y_1|$, which implies that $y_1 = x_1'u$ for some $u \in \mathcal{A}^+$ and $x_1 \, R \, x_1'$. By the simplifiability of R, we also have $x_2 \cdots x_m \, R \, u y_2 \cdots y_n$. Thus, $u \in U_1$ and $uX^{n-1} \cap R(X^{m-1}) \neq \emptyset$. According to Lemma 4.1, we have $\varepsilon \in U_{m+n-1}$.

Conversely, if $\varepsilon \in U_n$, then choose $k = 1$ in Lemma 4.1. Hence, there exist a word $u \in U_1$ and integers $i, j \geq 0$ such that $i + j = n - 1$ and $uX^i \cap R(X^j) \neq \emptyset$. In other words, there exist words $x_1, \ldots, x_i, y_1, \ldots, y_j \in X$ such that $y_1 \cdots y_j \, R \, u x_1 \cdots x_i$. Since $u \in U_1$, we have $x = yu$ for some $y \in R(X)$ and $x \in X$. Furthermore, we know that $|x| \neq |y|$, since $u \neq \varepsilon$ by the definition of U_1. Since $y \in R(X)$, there exists $y' \in X$ such that $y' \, R \, y$. By multiplication, $y'y_1 \cdots y_j \, R \, yux_1 \cdots x_i$, which gives us a relation $y'y_1 \cdots y_j \, R \, x x_1 \cdots x_i$ on X^+. Since $|y| = |y'| \neq |x|$, we have $(y', x) \notin R$. This means that X is not a weak R-code. □

Note that if X is finite, there exist only finitely many different sets U_n, since all elements of U_n are suffixes of words in X. Therefore, the lengths of the elements are less than $\max\{|x| \mid x \in X\}$. Hence, the sequence of the sets U_i must be ultimately periodic, and it can be effectively determined whether a finite set of words is a relational code or not. This is described in the following procedure. An efficient way of implementing the Sardinas–Patterson test is given later in Section 4.5, where the algorithmic aspect is discussed more thoroughly; see Algorithm 4.3 and Example 4.8.

Procedure 4.1. SardinasPatterson(X, R, S)

Let the input be a finite set $X \subseteq \mathcal{A}^+$ and similarity relations R and S on words over $\mathcal{A}$.

1. Set $U_1 = R(X)^{-1}X \setminus \{\varepsilon\}$.
2. Iterate $U_n = R(X)^{-1}U_{n-1} \cup R(U_{n-1})^{-1}X$ for $n \geq 2$ until $U_n = U_{n-t}$ for some $t = 1, 2, \ldots, n-1$.
3. Check whether $\varepsilon \notin \bigcup_{j=1}^{n-1} U_j$. If the empty word belongs to the union, then return `FALSE`. Otherwise, continue.
4. Check whether $R_X \subseteq S_X$. If the answer is positive, then return `TRUE`. Otherwise, return `FALSE`.

The output is `TRUE` if and only if X is an (R, S)-code.

The correctness of the procedure is an immediate consequence of Theorem 4.4 and Theorem 4.3. If $\varepsilon \in \bigcup_{j=1}^{n-1} U_j$, then X is not a weak R-code by the modified Sardinas–Patterson theorem. By the characterization of Theorem 4.3, this means that X cannot be an (R, S)-code either. On the other hand, positive answers in steps 3 and 4 imply that X is an (R, R)-code and $R_X \subseteq S_X$. Thus, X is an (R, S)-code by Theorem 4.3.

The procedure can also be used for codes of partial words. A set X of partial words is a *pcode* if for all integers $m, n \geq 1$ and partial words $u_1, \ldots, u_m, v_1, \ldots, v_n$, the condition

$$u_1 \cdots u_m \uparrow v_1 \cdots v_n$$

implies $m = n$ and $u_i = v_i$ for $i = 1, 2, \cdots, m$. By the considerations of Section 3.2.1, we conclude that pcodes correspond to $(R_\uparrow, \iota)$-codes. Hence, applying the previous procedure for $X \subseteq \mathcal{A}_\diamond^+$, $R = R_\uparrow$ and $S = \iota$ we get the following corollary originally proved in [11].

Corollary 4.3. *There is an algorithm to decide whether a set of partial words is a pcode or not.*

4.1.2 Minimal and Maximal Relations

Using algorithms based on the idea of the Sardinas–Patterson theorem we can test whether a set of words is a code, a pcode, or more generally, an (R, S)-code. Moreover, alteration relations and fidelity relations enable us to analyze coding properties of relational codes in more detail. For this purpose, define the *size* $\mathrm{sz}(R)$ of a similarity relation R to be the number of pairs in the corresponding compatibility relation of letters, i.e.,

$$\mathrm{sz}(R) = |R_\mathcal{A}|.$$

We may study the error capacity of a relational code X by measuring possible alteration relations. In this way, we may ask what is the maximal size of the alteration relation if the fidelity relation is fixed. On the other hand, we may investigate the coding precision of X, i.e., how small the fidelity relation can be if the alteration relation is known.

In order to analyze coding properties of word sets in this respect, we have to be able to find the relations which are optimal, i.e., minimal or maximal in the following sense.

Definition 4.2. Let X be an (R, S)-code. The similarity relation R is called a *maximal alteration relation* with respect to X and S if, for all $R' \supset R$, X is not an (R', S)-code. The set of maximal alteration relations with respect to X and S is denoted by $A_{\max}(X, S)$. Similarly, the relation S is called a *minimal fidelity relation* with respect to X and R if, for all $S' \subset S$, X is not an (R, S')-code. The set of minimal fidelity relations with respect to X and R is denoted by $F_{\min}(X, R)$.

Note that every (R, S)-code is an (ι, S)-code by Theorem 4.1(i) and an (R, Ω)-code by Theorem 4.1(ii). Thus, the concepts opposed to the ones defined above are trivial: For an (R, S)-code X, the identity relation is the unique minimal alteration relation and the universal relation is the unique maximal fidelity relation. Before describing algorithms which find the sets $A_{\max}(X, S)$ and $F_{\min}(X, R)$, we make a few easy observations.

Theorem 4.5. *The minimal and maximal relations have the following properties:*

(i) *X is a code if and only if there exists a similarity relation S such that $A_{\max}(X, S) \neq \emptyset$.*

(ii) *X is a code if and only if there exists a similarity relation R such that $F_{\min}(X, R) \neq \emptyset$.*

(iii) *X is an (R, R)-code if and only if $F_{\min}(X, R) \neq \emptyset$.*

(iv) *For all (R, R)-codes X, $F_{\min}(X, R)$ consists of a unique element.*

(v) *If $S_1 \subset S_2$, then for all $R \in A_{\max}(X, S_1)$ there exists $R' \in A_{\max}(X, S_2)$ such that $R \subseteq R'$.*

Proof. (i): Clearly, if X is a code, then there exist maximal alteration relations at least for $S = \iota$. On the other hand, suppose that $R \in A_{\max}(X, S)$ for some relation S. Then X is an (R, S)-code and consequently a code by Corollary 4.2.

(ii): This is equivalent to Case (i). The set $F_{\min}(X, R)$ is empty for all R if and only if $A_{\max}(X, S)$ is empty for all S.

(iii): Suppose first that $S \in F_{\min}(X, R)$. Hence, X is an (R, S)-code and therefore an (R, R)-code by Theorem 4.3. Conversely, if X is an (R, R)-code, then $F_{\min}(X, R)$ is trivially nonempty.

(iv): The intersection S' of all similarity relations S such that X is an (R, S)-code is a similarity relation. Using Theorem 4.2, we conclude that X is an (R, S')-code. Hence, it must be the unique minimal fidelity relation in $F_{\min}(X, R)$.

(v): Let $S_1 \subset S_2$ and let R belong to $A_{\max}(X, S_1)$. Hence, X is an (R, S_1)-code and, by Theorem 4.1(ii), it is also an (R, S_2)-code. Thus, either R is maximal with respect to X and S_2 or $R \subset R'$ for some maximal $R' \in A_{\max}(X, S_2)$. □

Note that there may be several maximal relations in $A_{\max}(X, S)$, whereas by cases (iii) and (iv) of Theorem 4.5, the set $F_{\min}(X, R)$ is either empty or singleton. For example, in Example 4.1 both relations R_1 and R_2 are maximal. With respect to X these two similarity relations seem to have symmetric roles and they have the same size. This need not be the case in general. A more complicated case will be seen later in Example 4.3.

Next we present two algorithms for the minimal and maximal relations. In the pseudocode of the following algorithm, the ith letter in a word x is denoted by $x(i)$.

Algorithm 4.1. MINIMALFIDELITY(X, R)

`INPUT`: a similarity relation R on $\mathcal{A}^*$, a finite (R, R)-code X.

```
1  S ← ∅
2  FOR EACH {x, y} such that (x, y) ∈ R_X DO
3    FOR i ← 1 TO |x| DO S ← S ∪ {(x(i), y(i))}
4  return ⟨S⟩
```

`OUTPUT`: the minimal fidelity relation in $F_{\min}(X, R)$.

Before giving an example, we prove that the previous algorithm works correctly and in polynomial time.

Theorem 4.6. *If X is a finite (R, R)-code,* MINIMALFIDELITY(X, R) *gives $F_{\min}(X, R)$. Denote the number of pairs of letters in the finite representation of the input R by m and denote $n = \sum_{x \in X} |x|$. Then the time complexity of* MINIMALFIDELITY(X, R) *is $\mathcal{O}(n^2 m)$.*

Proof. Since X is an (R, R)-code, the unique minimal element S' belonging to $F_{\min}(X, R)$ must be a subset of R. On the other hand, $R_X \subseteq S'_X$ by Theorem 4.3. Thus, we must have $R_X = S'_X$. Note that this does not mean that $S' = R$, since in R there may exist pairs of letters which never occur in any R-similar words of X. Now the algorithm MINIMALFIDELITY(X, R) ensures that, for all $x, y \in X$, the relation $x\,R\,y$ implies $x\,S\,y$, i.e, $R_X \subseteq S_X$. Furthermore, the relation S is necessarily minimal. Indeed, if we omit any

pair (a, b) with $a \neq b$ from S, then for some words $x, y \in X$ with $x\,R\,y$, we would have $(x, y) \notin S$.

We assume that in the algorithm the (infinite) similarity relation R is given using a finite representation $\langle r_1, \ldots, r_l \rangle$ defined in Section 3.1. Suppose that the input relation R has a representation consisting of m pairs of letters. Then the operation of deciding whether two letters are R-similar requires comparing the letters with the m related pairs and this can be done in time $\mathcal{O}(m)$. The two `FOR`-loops require at most $\mathcal{O}(n^2)$ comparisons. Hence, the complexity is $\mathcal{O}(n^2 m)$. □

The following example describes how Algorithm 4.1 works in practice.

Example 4.2. Consider the set $X = \{aba, bba, dbc, adcd, bccd\}$ in the four letter alphabet $\mathcal{A} = \{a, b, c, d\}$. Let

$$R = \langle (a, b), (b, c), (c, d), (d, a) \rangle.$$

The algorithm MinimalFidelity(X, R) compares R-similar words in X. From the relation $aba\,R\,bba$ we get $S \leftarrow \{(a, b), (b, b), (a, a)\}$ and $adcd\,R\,bccd$ gives us $S \leftarrow S \cup \{(a, b), (d, c), (c, c), (d, d)\}$. Hence, the algorithm returns

$$\langle (a, b), (b, b), (a, a), (d, c), (c, c), (d, d) \rangle = \langle (a, b), (d, c) \rangle.$$

Since X is an (R, R)-code, the output is the minimal fidelity relation with respect to X and R.

Finding the maximal alteration relations $A_{\max}(X, S)$ is a more complicated task. By Theorem 4.3 there are two properties that restrict the maximal alteration relations. Namely, we must have $R_X \subseteq S_X$, but at the same time X must be a weak R-code. We do not know which one of these conditions is more restrictive. Therefore, in the following algorithm we systematically run through all similarity relations on $\mathcal{A}^*$ using a directed graph $G(\mathcal{A})$ defined as follows. The set of vertices V in $G(\mathcal{A})$ is the set of all similarity relations on $\mathcal{A}^*$. Note that since a similarity relation is induced by a relation on letters and the alphabet is finite, the vertex set is also finite. The set of edges is

$$E = \{(R_1, R_2) \mid R_2 = R_1 \cup \{(a, b), (b, a)\} \text{ for some } a \neq b\}.$$

Recall that similarity relations are reflexive and symmetric, and the size of a similarity relation R satisfies $\mathrm{sz}(R) = |R_{\mathcal{A}}| \geq |\mathcal{A}|$. Moreover, the maximal size is the size of the universal similarity relation $\mathrm{sz}(\Omega) = |\mathcal{A}|^2$.

Algorithm 4.2. MAXIMALALTERATION$(\mathcal{A}, X, S)$

INPUT: a finite alphabet $\mathcal{A}$, a finite set $X \subseteq \mathcal{A}^+$, a similarity relation S.

1 Construct a directed graph $G(\mathcal{A}) = (V, E)$ by setting
2 $V \leftarrow$ the set of similarity relations on $\mathcal{A}^*$.
3 $E = \{(R_1, R_2) \mid R_2 = R_1 \cup \{(a, b), (b, a)\}$ for some $a \neq b\}$.
4 FOR $i = |\mathcal{A}|$ TO $|\mathcal{A}|^2$ DO
5 FOR EACH $R \in V$ such that $\mathrm{sz}(R) = i$ DO
6 IF X is not an (R, S)-code THEN
7 modify V and E by deleting R and all vertices accessible from R.
8 return the set of all the vertices $R \in V$ with no edges starting from R.

OUTPUT: $A_{\max}(X, S)$.

Theorem 4.7. *Let $X \subseteq \mathcal{A}^+$ be finite and let S be a similarity relation. Algorithm* MAXIMALALTERATION$(\mathcal{A}, X, S)$ *returns the set $A_{\max}(X, S)$. If the alphabet $\mathcal{A}$ is of fixed size, then the algorithm works in polynomial time.*

Proof. For each compatibility relations on $\mathcal{A}$, the algorithm decides whether X is an (R, S)-code or not. This can be done, for example, using Algorithm 4.3. For some vertices which are not (R, S)-codes, we do not need to use any algorithm because of deletions in line 7. Indeed, if X is not an (R, S)-code for a relation R, then X is not an (R', S)-code for any of the relations $R' \supseteq R$ by Theorem 4.1(i). This justifies the modifications of the directed graph. In other words, only relations corresponding to relational codes are maintained in the graph. The directed edges describe the increasing order on the size of the relations (vertices). Thus, after deleting all vertices corresponding to non-codes, the remaining vertices with no outgoing edges must correspond to maximal relations.

Let us now suppose that the alphabet $\mathcal{A}$ is of fixed size. Then the construction of the graph $G(\mathcal{A})$ takes a fixed number of operations. Similarly, running through the graph can be done in a fixed time and the size of the relations is bounded by a fixed number. Thus, the complexity of our algorithm is just a constant multiplied by the time used for testing the (R, S)-code property. By Theorem 4.23, this can be done in polynomial time. □

Let us consider the following example showing how the algorithm works over a three letter alphabet.

Example 4.3. Let $\mathcal{A} = \{a, b, c\}$, $X = \{ab, bccb, ca\}$ and $S = \langle (a, b), (a, c) \rangle$. The directed graph $G(\mathcal{A}) = (V, E)$ is illustrated in Figure 4.2. It is clear that X is a code, since it is even a prefix code (no code word is a prefix of another code word). For all relations R with one generator, the

set X is also an (R,S)-code. Indeed, comparing the two first letters of each of the words in X we notice that at least two generator pairs in $R_{\mathcal{A}}$ are needed in order to achieve two different R-similar words in X^+. In the case $R = \langle (a,b),(a,c)\rangle$, Procedure 4.1 (SardinasPatterson(X,R,S)) reveals that X is an (R,S)-code. In the other cases where the generator set consists of two elements, we have nontrivial relations such as $bccb\,R\,ab \cdot ca$ and $bccb\,R\,ca \cdot ab$. Thus, if $R = \langle (a,b),(b,c)\rangle$ or $R = \langle (a,c),(b,c)\rangle$, then the set X is not an (R,S)-code and these vertices are deleted. Consequently, the vertex Ω is also deleted. The deleted vertices are marked with a double circle in Figure 4.2. Hence, in the final step we have two vertices with no outgoing edges, and the algorithm MaximalAlteration$(\mathcal{A},X,S)$ returns $A_{\max}(X,S) = \{\langle (b,c)\rangle, \langle (a,b),(a,c)\rangle\}$. Note that these two maximal alteration relations are by no means isomorphic. They do not even have the same size.

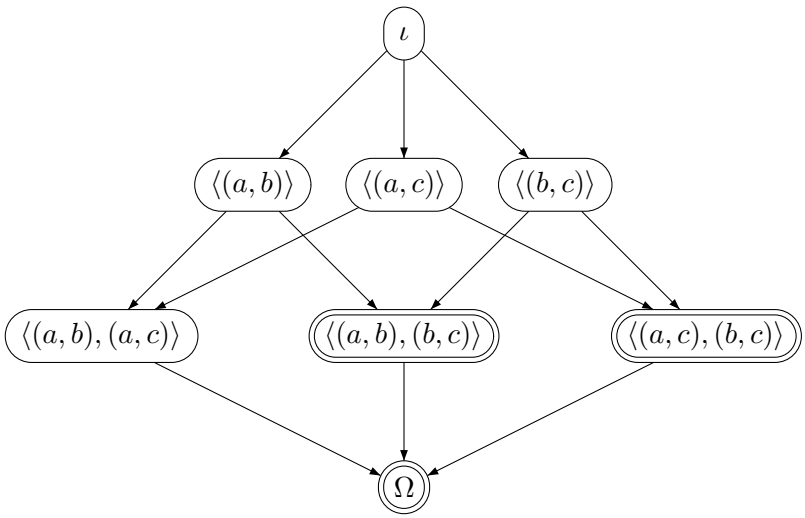


Figure 4.2: The graph $G(\mathcal{A})$ of Example 4.3

From another viewpoint, if we allow arbitrary alphabets, the problem of finding maximal alteration relations is actually very difficult. Namely, the corresponding decision problem is NP-complete. The abbreviation NP stands for nondeterministic polynomial time; for more on NP-complete problems, see [39]. Define the number $M_R(X,S)$ to be the maximal size of a relation in $A_{\max}(X,S)$, i.e., $M_R(X,S) = \max\{\mathrm{sz}(R) \mid R \in A_{\max}(X,S)\}$. We formulate the following problem:

Problem:	MAXIMAL RELATION
Instance:	A finite alphabet $\mathcal{A}$, a set of words $X \subseteq \mathcal{A}^+$, a similarity relation S on $\mathcal{A}^*$ and a positive integer k
Question:	Is $M_R(X,S) \geq k$?

The problem above is related to the following problem of graphs. Let $G = (V,E)$ be a graph. A set $W \subseteq V$ is a *vertex cover* of G if for each edge $(u,v) \in E$ at least one of u and v belongs to W. The *cover number* $\mathrm{c}(G)$ of a graph G is the minimal cardinality of a vertex cover of G.

Problem:	VERTEX COVER
Instance:	A finite graph $G = (V,E)$ and a positive integer k
Question:	Is $\mathrm{c}(G) \leq k$?

This problem is known to be NP-complete. A proof can be found in [39]. Next we will show how to reduce this problem to the problem MAXIMAL RELATION.

Theorem 4.8. *The problem MAXIMAL RELATION is NP-complete.*

Proof. First we must show that MAXIMAL RELATION is an NP-problem. This is clear since, for an alphabet $\mathcal{A}$, a set $X \subseteq \mathcal{A}^+$, a positive integer k, a relation S on $\mathcal{A}^*$ and an arbitrary relation R on $\mathcal{A}^*$ with $\mathrm{sz}(R) \geq k$, we can verify in polynomial time whether X is an (R,S)-code. If the answer is positive, then clearly $M_R(X,S) \geq k$.

Secondly, our aim is to prove that the NP-complete problem VERTEX COVER can be polynomially reduced to the problem MAXIMAL RELATION, which means that a polynomial time algorithm solving the second problem induces a polynomial time algorithm for the first problem. More formally, it means that any input x of the problem VERTEX COVER can be turned into an input $f(x)$ of MAXIMAL RELATION in polynomial time and $f(x)$ is a positive instance of MAXIMAL RELATION if and only if x is a positive instance of VERTEX COVER.

Next we define the function f which maps a pair (G,k) to a four-tuple $(\mathcal{A},X,S,l)$ in the following way. Assume that the graph $G = (V,E)$ has vertices $V = \{v_1,\ldots,v_n\}$ and edges $E = \{e_1,\ldots,e_m\}$. We may assume that the graph G has no isolated vertices, i.e., vertices of degree zero, since they are not considered in the VERTEX COVER problem. For each edge $e_i = (v_{i_1}, v_{i_2})$ we define two words $iv_{i_1}v_{i_2}$ and iaa. Let X consist of all these words for every $i = 1,2,\ldots,m$. We also choose

$$\begin{aligned} S &= \iota, \\ \mathcal{A} &= \{1,2,\ldots,m\} \cup \{a\} \cup \{v_1,\ldots,v_n\}, \\ l &= |\mathcal{A}|^2 - 2k - (m^2 - m). \end{aligned}$$

Thus, the alphabet has cardinality $m+n+1$. Denote by $\|X\|$ the sum of the lengths of all words in X. Clearly $|X| = 2m$ and since all the words are of length 3 we have $\|X\| = 6m$. Thus, this construction is polynomial.

Now suppose that W is a vertex cover of G and $|W| \leq k$. We show that there is a relation R with $\mathrm{sz}(R) \geq l$ such that X is an (R,ι)-code. First define

$$T = \{(i,j) \mid i,j \in \{1,2,\ldots,m\}, i \neq j\}$$

and

$$U = \{(a,v) \mid v \in W\} \cup \{(v,a) \mid v \in W\}.$$

Now let us choose

$$R = \langle \Omega_{\mathcal{A}} \setminus (T \cup U) \rangle.$$

This relation is of size $\mathrm{sz}(R) = |\mathcal{A}|^2 - 2|W| - (m^2 - m)$ which, by the assumption $|W| \leq k$, is greater than or equal to l. Now consider all possible pairs of words in $X \times X$. If $i \neq j$, then $(iv_{i_1}v_{i_2}, jv_{j_1}v_{j_2}) \notin R$ by the definition of T. Thus, we have to compare only words starting with the same letter. For each $i = 1,2,\ldots,m$ there is only one such pair, namely $(iv_{i_1}v_{i_2}, iaa)$. Since W is a vertex cover, at least one of v_{i_1} and v_{i_2} belongs to W. Thus, at least one of the pairs (a,v_{i_1}) and (a,v_{i_2}) is in U. Since all the words in X are of length 3 and they are R-similar only with themselves, we conclude that X is an (R,ι)-code with $\mathrm{sz}(R) \geq l$.

Conversely, suppose that there is a relation R of size $\mathrm{sz}(R) \geq l$ such that X is an (R,ι)-code. Denote

$$W = \{v \in V \mid \ (a,v) \notin R\}.$$

Since X is an (R,ι)-code, we know that $T \cap R = \emptyset$. Otherwise, $iaa\,R\,jaa$ for two different i and j in $\{1,2,\ldots,m\}$ and the coding property does not hold. Hence, let us consider words starting with the same letter $i = 1,2,\ldots,m$. For each i, we have a unique pair of words $(iv_{i_1}v_{i_2}, iaa)$. Since X is an (R,ι)-code we have to have $(iaa, iv_{i_1}v_{i_2}) \notin R$. Thus, at least one of the relations (a,v_{i_1}) and (a,v_{i_2}) is not in R. This implies that at least one of the vertices v_{i_1} and v_{i_2} is in W and W is really a vertex cover of G. The number of letter pairs not belonging to R is less than or equal to $|\mathcal{A}|^2 - l = 2k + (m^2 - m)$. The letters V do not occur in the $m^2 - m$ pairs of T not belonging to R and therefore, these pairs have no effect on W. Hence, $|W| \leq 2k$. Since R is symmetric, we finally conclude that $|W| \leq k$. □

4.2 Relationally Free Monoids

By definition, all nonempty elements in the monoid X^* generated by an (R,S)-code X have a "relationally unique" X-factorization. In this section we first consider relationally unique factorizations more closely and introduce

relationally free monoids taking (R, S)-unique factorization extensions as a starting point. In Section 4.2.2 we characterize these extensions and free monoids using stability conditions. We note that here our considerations are restricted to monoids, but all the results can be modified for semigroups as well; see [45]. We emphasize that from here on, all the monoids considered in this chapter lie within a fixed free monoid $\mathcal{A}^*$.

4.2.1 Unique Factorization

Let $\overline{M}$ be a monoid containing a monoid M. Denote the base of $\overline{M}$ by B. Since $M \subseteq \overline{M} = B^*$, each word $u \in M$ has at least one B-factorization: $u = u_1 \cdots u_m$ with $u_i \in B$ for $i = 1, 2, \ldots, m$. The element u is said to possess an (R, S)-*unique B-factorization in M* if, for every word $v_1 \cdots v_n \in R(u) \cap M$ where $v_j \in B$ for $j = 1, 2, \ldots, n$, we have

$$n = m \text{ and } u_i \, S \, v_i \text{ for all } i = 1, 2, \ldots, m. \tag{4.3}$$

Note that, by the reflexivity of R and S, an (R, S)-unique B-factorization implies an (ι, ι)-unique B-factorization, i.e., a unique B-factorization. Moreover, we say that $u \in M$ possesses an (R, S)-*unique B-factorization in $\overline{M}$* if (4.3) holds also for every $v_1 \cdots v_n \in R(u) \cap \overline{M}$.

Next we define two extensions of the monoid M with respect to (R, S)-unique factorization.

Definition 4.3. A monoid $\overline{M}$ containing M is called an *inner (R, S)-unique factorization extension* of M if every element of M has an (R, S)-unique Base$(\overline{M})$-factorization in M. Moreover, the monoid $\overline{M}$ is called an *outer (R, S)-unique factorization extension* of M if every element of M has an (R, S)-unique Base$(\overline{M})$-factorization in $\overline{M}$.

Hence, every outer (R, S)-unique factorization extension of M is also an inner (R, S)-unique factorization extension. In the sequel, if the type of the (R, S)-unique factorization extension is not specified, the statement is valid for both inner or outer extensions. For these extensions we use the abbreviation *(R, S)-ufe*.

In the above, the (R, S)-ufe $\overline{M}$ is called a *strong R-unique factorization extension* if $S = \iota$ and a *weak R-unique factorization extension* if $R = S$. An (ι, ι)-ufe coincides with the definition of a unique factorization extension given in Section 2.4. The following two theorems describing the role of these special cases can be compared with Theorem 4.3 and Corollary 4.2. The essential part of the proofs of Theorems 4.9 and 4.10 is represented in a separate lemma below.

Lemma 4.2. *Let $\overline{M} \subseteq \mathcal{A}^*$ be a monoid with base B and containing a monoid M. Assume that $u \in M$ has an (R, S)-unique B-factorization in $L \in \{M, \overline{M}\}$. For similarity relations R' and S', we have:*

(i) If $R' \subseteq R$, then u has an (R', R')-unique B-factorization in L.

(ii) If $R(u) \cap L \subseteq S'(u) \cap L$, then u has an (R, S')-unique B-factorization in L.

Proof. Suppose that the words $u = u_1 \cdots u_m \in M$ and $v = v_1 \cdots v_n \in L$ satisfy $u_1 \cdots u_m \, R' \, v_1 \cdots v_n$, where $u_i, v_j \in B$ for all i and j. If $R' \subseteq R$, we have $u_1 \cdots u_m \, R \, v_1 \cdots v_n$. Since u has an (R, S)-unique B-factorization in L, we have $m = n$ and $u_i \, S \, v_i$ for all $i = 1, 2, \ldots, m$. Especially, this means that $|u_i| = |v_i|$. Using the simplification rule of similarity relations in the case $u_1 \cdots u_m \, R' \, v_1 \cdots v_n$, we get $u_i \, R' \, v_i$ for all i. Hence, the first claim is proved.

Suppose next that $R(u) \cap L \subseteq S'(u) \cap L$. Let u and v be as above. If $u \, R \, v$, then $v \in R(u) \cap L$ and, by the assumption, $u_1 \cdots u_m \, S' \, v_1 \cdots v_n$. Using the length argument and simplification rule as above, we have $u_i \, S' \, v_i$ for all i. Hence, u has an (R, S')-unique B-factorization in L. □

Theorem 4.9. *Let M be a submonoid of $\mathcal{A}^*$. Every inner (resp. outer) (R, S)-ufe of M is an inner (resp. outer) (ι, ι)-ufe of M.*

Proof. Let $\overline{M}$ be an inner (R, S)-ufe of M and let B be the base of $\overline{M}$. By the definition of $\overline{M}$, every element of M has an (R, S)-unique B-factorization in M. Since $\iota \subseteq R$, every element of $\overline{M}$ also has an (ι, ι)-unique B-factorization in M by Lemma 4.2(i). Thus, $\overline{M}$ is an inner (ι, ι)-ufe of M. The proof for the outer (R, S)-ufe is similar. □

For inner and outer (R, S)-unique factorization extensions we have characterizations in terms of weak R-unique factorization extensions with some additional conditions concerning the order of the similarity relations R and S.

Theorem 4.10. *Let $\overline{M} \subseteq \mathcal{A}^*$ be a monoid containing a monoid M. Then the following statements hold:*

(i) $\overline{M}$ is an inner (R, S)-ufe of M if and only if $\overline{M}$ is an inner (R, R)-ufe of M and $R_M \subseteq S_M$.

(ii) $\overline{M}$ is an outer (R, S)-ufe of M if and only if $\overline{M}$ is an outer (R, R)-ufe of M and, for all $x \in M$, we have $R(x) \cap \overline{M} \subseteq S(x) \cap \overline{M}$.

Proof. Let $\overline{M}$ be an inner (R, S)-ufe of M. Let B be the base of $\overline{M}$. By the definition of $\overline{M}$, every element of M has an (R, S)-unique B-factorization in M. By Lemma 4.2(i), every element of M has also an (R, R)-unique B-factorization in M. Hence, $\overline{M}$ is an inner (R, R)-ufe of M. Clearly, $R_M \subseteq S_M$ for all inner (R, S)-unique factorization extensions of M.

Conversely, suppose that $\overline{M}$ is an inner (R, R)-ufe of M and $R_M \subseteq S_M$. Let B be the base of $\overline{M}$. Consider an element $x \in M$. Since $R_M \subseteq S_M$, it follows that $R(x) \cap M \subseteq S(x) \cap M$. Hence, the word x has an (R, S)-unique

B-factorization in M by Lemma 4.2(ii). This holds for all $x \in M$. Therefore, $\overline{M}$ is an inner (R,S)-ufe of M.

In order to prove claim (ii), replace the condition $R_M \subseteq S_M$ with the condition $R(x) \cap \overline{M} \subseteq S(x) \cap \overline{M}$ and consider R-compatible factorizations of elements of M in the monoid $\overline{M}$. □

Next we define relationally free monoids in terms of unique factorization extensions.

Definition 4.4. A monoid $M \subseteq \mathcal{A}^*$ which is its own (R,S)-ufe is called *(R,S)-free*.

Note that in the definition we do not have to specify which type of (R,S)-ufe we mean since the definitions of outer and inner extensions coincide in this case. *Strong R-freeness* and *weak R-freeness* of an (R,S)-free monoid are defined similarly as above, i.e., $S = \iota$ and $R = S$, respectively.

We may also characterize (R,S)-free monoids using (R,S)-codes in the following way.

Theorem 4.11. *Let $X \subseteq \mathcal{A}^*$. The following conditions are equivalent.*

(i) X is an (R,S)-code.
(ii) X^* is (R,S)-free and X is its base.

Proof. Suppose first that X is an (R,S)-code. By Corollary 4.2, X is a code and therefore it is also the base of the free monoid X^* by Theorem 2.2. Consider now an element of X^* with an X-factorization $x_1 \cdots x_m$. Since X is an (R,S)-code, the relational decoding condition (4.1) holds for all $y_1, \ldots, y_n \in X$. Thus X^* is its own (R,S)-ufe and therefore it is (R,S)-free.

Conversely, suppose that X^* is (R,S)-free and X is its base. Since X^* is its own (R,S)-ufe and $\mathrm{Base}(X^*) = X$, the condition (4.1) holds for all $x_1, \ldots, x_m, y_1, \ldots, y_n \in X$. Thus, X is an (R,S)-code. □

By the above theorem, it is clear the (ι,ι)-free monoids are the free monoids in the original meaning of freeness; see Theorem 2.2. Relational freeness of a submonoid of $\mathcal{A}^*$ implies the following facts about the considered similarity relations.

Theorem 4.12. *Let M be an (R,S)-free submonoid of $\mathcal{A}^*$. The following conditions hold.*

(i) If $S \subseteq R$, then $R_M = S_M$.
(ii) If $R \cap S = \iota$, then $R_M = \iota_M$.

Proof. Since M is an (R,S)-free monoid, it is an inner (R,S)-ufe of itself. Hence, $R_M \subseteq S_M$ by Theorem 4.10. If $S \subseteq R$, then also $S_M \subseteq R_M$. Thus, $R_M = S_M$. On the other hand, if $R \cap S = \iota$ and $R_M \subseteq S_M$, then $R_M = R_M \cap S_M = \iota_M$. □

The next result follows from the code characterization of (R, S)-free monoids.

Theorem 4.13. *A monoid $M \subseteq \mathcal{A}^*$ is (R, S)-free if and only if M is (R, R)-free and $R_B \subseteq S_B$ for the base B of M.*

Proof. Let B be the base of the monoid M. By Theorem 4.11, M is (R, S)-free if and only if B is an (R, S)-code. By Theorem 4.3, B is an (R, S)-code if and only if B is an (R, R)-code and $R_B \subseteq S_B$. Using again Theorem 4.11, this is true if and only if $M = B^*$ is (R, R)-free and $R_B \subseteq S_B$. □

We have also the following corollary.

Corollary 4.4. *The free monoid $\mathcal{A}^*$ is (R, S)-free if and only if $R \subseteq S$.*

Proof. By the definition of a similarity relation, the monoid $\mathcal{A}^*$ is (R, R)-free and $\mathcal{A}$ is its base. Thus, by the previous theorem, $\mathcal{A}^*$ is (R, S)-free if and only if $R_{\mathcal{A}} \subseteq S_{\mathcal{A}}$. □

Let us now compare the conditions on the order of the relations R and S in the characterizations of the extensions in Theorem 4.10 and Theorem 4.13. Note that in these, one cannot replace one condition with another. Indeed, let B be the base of a monoid M and let $\overline{M}$ be an extension of M. Then the following implications hold:

$$(\forall\, x \in M : R(x) \cap \overline{M} \subseteq S(x) \cap \overline{M}) \implies R_M \subseteq S_M \implies R_B \subseteq S_B. \quad (4.4)$$

However, the converse implications are not valid in general. Consider the following example.

Example 4.4. Let M be a monoid with base $B = \{a, ac, dd, ddb\}$ and let $R = \langle (a, b), (b, c), (c, d) \rangle$ and $S = \iota$. In Example 4.5 we show that the monoid $\overline{M}$ generated by $\{a, ac, b, dd\}$ is an inner (R, R)-ufe of M. However, it is not an inner (R, S)-ufe of M even though we clearly have $R_B \subseteq S_B$. Indeed, there does not exist any inner (R, S)-ufe of M, since $(dda, ddb) \in R_M \setminus S_M$. Neither does there exist any outer (R, S)-ufe of M, since every outer (R, S)-ufe is also an inner (R, S)-ufe.

Next consider a similarity relations $S' = \langle (a, b) \rangle$. We have $R_M \subseteq S'_M$ and therefore the inner (R, R)-ufe $\overline{M}$ is an inner (R, S')-ufe of M. On the other hand, there does not exist any outer (R, S')-ufe of M. Namely, by Example 4.5, the monoid $\widehat{M} = \{a, b, c, d\}^*$ is the only outer (R, R)-ufe of M, but it is not an outer (R, S')-ufe. For $ac \in M$, we have $ab \in R(ac) \cap \widehat{M}$, but $ab \notin S'(ac) \cap \widehat{M}$.

Note that since an (R,S)-free extension is an outer and an inner (R,S)-unique factorization extension of itself, the inclusion conditions of (4.4) coincide. Namely, suppose that $M = \overline{M}$ and M is (R,S)-free. Then the (R,R)-unique factorization of its elements ensures that

$$R_B \subseteq S_B \Leftrightarrow R_M \subseteq S_M$$

and, for all $x \in M = \overline{M}$, it clearly holds that $R(x) \cap \overline{M} \subseteq S(x) \cap \overline{M}$.

4.2.2 Stability

Free monoids and semigroups are traditionally characterized by the notion of stability; for example, see [8]. In this section we generalize this idea for (R,S)-unique factorizations.

Definition 4.5. A monoid $\overline{M} \subseteq \mathcal{A}^*$ is called *intrinsically* (R,S)*-stable* over a monoid M if $M \subseteq \overline{M}$ and for all $u,v,w,u',v',w' \in \mathcal{A}^*$ satisfying conditions

(i) $u\,R\,u'$, $w\,R\,w'$ and $v\,R\,v'$,
(ii) $uwv, u'w'v' \in M$ and $uw, v, u', w'v' \in \overline{M}$,

we have $u,w \in \overline{M}$ and $u\,S\,u'$. Similarly, $\overline{M}$ is called *extrinsically* (R,S)*-stable* over M if condition (ii) above is replaced by

(ii)' $(uwv \in M$ or $u'w'v' \in M)$ and $uw, v, u', w'v' \in \overline{M}$.

If a monoid is (R,S)-stable over itself, we may briefly call it (R,S)*-stable*.

Note that if a monoid is (R,S)-stable over itself, the definitions of intrinsic and extrinsic (R,S)-stability coincide. As above, we talk about *strong* and *weak* R*-stability* depending upon whether $S = \iota$ or $S = R$. The intrinsic (R,S)-stability of $\overline{M}$ over a monoid M is illustrated in Figure 4.3.

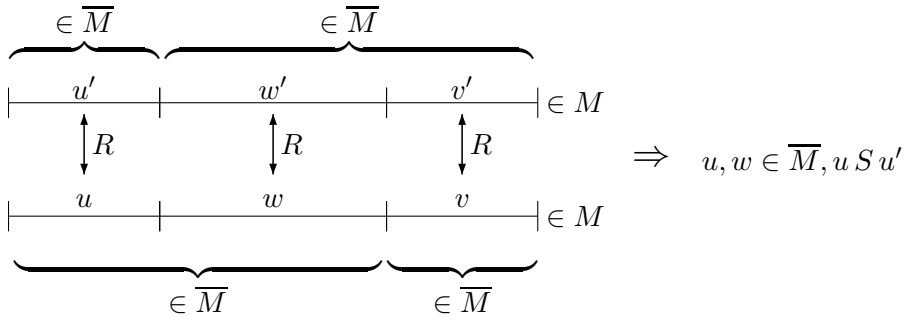


Figure 4.3: Illustration of intrinsic (R,S)-stability of $\overline{M}$ over M

Remark 4.1. The definition of an (R,S)-stable monoid coincides with the original definition of a *stable* word monoid in the case $R = S = \iota$. Note that the relation R makes the definition of intrinsic (resp. extrinsic) (R,S)-stability more complicated compared to the original definition of stability: A monoid $\overline{M}$ is stable over M if, for $u, w, v \in \mathcal{A}^*$,

$$uwv \in M \quad \text{and} \quad u, uw, wv, v \in \overline{M} \implies w \in \overline{M}.$$

In the original definition we do not need words u', v', w', since the identity relation ensures that the words uwv and $u'w'v'$ are exactly the same. Furthermore, in the original definition we assume that $u \in \overline{M}$ while in our new definition we only have $u \in R(\overline{M})$, since u' is in $\overline{M}$. In the relational case, we want to specify that u belongs to $\overline{M}$ as a consequence of stability; see Remark 4.2.

Next we prove that stability and unique factorization are related to each other.

Theorem 4.14. *Let M be a monoid in $\mathcal{A}^*$. A submonoid of $\mathcal{A}^*$ is an inner (resp. outer) (R,S)-ufe of M if and only if it is intrinsically (resp. extrinsically) (R,S)-stable over M.*

Proof. Let us prove the theorem for inner extensions and intrinsic stability. The proof for the outer and extrinsic case is similar. Assume that M is a submonoid of $\overline{M}$. Let $\overline{M}$ be intrinsically (R,S)-stable over M and let B be the base of $\overline{M}$. Suppose now that $\overline{M}$ is not an inner (R,S)-ufe of M. Then there exist words $x_1, \ldots, x_m, y_1, \ldots, y_n \in B$ such that $x_1 \cdots x_m, y_1 \cdots y_n \in M$, $x_1 \cdots x_m \, R \, y_1 \cdots y_n$ and $(x_i, y_i) \in S$ for $i = 1, 2, \ldots, k-1 < \min\{m, n\}$, but $(x_k, y_k) \notin S$. By symmetry, we may suppose that $|x_k| \leq |y_k|$. Let us denote $y_k = y'y''$, where $y', y'' \in \mathcal{A}^*$ and $|y'| = |x_k|$, and $x_{k+1} \cdots x_m = x''x$, where $x'', x \in \mathcal{A}^*$ and $|x''| = |y''|$. Now choose

$$\begin{cases} u &= y_1 \cdots y_{k-1}y', \\ w &= y'', \\ v &= y_{k+1} \cdots y_n, \end{cases} \quad \text{and} \quad \begin{cases} u' &= x_1 \cdots x_k, \\ w' &= x'', \\ v' &= x. \end{cases}$$

Since $|x_i| = |y_i|$ for $i = 1, 2, \ldots, k-1$, the conditions (i) and (ii) of Definition 4.5 are satisfied. Since $\overline{M}$ is intrinsically (R,S)-stable over M, we have $y_1 \cdots y_{k-1}y', y'' \in \overline{M}$ and $y_1 \cdots y_{k-1}y' \, S \, x_1 \cdots x_k$.

If $|x_k| = |y_k|$, then $y' = y_k$ and $x_1 \cdots x_k \, S \, y_1 \cdots y_k$. By the simplifiability of similarity relations, we have $x_k \, S \, y_k$, which is a contradiction. Hence, we must have $|x_k| < |y_k|$. This in turn enables us to use the intrinsic (R,S)-stability over M again. We choose

$$\begin{cases} u &= y_1 \cdots y_{k-1}, \\ w &= y', \\ v &= y''y_{k+1} \cdots y_n, \end{cases} \quad \text{and} \quad \begin{cases} u' &= x_1 \cdots x_{k-1}, \\ w' &= x_k, \\ v' &= x_{k+1} \cdots x_m, \end{cases}$$

and use the intrinsic (R,S)-stability to conclude that $y' \in \overline{M}$. Since $|x_k| < |y_k|$, we have $y'' \neq \varepsilon$. Since $|y'| = |x_k|$ and $x_k \neq \varepsilon$ as an element of the base B, we also have $y' \neq \varepsilon$. Thus $y_k = y'y'' \in (\overline{M} \setminus \{\varepsilon\})^2$. This is impossible since y_k is an indecomposable element of the base B. Hence, $\overline{M}$ must be an inner (R,S)-ufe of M.

Conversely, let $\overline{M}$ be an inner (R,S)-ufe of M and let B be the base of $\overline{M}$. Furthermore, assume that words $u,v,w,u',v',w' \in \mathcal{A}^*$ satisfy the conditions (i) and (ii). Thus we may write $uw, v, u', w'v'$ as products of elements of the base B:

$$\begin{aligned} uw &= x_1 \cdots x_k, \\ v &= v_1 \cdots v_l, \\ u' &= u_1 \cdots u_m, \\ w'v' &= y_1 \cdots y_n. \end{aligned}$$

Since $u \,R\, u'$, $w \,R\, w'$ and $v \,R\, v'$, we have by the multiplicativity of similarity relations that

$$x_1 \cdots x_k v_1 \cdots v_l \,R\, u_1 \cdots u_m y_1 \cdots y_n.$$

Since $\overline{M}$ is an inner (R,S)-ufe of M and the words $x_1 \cdots x_k v_1 \cdots v_l$ and $u_1 \cdots u_m y_1 \cdots y_n$ belong to M, we conclude that $k+l = m+n$ and corresponding elements of both sides are S-compatible and furthermore of the same length. We have

$$u' = u_1 \cdots u_m \,S\, x_1 \cdots x_m = u \quad \text{and} \quad w = x_{m+1} \cdots x_k \in B^*.$$

In other words, $u, w \in \overline{M}$ and $u \,S\, u'$. Hence, $\overline{M}$ is intrinsically (R,S)-stable over M. □

Remark 4.2. In Definition 4.5, when characterizing the (R,S)-unique factorization extensions using (R,S)-stability, it is necessary that the assumptions (i) and (ii) imply that both u and w belong to $\overline{M}$. This is elucidated by the following example. Let $R = \langle (b,c) \rangle$ and $X = \{abcca, ac, cca\}$. Now the monoid X^* is not (R,R)-ufe of itself. However, excluding the requirement $u \in \overline{M}$ from the definition of (R,R)-stability would yield that X^* is (R,R)-stable. For example, substituting $u = ab$, $w = cca$, $v = ac$, $u' = ac$, $w' = cca$, $v' = ac$, we see that the above mentioned assumptions (i) and (ii) are satisfied and $w \in X^*$, but $u \notin \overline{M}$.

Theorem 4.14 gives as an easy consequence the following result concerning (R,S)-stable and (R,S)-free monoids. It is called here the *generalized Schützenberger's criterion*. Note that the usual formulation of Schützenberger's criterion for monoids (Theorem 2.3) follows easily by assigning $R = S = \iota$.

Corollary 4.5 (Generalized Schützenberger's criterion)**.** *A submonoid of $\mathcal{A}^*$ is (R,S)-free if and only if it is (R,S)-stable.*

Proof. By the definition of (R,S)-freeness, (R,S)-free submonoid $\overline{M}$ of $\mathcal{A}^*$ is an (R,S)-ufe of itself. This is possible if and only if $\overline{M}$ is (R,S)-stable (over itself) by the previous theorem. □

4.3 Relational Hulls

Using the stability results of the previous section it is easy to prove the following closure property of (R,S)-unique factorization extensions.

Theorem 4.15. *Let M be a submonoid of $\mathcal{A}^*$. Any intersection of inner (resp. outer) (R,S)-unique factorization extensions of M is an inner (resp. outer) (R,S)-unique factorization extension of M.*

Proof. We prove the claim for inner extensions. The proof for outer extensions is similar. Let $\overline{M}_i$ be an inner (R,S)-ufe of M for each $i \in \mathcal{I}$ and set $\overline{M} = \cap_{i\in\mathcal{I}}\overline{M}_i$. Clearly $\overline{M}$ is a monoid as an intersection of monoids. Moreover, it is nonempty, since the intersection contains M. Consider now words u, w, v, u', w', v' satisfying $u\,R\,u'$, $w\,R\,w'$ and $v\,R\,v'$. Assume that $uwv, u'w'v' \in M$ and $uw, v, u', w'v' \in \overline{M}$. By the definition of $\overline{M}$, this means that $uw, v, u', w'v' \in \overline{M}_i$ for all $i \in \mathcal{I}$. Since every $\overline{M}_i$ is an inner (R,S)-ufe of M, every $\overline{M}_i$ is intrinsically (R,S)-stable over M by Theorem 4.14. Hence, we have $u, w \in \overline{M}_i$ for all $i \in \mathcal{I}$ and $u\,S\,u'$. This means that $u, w \in \overline{M}$ and $u\,S\,u'$, i.e., $\overline{M}$ is intrinsically (R,S)-stable. Using Theorem 4.14 again, we conclude that $\overline{M}$ is an inner (R,S)-ufe of M. □

As a corollary of the previous theorem we get the following result concerning (R,S)-free monoids. It is called here the generalized Tilson's result. Note that Corollary 2.1 is obtained as a special case $R = S = \iota$.

Corollary 4.6 (Generalized Tilson's result)**.** *Any intersection of (R,S)-free submonoids of $\mathcal{A}^*$ is (R,S)-free.*

Proof. Let M_i be an (R,S)-free submonoid of $\mathcal{A}^*$ for each $i \in \mathcal{I}$. Clearly the intersection $M = \cap_{i\in\mathcal{I}}M_i$ is a monoid. By the definition of (R,S)-freeness, every M_i is an (R,S)-ufe of itself. Thus, every M_i is an inner (and outer) (R,S)-ufe of M. By Theorem 4.15, M is an (R,S)-ufe of itself and therefore (R,S)-free. □

Note that the previous result could have been proved also using Corollary 4.5. In that case the proof is similar to the proof of Theorem 4.15.

Let X be an arbitrary subset of $\mathcal{A}^*$. Consider now the set of all monoids which are inner (resp. outer) (R,S)-unique factorization extensions of X^*.

Note that this set may be empty as was already seen in Example 4.4. On the other hand, it follows from Theorem 4.15 that the set is closed under intersection. Thus, we may define the smallest extensions as follows.

Definition 4.6. Let $X \subseteq \mathcal{A}^*$. Suppose that there exists inner (R,S)-unique factorization extensions of X^*. Then the monoid

$$\widehat{I}_{R,S}(X) = \bigcap_{\substack{\text{M is an inner} \\ (R,S)\text{-ufe of } X^*}} M,$$

which is the smallest inner (R,S)-ufe of X^*, is called the *inner (R,S)-unique factorization hull* of X or, shortly, the *inner (R,S)-hull* of X. Similarly, the nonempty intersection of the outer (R,S)-unique factorization extensions of X is called the *outer (R,S)-unique factorization hull* or, shortly, the *outer (R,S)-hull* of X. It is denoted by $\widehat{O}_{R,S}(X)$.

Note that the hulls are defined for arbitrary sets of words while the unique factorization extensions are defined for monoids.

Remark 4.3. If X is a generating set of a monoid M, then $X^* = M = M^*$ and therefore $\widehat{I}_{R,S}(M) = \widehat{I}_{R,S}(X)$ and $\widehat{O}_{R,S}(M) = \widehat{O}_{R,S}(X)$.

The existence of the hulls depends on the relations R and S and the set X itself. By Corollary 4.4, there always exist inner and outer (R,S)-unique factorization extensions whenever $R \subseteq S$. Namely, in this case $\mathcal{A}^*$ is an inner and outer (R,S)-ufe of any of its submonoids. This especially means that $\widehat{I}_{R,R}(X)$ and $\widehat{O}_{R,R}(X)$ exist for any $X \subseteq \mathcal{A}^*$. For simplicity, we denote $\widehat{I}_{R,R}(X) = \widehat{I}_R(X)$ and $\widehat{O}_{R,R}(X) = \widehat{O}_R(X)$ in the sequel. These hulls are succinctly called the *weak inner* and the *weak outer R-hulls* of X respectively. Weak hulls play an important role among relational hulls as will be stated in the following theorem.

Theorem 4.16. *Let X be a subset of $\mathcal{A}^*$. The inner (R,S)-hull of X exists if and only if $R_{X^*} \subseteq S_{X^*}$, in which case $\widehat{I}_{R,S}(X) = \widehat{I}_R(X)$.*

Proof. Since the inner (R,R)-hull of an arbitrary set $X \subseteq \mathcal{A}^*$ always exists and the hull is an (R,R)-unique factorization extension of X, the condition $R_{X^*} \subseteq S_{X^*}$ is necessary and sufficient for the existence of the (R,S)-hull by Theorem 4.10.

Suppose now that an inner (R,S)-hull of a set X exists. By Theorem 4.10, every inner (R,S)-ufe of X is a weak R-ufe of X. Thus, the smallest weak R-ufe of X is contained in the intersection of all inner (R,S)-unique factorization extensions of X. In other words, $\widehat{I}_R(X) \subseteq \widehat{I}_{R,S}(X)$. Suppose that $\widehat{I}_R(X) \neq \widehat{I}_{R,S}(X)$. Now the (R,R)-ufe $\widehat{I}_R(X)$ is not an (R,S)-ufe of X. Hence, there exist $x, y \in X^*$ such that $(x,y) \in R \setminus S$. This contradicts the above mentioned condition on the inclusion of the relations. Thus, we must have $\widehat{I}_R(X) = \widehat{I}_{R,S}(X)$. □

Using similar considerations, we may also prove the corresponding result for outer (R, S)-hulls.

Theorem 4.17. *Let X be a subset of $\mathcal{A}^*$. The outer (R, S)-hull of X exists if and only if, for all $x \in X^*$, we have $R(x) \cap \widehat{O}_R(X) \subseteq S(x) \cap \widehat{O}_R(X)$, in which case $\widehat{O}_{R,S}(X) = \widehat{O}_R(X)$.*

Let us now turn to (R, S)-free monoids. If the set of (R, S)-free monoids containing $X \subseteq \mathcal{A}^*$ is not empty, we define that the (R, S)-*free hull* of X is

$$\widehat{F}_{R,S}(X) = \bigcap_{\substack{M \supseteq X, \\ \text{M is } (R,S)\text{-free}}} M.$$

The existence of this smallest (R, S)-free monoid containing X is based on the generalized Tilson's result (Corollary 4.6). As above we use a shorter notation $\widehat{F}_{R,R}(X) = \widehat{F}_R(X)$ for weak R-free hulls. For all sets $X \subseteq \mathcal{A}^*$, the weak R-free hull of X exists, since $\mathcal{A}^*$ is always (R, R)-free. Moreover, for (R, S)-free hulls we have a similar characterization as above.

Theorem 4.18. *Let X be a subset of $\mathcal{A}^*$. Let B be the base of the monoid $\widehat{F}_R(X)$. The (R, S)-free hull of X exists if and only if $R_B \subseteq S_B$, in which case $\widehat{F}_{R,S}(X) = \widehat{F}_R(X)$.*

The proof of this theorem is based on the characterization of (R, S)-free monoids in Theorem 4.13 and on similar considerations as in the proof of Theorem 4.16.

Let X be an arbitrary subset of $\mathcal{A}^*$. Clearly the outer (R, S)-hull of X is an inner (R, S)-ufe of X^*. Moreover, the (R, S)-free hull of X is an outer (R, S)-ufe of X^*. By the minimality of hulls, we therefore have

$$\widehat{I}_{R,S}(X) \subseteq \widehat{O}_{R,S}(X) \subseteq \widehat{F}_{R,S}(X). \tag{4.5}$$

Suppose further that Y is a set containing X. By the minimality of hulls, it is also clear that

$$\widehat{I}_{R,S}(X) \subseteq \widehat{I}_{R,S}(Y), \tag{4.6}$$
$$\widehat{O}_{R,S}(X) \subseteq \widehat{O}_{R,S}(Y), \tag{4.7}$$
$$\widehat{F}_{R,S}(X) \subseteq \widehat{F}_{R,S}(Y). \tag{4.8}$$

At the end of Section 4.3.1 we further consider inclusion properties of hulls.

4.3.1 Procedures

Next we consider a method of finding the hulls in practice. By the characterizations of the previous section, we can restrict our considerations to

finding weak R-hulls. If the weak hulls are (R,S)-hulls, this can be verified algorithmically by considering the inclusion of the relations R and S; see Theorems 4.16–4.18.

Let X be a finite subset of $\mathcal{A}^*$. In order to construct an inner (R,R)-unique factorization hull of X with base Y, we must block "nontrivial" relations in X^*. For this purpose, we say that a pair of words $(u,v) \in Y \times Y$ is an *inner R-match for Y over X* if u and v are at the same position in a relation on X^*, i.e., there exist words $x', x'', y', y'' \in Y^*$ such that

$$x'ux'', y'vy'' \in X^*,\ x'ux'' \,R\, y'vy'' \text{ and } |x'| = |y'|. \tag{4.9}$$

An *outer R-match for Y over X* is defined similarly, except that condition (4.9) is replaced by the weaker condition

$$(x'ux'' \in X^* \text{ or } y'vy'' \in X^*),\ x'ux'' \,R\, y'vy'' \text{ and } |x'| = |y'|, \tag{4.10}$$

where only one of the words $x'ux''$ and $y'vy''$ must belong to X^*. An inner or an outer R-match is called *nontrivial* if $(u,v) \notin R$. Otherwise, the pair is called *trivial*. Note that by the simplifiability of similarity relations, an R-match (u,v) is trivial if and only if $|u| = |v|$. Let us denote the set of nontrivial inner (resp. outer) R-matches for Y over X by $C^i_{R,X}(Y)$ (resp. $C^o_{R,X}(Y)$). Using these sets we can characterize (R,R)-unique factorization extensions of X^* in the following way.

Lemma 4.3. *Let X be a set of $\mathcal{A}^*$ and let B be the base of a monoid M such that $X^* \subseteq M$. The monoid M is an inner (resp. outer) (R,R)-ufe of X^* if and only if $C^i_{R,X}(B) = \emptyset$ (resp. $C^o_{R,X}(B) = \emptyset$).*

Proof. We give a proof for the inner (R,R)-unique factorization extensions. The proof for the outer case is similar. If M is an inner (R,R)-ufe of X, it is clear that $C^i_{R,X}(B)$ must be empty. Conversely, suppose that $C^i_{R,X}(B) = \emptyset$. Consider words $x_1, \ldots, x_m, y_1, \ldots, y_n \in B$ such that $x_1 \cdots x_m \,R\, y_1 \cdots y_n$ and $x_1 \cdots x_m, y_1 \cdots y_n \in X^*$. Since $C^i_{R,X}(B) = \emptyset$, we must have $x_1 \,R\, y_1$. This implies that $|x_1| = |y_1|$. Thus also $x_2 \,R\, y_2$, for otherwise, $C^i_{R,X}(B) \neq \emptyset$. Now $|x_1x_2| = |y_1y_2|$. Continuing similarly, we see that $x_i \,R\, y_i$ for all $i = 1, 2, \ldots, \min\{m,n\}$. By R-compatibility, $|x_1 \cdots x_m| = |y_1 \cdots y_n|$, which implies that $n = m$. Hence, M is an inner (R,R)-ufe of X^*. □

For the next procedure we need one more definition. For a word $u \in Y$ we define a set $D^i_{R,X}(u,Y)$: A word v belongs to $D^i_{R,X}(u,Y)$ if and only if $v = u$ or for some positive integer n there exist words $u = u_0, u_1, \ldots, u_{n-1}$, $u_n = v \in Y$ such that for $j = 0, 1, \ldots, n-1$ the pair (u_i, u_{i+1}) is a trivial inner R-match for Y over X. By requiring that (u_i, u_{i+1}) is only a trivial outer R-match, we obtain a set denoted by $D^o_{R,X}(u,Y)$. Let us now define the following iterative procedure similar to the procedures introduced in [48].

Procedure 4.2. INNERHULL $P_i(X, R)$

Let the input be a finite set $X \subseteq \mathcal{A}^*$ and a similarity relation R on $\mathcal{A}^*$. Set $X_0 = X \setminus \{\varepsilon\}$, and iterate for $j \geq 0$:

1. Choose an inner match $(u, v) \in C^i_{R,X}(X_j)$ such that $u = u'u''$, where $|u'| = |v|$ and $u'' \in \mathcal{A}^+$. If no such pair exists, then stop and return $P_i(X, R) = X_j$.
2. Set $R'(u) = \{\mathrm{pref}_{|u'|}(w) \mid w \in D^i_{R,X}(u, X_j)\}$ and set $R''(u) = \{\mathrm{suf}_{|u''|}(w) \mid w \in D^i_{R,X}(u, X_j)\}$.
3. Set $X_{j+1} = \Big(X_j \setminus D^i_{R,X}(u, X_j)\Big) \cup R'(u) \cup R''(u)$.

The output $P_i(X, R)$ is the base of the inner (R, R)-hull of X.

When a word $u = u'u'' \in X_j$ is replaced by two new words u' and u'' in X_{j+1}, this is called a *split* of u into u' and u''. Note that in each iteration step, at least one of the words in X_j is split into two proper factors, since $\varepsilon \notin X_j$ for any $j \geq 0$. For a finite set of words there are only finitely many factors, and therefore the procedure must terminate. Next we prove that Procedure 4.2 computes the base of the inner (R, R)-hull of X correctly.

Theorem 4.19. *Let X be a finite subset of $\mathcal{A}^*$. Procedure 4.2 with input X returns the base of the inner (R, R)-hull of X, i.e., $P_i(X, R) = \mathrm{Base}(\widehat{I}_R(X))$.*

Proof. As mentioned above, Procedure 4.2 always terminates on any finite input $X \subseteq \mathcal{A}^*$. Suppose now that the procedure terminates after k iterations. Let us first show by induction that

$$X_j^* \subseteq \widehat{I}_R(X)$$

for all $j = 0, 1, \ldots, k$. The case $j = 0$ is clear by the definition of $\widehat{I}_R(X)$. Suppose now that $X_j^* \subseteq \widehat{I}_R(X)$ and $(u, v) \in C^i_{R,X}(X_j)$. We claim that

$$R'(u) \cup R''(u) \subseteq \widehat{I}_R(X).$$

Consider a word $w \in D^i_{R,X}(u, X_j)$. Assume first that $w = u$. We prove that u' and u'' belong to $\widehat{I}_R(X)$. By the intrinsic (R, R)-stability of $\widehat{I}_R(X)$ over X^*, by (4.9) and by the assumption $X_j^* \subseteq \widehat{I}_R(X)$, the words u, w, v, u', w', v' in Figure 4.3 of Section 4.2.2 can be replaced by the words $x'u', u''$, $x'', y'v, \mathrm{pref}_{|u''|}(y'')$ and $\mathrm{suf}_{|y''|-|u''|}(y'')$, respectively. Since $x'u$, x'', $y'v$, $y'' \in X_j^* \subseteq \widehat{I}_R(X)$, the intrinsic (R, R)-stability of $\widehat{I}_R(X)$ implies

$$x'u', u'' \in \widehat{I}_R(X).$$

Similarly, replacing the words of Figure 4.3 by the words $x', u', u''x'', y', v, y''$ we see that

$$x', u' \in \widehat{I}_R(X),$$

since $x'u', u''x'', y', vy'' \in \widehat{I}_R(X)$. Hence, we have u', $u'' \in \widehat{I}_R(X)$.

Suppose then that $w \in D^i_{R,X}(u, X_j)\backslash\{u\}$ and, for some positive integer n, there exist words $u = u_0, u_1, \ldots, u_{n-1}, u_n = w \in X_j$ such that the pairs (u_i, u_{i+1}) are trivial inner R-matches for X_j over X. Furthermore, assume that the words $u'_i = \mathrm{pref}_{|u'|}(u_i)$ and $u''_i = \mathrm{suf}_{|u''|}(u_i)$ belong to $\widehat{I}_R(X)$ for $i = 0, 1, \ldots, n-1$. We show that also $u'_n = \mathrm{pref}_{|u'|}(u_n)$ and $u''_n = \mathrm{suf}_{|u''|}(u_n)$ belong to $\widehat{I}_R(X)$. We use the intrinsic (R,R)-stability of $\widehat{I}_R(X)$ as above. In this case, the words u, w, v, u', w', v' in Figure 4.3 are replaced by the words $y'u'_n, u''_n, y'', x'u'_{n-1}, u''_{n-1}, x''$, respectively. Since $y'u_n$, y'', $x'u'_{n-1}$ and $u''_{n-1}x''$ belong to $\widehat{I}_R(X)$, we have $y'u'_n, u''_n \in \widehat{I}_R(X)$. Note that we used the fact that $|u_{n-1}| = |u_n|$. Replacing the words of Figure 4.3 by the words $y', u'_n, u''_n y'', x', u'_{n-1}, u''_{n-1}x''$, we conclude that $y', u'_n \in \widehat{I}_R(X)$, since $y'u'_n, u''_n y'', x', u_{n-1}x'' \in \widehat{I}_R(X)$. We have thus proved that $u'_n, u''_n \in \widehat{I}_R(X)$. Furthermore, this means that $R'(u) \cup R''(u) \subseteq \widehat{I}_R(X)$. Thus, in step 3 of Procedure 4.2 we modify X_j in such a way that we add only elements which must belong to the inner (R,R)-hull of X and we do not delete any essential elements. Namely, $X \subseteq X^*_j \subseteq X^*_{j+1}$, since $D^i_{R,X}(u, X_j) \subseteq R'(u)R''(u)$. Therefore, $X^*_{j+1} \subseteq \widehat{I}_R(X)$.

Since $C^i_{R,X}(X_k) = \emptyset$, the set X_k consists only of the indecomposable elements of X^*_k. Namely, consider words $x, x', x'' \in X_k$ such that $x = x'x''$. Since every $x \in X_k$ is a factor of some word in X^*, we have $(x, x') \in C^i_{R,X}(X_k)$, which is a contradiction. Thus, X_k is the base of X^*_k. Moreover, by Lemma 4.3, the monoid X^*_k is an inner (R,R)-ufe of X^*. Hence, $X \subseteq X^*_k \subseteq \widehat{I}_R(X)$ and the minimality of the inner (R,R)-hull of X implies that $X^*_k = \widehat{I}_R(X)$. Thus, $P_i(X,R) = X_k = \mathrm{Base}(\widehat{I}_R(X))$. □

The procedure for finding the base of the outer (R,R)-hull of X is very similar to Procedure 4.2. It is obtained by replacing $C^i_{R,X}(X_j)$ by $C^o_{R,X}(X_j)$ and $D^i_{R,X}(u, X_j)$ by $D^o_{R,X}(u, X_j)$. We denote this procedure for outer hulls by $P_o(X,R)$. Modifying slightly the previous proof, it is easy to see that $P_o(X,R)$ returns the base of $\widehat{O}_R(X)$.

We may also use Procedure 4.2 to obtain the (R,R)-free hull of X. Let us define that $\widehat{I}^0_R(X) = X$ and

$$\widehat{I}^j_R(X) = \widehat{I}_R(\widehat{I}^{j-1}_R(X))$$

for all integers $j > 0$. The notation $\widehat{O}^j_R(X)$ is defined similarly. Now we have the following result.

Theorem 4.20. *Let X be a subset of $\mathcal{A}^*$. Then for all $j \geq 0$ we have*

$$\widehat{I}_R^j(X) \subseteq \widehat{O}_R^j(X) \subseteq \widehat{F}_R(X).$$

Moreover, for a finite X, there exists $k \geq 0$ such that $\widehat{I}_R^k(X) = \widehat{I}_R^{k+1}(X)$, in which case

$$\widehat{I}_R^k(X) = \widehat{O}_R^k(X) = \widehat{F}_R(X).$$

Proof. For $j = 0$, the claim $\widehat{I}_R^0(X) = \widehat{O}_R^0(X) = X \subseteq \widehat{F}_R(X)$ is clear. Suppose then that $\widehat{I}_R^j(X) \subseteq \widehat{O}_R^j(X) \subseteq \widehat{F}_R(X)$ for some integer j. Using properties (4.5)–(4.7) of the previous section, we now have

$$\widehat{I}_R^{j+1}(X) \overset{(4.6)}{\subseteq} \widehat{I}_R(\widehat{O}_R^j(X)) \overset{(4.5)}{\subseteq} \widehat{O}_R^{j+1}(X) \overset{(4.7)}{\subseteq} \widehat{O}_R(\widehat{F}_R(X)) \overset{(4.5)}{\subseteq} \widehat{F}_R(\widehat{F}_R(X)).$$

Since $\widehat{F}_R(X)$ is an (R,R)-ufe over itself, we have $\widehat{F}_R(\widehat{F}_R(X)) = \widehat{F}_R(X)$. Thus, the first claim is proved.

For the second claim, let X be finite. By Procedure 4.2, the base of $\widehat{I}_R(X)$ contains only factors of X. Since, by Remark 4.3, $\widehat{I}_R(\widehat{I}_R^j(X)) = \widehat{I}_R(\mathrm{Base}(\widehat{I}_R^j(X)))$ for $j \geq 1$, we inductively conclude that the base of $\widehat{I}_R^j(X)$ is a set of factors of X. For a finite set X, there exist only finitely many factors. Hence, we must have $\widehat{I}_R^{k+1}(X) = \widehat{I}_R^k(X)$ for some k. But this means that $\widehat{I}_R^k(X)$ is an inner (R,R)-ufe of itself. Thus, it is (R,R)-free. Since $\widehat{I}_R^k(X) \subseteq \widehat{O}_R^k(X) \subseteq \widehat{F}_R(X)$, we must have $\widehat{I}_R^k(X) = \widehat{O}_R^k(X) = \widehat{F}_R(X)$ by the minimality of the (R,R)-free hull $\widehat{F}_R(X)$. □

The previous theorem implies that we can use the following iterative procedure for finding the base of the weak R-free hull of a finite set of words.

Procedure 4.3. FreeHull $P_f(X,R)$

Let the input be a finite set $X \subseteq \mathcal{A}^*$ and a similarity relation R on $\mathcal{A}^*$. Set $X_0 = X$, and iterate for $j \geq 0$:

1. Set $X_{j+1} = P_i(X_j, R)$.
2. If $X_j = X_{j+1}$, then stop and return $P_f(X,R) = X_j$.

The output $P_f(X,R)$ is the base of the (R,R)-free hull of X.

Note that this procedure is based on iterative calculation of inner (R,R)-hulls though by Theorem 4.20 we could as well use an algorithm which counts outer (R,R)-hulls iteratively. Next we will give some examples of these procedures and hulls. In the first example, the inner (R,R)-hull $\widehat{I}_R(X)$ is a proper subset of the outer (R,R)-hull of X. More precisely,

$$\widehat{I}_R(X) \subsetneq \widehat{O}_R(X) = \widehat{F}_R(X).$$

Example 4.5. Let $\mathcal{A} = \{a, b, c, d\}$ and consider the set $X = \{a, ac, dd, ddb\}$ and the similarity relation $R = \langle (a, b), (b, c), (c, d) \rangle$. Table 4.1 shows the intermediate sets X_j of Procedure 4.3. Note that the third column is $C^i_{R,X_j}(X_j)$ instead of $C^i_{R,X}(X_j)$. The relations in the rightmost column justify the structure of $C^i_{R,X_j}(X_j)$. For the sake of clarity, words of X_j in the relations are separated by dots.

j	X_j	$C^i_{R,X_j}(X_j)$	reasoning
0	$\{a, ac, dd, ddb\}$	$\{(dd, ddb)\}$	$dd \cdot a \, R \, ddb$
1	$\{a, ac, b, dd\}$	$\{(a, ac), (b, ac)\}$	$a \cdot b \, R \, ac \, R \, b \cdot b$
2	$\{a, b, c, dd\}$	$\{(c, dd)\}$	$c \cdot c \, R \, dd$
3	$\{a, b, c, d\}$	$\emptyset$	

Table 4.1: Calculations for the (R, R)-free hull of Example 4.5.

In each iteration step of $P_f(X, R)$ we use Procedure 4.2. In other words, $X_j = \mathrm{Base}(\widehat{I}_R(X_{j-1}))$. Note that $C^o_{R,X}(X_1) = \{(a, ac), (b, ac)\}$, whereas $C^i_{R,X}(X_1) = \emptyset$. Namely in $a \cdot b \, R \, ac \, R \, b \cdot b$ we have $ac \in X^*$, but ab and bb belong to $X_1^* \backslash X^*$. Furthermore, $C^o_{R,X}(X_2) = \{(c, dd)\}$ because of $c \cdot c \, R \, dd$ and $dd \in X^*$. Finally we get $X_3 = \{a, b, c, d\}$, which is the base of the weak R-free hull of X. Clearly, $C^o_{R,X}(X_3, R) = \emptyset$ and therefore $X_3 = \mathrm{Base}(\widehat{O}_R(X))$. Hence, $\widehat{I}_R(X) = X_1^* \subsetneq X_3^* = \widehat{O}_R(X) = \widehat{F}_R(X)$.

Next we show that the outer (R, R)-hull $\widehat{O}_R(X)$ can be a proper subset of the (R, R)-free hull of X, i.e., $\widehat{I}_R(X) = \widehat{O}_R(X) \subsetneq \widehat{F}_R(X)$.

Example 4.6. Consider the alphabet $\mathcal{A} = \{e, f, g, h, i\}$ and the set $X = \{eee, fffi, ggi, hh, i\}$ with the similarity relation $R = \langle (e, f), (f, g), (g, h) \rangle$. Table 4.2 simulates the use of Procedure 4.2 in the calculation of $\widehat{F}_R(X)$ by Procedure 4.3. Note that for $X = \{eee, fff, ggi, hh, i\}$ we have $C^i_{R,X}(Y_2) = C^o_{R,X}(Y_2) = \emptyset$. Hence,

$$\mathrm{Base}(\widehat{I}_R(X)) = \mathrm{Base}(\widehat{O}_R(X)) = Y_2.$$

The set $C^o_{R,Y_2}(Z_2)$ is clearly empty and therefore $Z_2^* = \widehat{I}_R(Y_2) = \widehat{O}_R(Y_2)$. Note that the split of the factor fff induces the split of eee in Z_0. Similarly the factors ff, gg and hh split in Z_1, since they belong to $D^i_{R,Y_2}(ee, Z_1)$ because of the following relations on the monoid Y_2^*:

$$ee \cdot e \, R \, ff \cdot f, \quad ff \cdot f \cdot gg \, R \, gg \cdot f \cdot ff \quad \text{and} \quad gg \, R \, hh.$$

Since Z_2 is an inner (R, R)-ufe of itself, it is (R, R)-free. Moreover, $\widehat{I}^2_R(X) = Z_2^* = \widehat{I}^3_R(X)$. Thus, by Theorem 4.20, we have

$$\widehat{I}^2_R(X) = \widehat{O}^2_R(X) = \widehat{F}_R(X).$$

X	$P^i(X,R)$
$\{eee, fffi, ggi, hh, i\}$	$Y_0 = X$
	$(u,v) = (fffi, eee)$ since $fffi\,R\,eee \cdot i$
	$D^i_{R,X}(u,Y_0) = \{fffi\}$
	$Y_1 = \{eee, fff, ggi, hh, i\}$
	$(u,v) = (ggi, hh)$ since $ggi\,R\,hh \cdot i$
	$D^i_{R,X}(u,Y_1) = \{ggi\}$
	$Y_2 = \{eee, fff, gg, hh, i\}$
$\{eee, fff, gg, hh, i\}$	$Z_0 = X$
	$(u,v) = (fff, gg)$ since $fff \cdot gg\,R\,gg \cdot fff$
	$D^i_{R,X}(u,Z_0) = \{eee, fff\}$
	$Z_1 = \{e, ee, f, ff, gg, hh, i\}$
	$(u,v) = (ee, e)$ since $ee \cdot e\,R\,e \cdot ee$
	$D^i_{R,X}(u,Z_1) = \{ee, ff, gg, hh\}$
	$Z_2 = \{e, f, g, h, i\}$

Table 4.2: Calculations for the (R,R)-free hull of Example 4.6.

We may now combine the previous two examples to verify that it is possible to have

$$\widehat{I}_R(X) \subsetneq \widehat{O}_R(X) \subsetneq \widehat{F}_R(X).$$

Example 4.7. Consider a set $X = \{a, ac, dd, ddb, eee, fffi, ggi, hh, i\}$ in a nine letter alphabet and let $R = \langle (a,b),(b,c),(c,d),(e,f),(f,g),(g,h)\rangle$. Since the alphabets and the relations in Examples 4.5 and 4.6 are independent, we may deduce from the previous calculations that

$$\begin{aligned}
\mathrm{Base}(\widehat{I}_R(X)) &= \{a, ac, b, dd, eee, fff, gg, hh, i\},\\
\mathrm{Base}(\widehat{O}_R(X)) &= \{a, b, c, d, eee, fff, gg, hh, i\},\\
\mathrm{Base}(\widehat{F}_R(X)) &= \{a, b, c, d, e, f, g, h, i\}.
\end{aligned}$$

Observe that iterating Procedure 4.2 with input X and R sufficiently many times, we do not necessarily get the outer (R,R)-hull of X. More precisely, arbitrary iterations of inner and outer hulls may not be included in each other. Namely, in our example we have

$$\widehat{I}^2_R(X) \setminus \widehat{O}_R(X) \neq \emptyset \quad \text{and} \quad \widehat{O}_R(X) \setminus \widehat{I}^2_R(X) \neq \emptyset,$$

since the base of $\widehat{I}^2_R(X)$ is $\{a, b, c, dd, e, f, g, h, i\}$.

Finally we note that the algorithms for finding inner, outer and free (R,S)-hulls can be implemented using generalized Spehner's graphs; see Section 4.5.

4.4 Defect Effect

Recall the famous defect theorem (Theorem 2.4). In this section we consider a defect effect with respect to a similarity relation R. Note that the original defect theorem does not hold in the relational case, i.e., the cardinality of $\mathrm{Base}(\widehat{F}_R(X))$ can be greater than the cardinality of X. For example, $\mathrm{Base}(\widehat{F}_R(X)) = \{ab, ac, bc\}$ for $X = \{abac, bc\}$ and $R = \langle (a,b),(b,c)\rangle$. Hence, we need a different kind of formulation. Let X be a finite subset of $\mathcal{A}^*$ and let R be a similarity relation. Let us consider a graph $G_R(X) = (V, E)$ defined as follows. The set V of vertices is X, and $(u, v) \in E$ if and only if $u\,R\,v$. We consider the connected components of G. Note that the set of vertices in the connected component containing x is exactly $(R_X)^+(x)$, where R^+ is the transitive closure of R. Denote the number of connected components of $G_R(X)$ by $c(X, R)$. We replace the cardinalities of the original defect theorem by the number of connected components and prove a defect theorem for inner (R, R)-hulls in the following form.

Theorem 4.21. *Let X be a finite subset of $\mathcal{A}^*$ and let B be the base of the inner (R, R)-hull of X. Then $c(B, R) \leq c(X, R)$, and the equality holds if and only if X is an (R, R)-code.*

Proof. If X is an (R, R)-code, then X^* is an inner (R, R)-ufe of itself and $B = X$ by Theorem 4.11. Thus the equality holds trivially. Suppose now that X is not an (R, R)-code. Hence, there exist words $x_1, \ldots, x_m, y_1, \ldots, y_n \in X$ such that $x_1 \cdots x_m\, R\, y_1 \cdots y_n$ and, for some $t \in \{1, 2, \ldots, \min\{n, m\}\}$, we have $x_s\, R\, y_s$ for $s = 1, 2, \ldots, t-1$, but $(x_t, y_t) \notin R$. Thus, (x_t, y_t) is a nontrivial inner R-match for X over X and $C^i_{R,X}(X_0) \neq \emptyset$ in Procedure 4.2. This means that the procedure does not return X_0, i.e., $P_i(X, R) = X_k$ for some positive integer k. By Theorem 4.19, Procedure 4.2 computes the base of the inner (R, R)-hull of X correctly. We show that $c(X_k, R) < c(X, R)$.

First we introduce some notation. The vertex set of $G_R(X_j)$ is denoted by V_j and the set of edges is denoted by E_j. An edge $(u, v) \in E_j$ is called a *light edge* if (u, v) is a (trivial) inner R-match for X_j over X. Otherwise, the edge is called *heavy*. Let us denote the set of vertices in the connected component of $G_R(X_j)$ containing a vertex u by $V_j(u)$. If there exists a path from u to v using only light edges, we denote $u \to_{L_j} v$. The partition of the vertices of $G_R(X_j)$ into *light components* $L_j(u) = \{v \in V_j(u) \mid u \to_{L_j} v\}$ is clearly a refinement of the partition of vertices into connected components. We note that $L_j(u)$ coincides with the set $D^i_{R,X}(u, X_j)$ by the definition. Figure 4.4 illustrates the edges and components of the graph $G_R(X_j)$.

For simplicity, in this proof we denote $c_j = c(X_j, R)$. Furthermore, the number of light components of $G_R(X_j)$ is denoted by l_j. Our proof is divided into two parts. First we prove that after each iteration step of Procedure 4.2 the number of light components of $G_R(X_j)$ cannot be greater

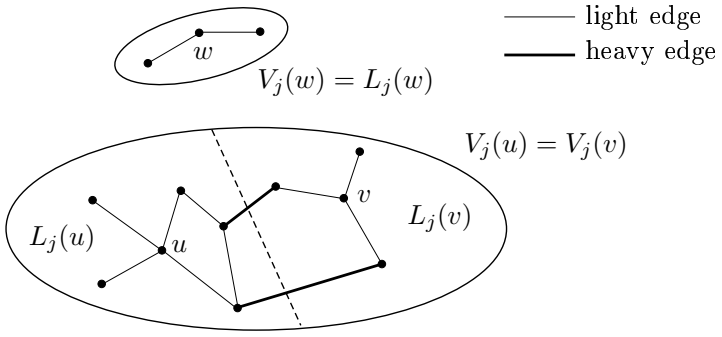


Figure 4.4: Components of the graph $G_R(X_j)$

than the number of the original connected components of $G_R(X)$. In other words,

$$l_j \leq c_0 \tag{4.11}$$

for every j satisfying $0 \leq j \leq k$.

Consider first the case $j = 0$. Let (u, v) be an arbitrary edge in E_0. The R-compatible words u and v of the vertex set V_0 belong to $X_0 = X$ and they form a trivial inner R-match for X_0 over X. Thus, there are no heavy edges in the graph $G_R(X)$ and therefore every connected component consists of exactly one light component. Hence, $l_0 = c_0$.

Suppose now that $l_j \leq c_0$. We will prove that $l_{j+1} \leq c_0$. Assume that $(u, v) \in C^i_{R,X}(X_j)$ and let $u = u'u''$, where $|u'| = |v|$ and $u'' \in \mathcal{A}^+$ as in Procedure 4.2. When the word u is split in the jth iteration step of Procedure 4.2, then all elements of $L_j(u) = D^i_{R,X}(u, X_j)$ split into two parts. In other words, the set $L_j(u)$ disappears and two new sets of vertices $R'(u)$ and $R''(u)$ are born. By the construction and the simplification rule, we know that $R'(u) \subseteq L_{j+1}(u')$ and $R''(u) \subseteq L_{j+1}(u'')$. In addition, it follows from $u' \, R \, v$ that $u' \rightarrow_{L_j} v$. Hence, we have $L_{j+1}(u') = L_{j+1}(v)$. Thus, the new vertices $R'(u)$ are connected to the old component containing the vertex v. To conclude, the light components $L_j(u)$ and $L_j(v)$ are changed to components $L_{j+1}(u'')$ and $L_{j+1}(v)$. Furthermore, by Procedure 4.2, we do not delete any edges inside any other light component than $L_j(u)$. Hence, these other light components cannot split into smaller light components. They may only become connected to other components. Therefore, the number of light components cannot increase in any iteration step, i.e., $l_{j+1} \leq l_j$. Hence, (4.11) follows by induction.

An example of a deletion of a light component $L_j(u)$ is given in Figure 4.5. The component $V_j(u)$ contains five light components, which form three new connected components $V_{j+1}(u_1)$, $V_{j+1}(u_2)$ and $V_{j+1}(u_3)$. The new

vertices $R'(u)$ and $R''(u)$ are connected to the old component $V_j(v)$. Note that, by the new light and heavy edges, the connected component $V_{j+1}(v)$ contains the sets $R'(u)$, $R''(u)$ and $V_j(w)$.

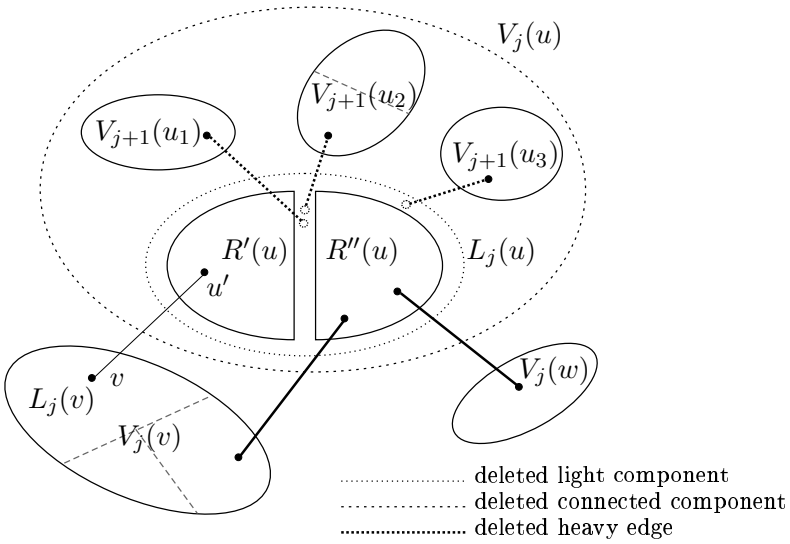


Figure 4.5: Split of u in $G_R(X_j)$.

It remains to show that in the last iteration round of the procedure the number of the light components strictly decreases and we get

$$l_k < c_0. \tag{4.12}$$

Assume now that in Procedure 4.2 we choose $(u, v) \in C^i_{R,X}(X_{k-1})$, where $u = u'u''$, $|u'| = |v|$ and $u'' \in \mathcal{A}^+$. More precisely, suppose that there exist $x', x'', y', y'' \in X^*_{k-1}$ such that $x'ux'', y'vy'' \in X^*$, $x'ux'' \,R\, y'vy''$ and $|x'| = |y'|$. Denote $y'' = y_1 \cdots y_n$, where $y_i \in X_{k-1}$ for all $i = 1, 2, \ldots, n$. By the above considerations, the set $R'(u)$ is connected to the old light component $L_{k-1}(v)$. Hence, let us consider the new vertices $R''(u)$ and the new light component $L_k(u'')$.

Assume first that $u'' \,R\, y_1$. This means that $L_k(u'') = L_k(y_1)$. Since the new component $L_k(u'')$ is now connected to the old component $L_{k-1}(y_1)$, this causes a decrease by one to the number of light connected components. Hence, $l_k < l_{k-1} \leq c_0$.

Suppose next that $(u'', y_1) \notin R$. If $y_1 \notin L_{k-1}(u)$, then y_1 is not split in the final iteration step and $y_1 \in X_k$. Hence, the pair (u'', y_1) is a nontrivial inner R-match for X_k over X by the relation

$$x'u'u''x'' \,R\, y'vy_1 \cdots y_n,$$

where $|x'u'| = |y'v|$. Therefore $C^i_{R,X}(X_k) \neq \emptyset$ and X_k is not the final outcome of the procedure $P_i(X, R)$; a contradiction.

Thus, we must have $y_1 \in L_{k-1}(u)$. We may denote $y_1 = y'_1 y''_1$, where $|y'_1| = |u'|$, $|y''_1| = |u''|$ and $y'_1 \in L_k(u')$. If $(u'', y'_1) \notin R$, then it is a nontrivial inner R-match for X_k over X by the relation

$$x'u'u''x'' \, R \, y'vy'_1y''_1y_2 \cdots y_n,$$

where $|x'u'| = |y'v|$. This is again impossible, since $C^i_{R,X}(X_k)$ must be empty. Thus $u'' \, R \, y'_1$ and (u'', y'_1) is a trivial inner R-match for X_k over X. Consequently, we have

$$u'' \rightarrow_{L_k} y'_1 \rightarrow_{L_k} u' \rightarrow_{L_k} v.$$

Hence, besides the new component $L_k(u')$, the component $L_k(u'')$ is connected to the old component $L_{k-1}(v)$, i.e., $L_k(u') = L_k(u'') = L_k(v)$, which causes a reduction in the number of light connected components. We conclude that (4.12) holds.

Inequality (4.12) implies $c_j < c_0$, since clearly the number of light components is greater than the number of connected components. This proves the defect effect for inner (R, R)-hulls. □

As a corollary, we also get the defect effect for the inner (R, S)-hulls.

Corollary 4.7. *Suppose that $\widehat{I}_{R,S}(X)$ exists and let B be the base of the inner (R, S)-hull of X. Then $c(B, R) \leq c(X, R)$, and the equality holds if and only if X is an (R, S)-code.*

Proof. This follows from the previous theorem and Theorem 4.16. Namely, if B is the base of the (R, S)-free hull of X, then it is the base of the (R, R)-free hull and $c(B, R) \leq c(X, R)$. As above, the equality holds if and only if X is an (R, S)-code. □

For the outer (R, S)-hull of X we have a similar defect effect. This can be proved by modifying the two previous proofs by replacing inner objects, e.g., $\widehat{I}_R(X)$, $C^i_{R,X}(X_j)$ and $D^i_{R,X}(u, X_j)$ by outer objects, e.g., $\widehat{O}_R(X)$, $C^o_{R,X}(X_j)$ and $D^o_{R,X}(u, X_j)$.

Theorem 4.22. *Suppose that $\widehat{O}_{R,S}(X)$ exists and let B be the base of the outer (R, S)-hull of X. Then $c(B, R) \leq c(X, R)$, and the equality holds if and only if X is an (R, S)-code.*

Using Procedure 4.3 it is easy to see that the defect effect of inner (R, S)-hulls produces a cumulative defect effect for (R, S)-free hulls. Recall from Remark 4.3 that $\widehat{I}^k_R(X) = \widehat{I}^k_R(X^*)$ for $k \geq 1$. Therefore, X^* is needed in the following corollary only in the initialization $k = 0$.

Corollary 4.8. *Let $X \subseteq \mathcal{A}^*$ be finite. Suppose that $\widehat{F}_{R,S}(X)$ exists and let B be its base. Let $k \geq 0$ be the smallest index such that $\widehat{I}_R^{k+1}(X^*) = \widehat{I}_R^k(X^*)$. Then*

$$c(B,R) \leq c(X,R) - k.$$

Moreover, $c(B,R) = c(X,R)$ if and only if X is an (R,S)-code.

Proof. Suppose first that X is an (R,S)-code. Then by Theorem 4.11, X^* is (R,S)-free and X is its base. Hence, $\widehat{F}_{R,S}(X) = X^*$ and the claim $c(B,R) = c(X,R)$ follows trivially. Since X^* is (R,S)-free, it means that $\widehat{I}_R(X^*) = X^* = \widehat{I}_R^0(X^*)$. Therefore, $k = 0$ in this case.

Suppose then that X is not an (R,S)-code. Since $\widehat{F}_{R,S}(X)$ exists, we have $R_B \subseteq S_B$ by Theorem 4.18. Consequently, $R_X \subseteq S_X$, which implies that X is not an (R,R)-code by Theorem 4.3. By Theorem 4.19, we have $X_{j+1} = P_i(X_j, R) = \mathrm{Base}(\widehat{I}_R(X_j))$ in the iterative calculation of $P_f(X,R)$ in Procedure 4.3. Recalling Remark 4.3, we conclude that $P_f(X,R)$ stops after $k+1$ iterations by our assumption. Since X is not an (R,R)-code, it follows that $P_i(X,R) = \mathrm{Base}(\widehat{I}_R(X)) \neq X$ by Theorem 4.21. Hence, the integer k must be positive. Moreover, in each of the first k iteration rounds, we have a defect effect by Theorem 4.21. Since $\widehat{F}_{R,S}(X) = \widehat{F}_R(X)$ by Theorem 4.18, we must have $B = \mathrm{Base}(\widehat{F}_R(X)) = P_f(X,R)$, which is the base of $\widehat{I}_R^k(X)$. Therefore, $c(B,R) \leq c(X,R) - k$ and $c(B,R) \neq c(X,R)$, since $k > 0$. □

Finally, we consider an application of these defect theorems. Recall from Section 4.1.1 that pcodes correspond to $(R_\uparrow, \iota)$-codes. Thus the previous defect theorems imply a defect theorem for partial words; see [24]. Naturally, we say that a monoid on partial words is *pfree* if and only if it is generated by a pcode. The *pfree hull* of a monoid X of partial words is the smallest pfree monoid containing X. Using our notation this means that pfree monoids are $(R_\uparrow, \iota)$-free and the pfree hull of X is the $(R_\uparrow, \iota)$-free hull of X. Now we state:

Corollary 4.9. *Let X be a finite set of partial words, i.e., a set of words over the alphabet $\mathcal{A}_\diamond$. Suppose that the pfree hull of X exists and let B be its base. Then $|B| \leq |X|$, and the equality holds if and only if X is a pcode.*

Proof. As mentioned above, the pfree hull is the $(R_\uparrow, \iota)$-free hull of X. Thus, by Corollary 4.8, we have $c(B, R_\uparrow) \leq c(X, R_\uparrow)$ and the equality holds if and only if X is an $(R_\uparrow, \iota)$-code. Since $M = B^*$ is an $(R_\uparrow, \iota)$-free monoid, we have $(R_\uparrow)_M \subseteq \iota_M$. This means that every connected component of $G_{R_\uparrow}(B)$ and $G_{R_\uparrow}(X)$ contains only one element. Thus $c(B, R_\uparrow) = |B|$ and $c(X, R_\uparrow) = |X|$. This proves our claim. □

Naturally, all previous defect theorems for words with similarity relations can also be formulated for partial words. For example, we get a cumulative defect effect by using inner $(R_\uparrow, \iota)$-hulls and the procedure $P_f(X, R_\uparrow)$.

4.5 Spehner Graphs

In many algorithms related to unique factorization in monoids, the graphs introduced by Spehner [73] are very useful. In this section we introduce generalized Spehner graphs in order to deal with problems concerning relationally unique factorizations. These graphs are practical tools in finding relational hulls (Procedure 4.2 and Procedure 4.3) and in the implementation of the Sardinas–Patterson procedure (Procedure 4.1). We give pseudocode algorithms for these tasks and simulate their use by some examples.

Let $X_1, \ldots, X_n$ be subsets of $\mathcal{A}^*$ and let $\mathcal{R}$ be a set of similarity relations R_{ij} on $\mathcal{A}^*$, where $1 \leq i < j \leq n$. The *generalized Spehner graph* $S_{\mathcal{R}}(X_1, \ldots, X_n)$ is defined as follows: The vertices of the graph are n-tuples $(u_1, \ldots, u_n)$, where each element u_i is a prefix of some word in X_i. The graph contains all vertices accessible from the (initial) vertex $(\varepsilon, \ldots, \varepsilon)$ using labelled directed edges:

$$(u_1, u_2, \ldots, u_n) \xrightarrow{(x_1, x_2, \ldots, x_n)} (v_1, v_2, \ldots, v_n),$$

where the elements of the label-tuple are words x_i belonging to $X_i \cup \{\varepsilon\}$ such that $x_i \in X_i$ if and only if $u_i = \varepsilon$. In addition, for $i < j$ we require that

$$\operatorname{pref}_{k_{ij}}(u_i x_i)\, R_{ij}\, \operatorname{pref}_{k_{ij}}(u_j x_j), \tag{4.13}$$

where $k_{ij} = \min\{|u_i x_i|, |u_j x_j|\}$ and

$$v_i = \operatorname{pref}_k(u_i x_i)^{-1}(u_i x_i), \tag{4.14}$$

where $k = \min\{|u_l x_l| \mid l = 1, 2, \ldots, n\}$. Thus, the head vertex $(v_1, v_2, \ldots, v_n)$ is completely determined by the tail vertex $(u_1, u_2, \ldots, u_n)$ and the label of the edge $(x_1, x_2, \ldots, x_n)$. We say that the element v_i is an *overflow* if it is not the empty word. Hence, in the initial vertex there are no overflows. Note that *loops*, i.e., edges from a vertex back to itself are allowed. As an example, we represent the Spehner graph $S_{\mathcal{R}}(X, Y)$ for $X = \{abda, ac\}$, $Y = \{ad, baa, c\}$ and $R_{12} = \langle (b, d) \rangle$ in Figure 4.6, where the label tuples are represented as column vectors for the sake of clarity.

Intuitively, the idea is that a Spehner graph $S_{\mathcal{R}}(X_1, \ldots, X_n)$ simulates a machine with n tapes. This multi-tape machine writes on the ith tape words of X_i catenating them to the end of the previously written word. In the beginning each tape is empty and the writing heads are in parallel positions. The machine can write only to those tapes where the length of the content is smallest. In addition, between each pair of tapes there is a special relation which restricts the words that can be written. For the contents of tapes i and j, we require that the shorter word is R_{ij}-related to the prefix of the longer word. The edges of the Spehner graph describe the words of X_i

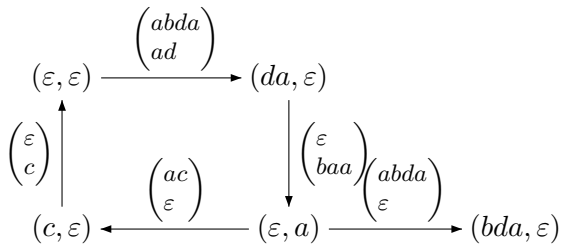


Figure 4.6: The Spehner graph $S_{\mathcal{R}}(X,Y)$ for $X = \{abda, ac\}$, $Y = \{ad, baa, c\}$ and $\mathcal{R} = \{R_{12}\} = \{\langle (b,d) \rangle\}$.

written on the n tapes in each step. The vertices of the Spehner graph consist of overflows. The overflow v_i indicates the suffix of the content on tape i obtained by deleting the prefix of length equal to the length of the shortest content on all tapes. The multi-tape machine with three tapes is illustrated in Figure 4.7.

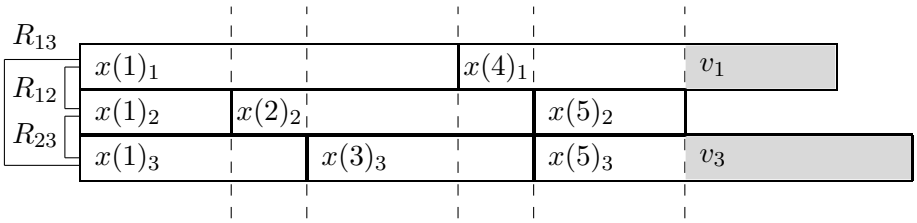


Figure 4.7: Example of a multi-tape machine, where R_{ij} are relations, $x(l)_i$ are words in X_i and v_i are the current overflows. Dashed lines indicate the positions which correspond to vertices of the Spehner graph.

In a Spehner graph, a *walk* of length m is a sequence $e_1 \cdots e_m$, where $e_l = u(l)u(l+1)$ is an edge from a vertex $u(l) = (u(l)_1, \ldots, u(l)_n)$ to a vertex $u(l+1) = (u(l+1)_1, \ldots, u(l+1)_n)$ labelled by $x(l) = (x(l)_1, \ldots, x(l)_n)$. It is *closed* if $u(1) = u(m+1)$. Let $w = e_1 \cdots e_m$ be a walk from $(\varepsilon, \ldots, \varepsilon)$ to an arbitrary state $(v_1, \ldots, v_n)$. By the construction of $S_{\mathcal{R}}(X_1, \ldots, X_n)$, the labels $(x(1)_1, \ldots, x(1)_n), \ldots, (x(m)_1, \ldots, x(m)_n)$ of w give us words $w_i = x(1)_i \cdots x(m)_i \in X_i^*$ for $i = 1, 2, \ldots, n$. Note that since $x(l)_i$ can be the empty word for some indices $1 \leq l \leq m$, the sequence $x(1)_i, \ldots, x(m)_i$ is not necessarily a real X_i-factorization of w_i. By (4.13) and (4.14) it is straightforward to prove by induction on the length of the walk m that

$$w_i (v_i)^{-1} \, R_{ij} \, w_j (v_j)^{-1} \tag{4.15}$$

for $1 \leq i < j \leq n$. In the sequel we are especially interested in closed walks from the initial state $(\varepsilon, \ldots, \varepsilon)$ back to itself. In that case there are no

overflows in the end of the walk and the labels of w indicate words $w_i \in X_i^*$ such that $w_i \, R_{ij} \, w_j$. Moreover, removing empty words $x(l)_i \notin X_i$ from the word $x(1)_i \cdots x(m)_i$ we get an X_i-factorization of w_i. These observations are fundamental for the algorithms of this section.

First we show how Spehner graphs give an efficient way of implementing the relational Sardinas–Patterson test. In practice, constructing sets U_i in Procedure 4.1 one by one may not be the fastest way of testing relational codes. Here we use generalized Spehner graphs in order to find out the elements in all of the sets U_i simultaneously. Actually, we need only a simplified version of the graph $S_{\mathcal{R}}(X, X)$. For the sake of clarity, we delete all labels of the edges and some unnecessary empty words and denote $R_{12} = R$. More precisely, the vertices of this simplified graph $S_R(X)$ are prefixes of words in X including the empty word ε, which acts as the initial vertex. The set of vertices consists of those edges accessible from ε using edges of the following type. There is an edge from a vertex $u \neq \varepsilon$ to a vertex v if, for some word $x \in X$, we have $v = R(x)^{-1}u$ or $v = R(u)^{-1}x$. In addition, from ε there is an edge to a vertex v if there exist words x and y in X such that $v = R(x)^{-1}y$ and $(x, y) \notin R$. Now the vertices at a distance one from ε form the set $U_1 = R(X)^{-1}X \setminus \{\varepsilon\}$. By the distance from u to v we mean the minimal length of a walk from u to v. Considering the construction of the simplified graph $S_R(X)$ we conclude by induction that a vertex v in $S_R(X)$ at a distance of exactly i from the state ε belongs to U_i. Hence, by Theorem 4.4, the set X is an (R, R)-code if and only if in $S_R(X)$ there is no closed walk from ε back to itself. Together with Theorem 4.3 this gives us the following algorithm for deciding whether a finite set is an (R, S)-code.

Algorithm 4.3. RelationalCodeTest(X, R, S)

`INPUT`: a finite set of words X, an alteration relation R, a fidelity relation S.

1 Construct the graph $S_R(X)$.
2 `IF` there is no walk from ε to ε in $S_R(X)$
3 `THEN IF` $R_X \subseteq S_X$ `return` X is an (R, S)-code
4 `ELSE return` X is not an (R, S)-code.

`OUTPUT`: the algorithm tells whether X is an (R, S)-code or not.

We assume that in the algorithm the (infinite) similarity relations are given using a finite representation $\langle r_1, \ldots, r_l \rangle$ defined in Section 3.1. Suppose that both of the input relations R and S have representations consisting of at most m pairs of letters. Then the operation of deciding whether two letters are R-similar or S-similar can be done in time $\mathcal{O}(m)$. Denote $n = \sum_{x \in X} |x|$. In the graph $S_R(X)$ there are at most $n + 1$ vertices. For each vertex, we construct the outgoing edges by comparing the vertex word to

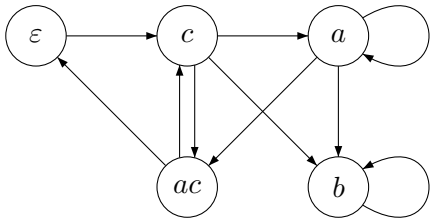
all words of X. This can be done using $\mathcal{O}(n)$ letter comparisons. Hence, the construction of $S_R(X)$ takes $\mathcal{O}(n^2m)$ time. Actually, the graph can be constructed even in time $\mathcal{O}(knm)$, where k is the number of words in X; see [30, p. 277] and [66]. Similarly, finding the paths in line 2 can be done in $\mathcal{O}(n^2)$ time using the breadth-first search [29, Section 22.2]. Furthermore, in line 3 we compare R-similarity and S-similarity of all pairs in $X \times X$ again in $\mathcal{O}(n^2m)$ time. Thus, the complexity of RelationalCodeTest(X, R, S) is $\mathcal{O}(n^2m)$. Furthermore, if we assume that the alphabet is fixed, then the input relations R and S can be considered to have constant size ($m = \mathcal{O}(1)$). This gives us quadratic time complexity. Hence, we have showed:

Theorem 4.23. *The algorithm* RelationalCodeTest(X, R, S) *tests whether X is an (R, S)-code or not. If the finite representations of the input relations consist of at most m pairs and $n = \sum_{x \in X} |x|$, then the time complexity of the algorithm is $\mathcal{O}(n^2m)$.*

Note also that, for regular languages, instead of the above combinatorial iterative algorithm, there is an automata theoretic (inefficient but effective) way to determine whether a regular set is an (R, S)-code or not. The question is reduced to the emptiness problem of regular languages; see [67] for the problem and for regular languages in general. Indeed, a (regular) set L is an (R, S)-code if and only if $xL^* \cap R(yL^*) = \emptyset$ for all $x, y \in L$ satisfying $(x, y) \notin R$ and $R_L \setminus S_L = \emptyset$.

Let us now demonstrate Algorithm 4.3 by considering the following example.

Example 4.8. Let $X = \{aac, ab, bb, ca\}$ and $R = \langle (a, c) \rangle$. The Spehner graph $S_R(X)$ and the sets U_i are given in Figure 4.8. Since there is a walk from ε to ε in $S_R(X)$, RelationalCodeTest(X, R, S) returns "X is not an (R, S)-code" for any input relation S.

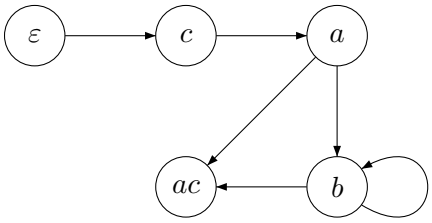


$$\begin{aligned} U_1 &= \{c\} \\ U_2 &= \{a, ac, b\} \\ U_i &= \{a, ac, b, c, \varepsilon\}, \ i \geq 3 \end{aligned}$$

Figure 4.8: The Spehner graph $S_R(X)$ and the sets U_i

On the other hand, setting $R' = \langle (a, b) \rangle$ we get Figure 4.9. Since there is no walk from ε to itself and trivially $R'_X \subseteq R'_X$, the algorithm RelationalCodeTest(X, R', R') returns "X is an (R', R')-code." How-

ever, RelationalCodeTest(X, R', ι) reveals that X is not an (R', ι)-code since $ab\,R'\,bb$, but $ab \neq bb$.



$$\begin{aligned} U_1 &= \{c\} \\ U_2 &= \{a\} \\ U_i &= \{ac, b\},\ i \geq 3 \end{aligned}$$

Figure 4.9: The Spehner graph $S_{R'}(X)$ and the sets U_i

Procedures for finding hulls for $X \subseteq \mathcal{A}^+$ have been considered, for example, in [48]. There, the idea is to find the words in X^+ not having unique X-factorizations and to make the factorizations equal by splitting the factors into smaller pieces. This method is based on stability. For relational hulls we can use a similar approach. Using generalized Spehner graphs we find all R-similar word pairs and refine their Base(X^*)-factorizations so that they satisfy the relational decoding condition (4.1). We present an algorithm that constructs the minimal generating set of the inner (R, R)-hull of X. The output is also the base of the inner (R, S)-hull for $S \neq R$ if the hull exists. Since X and $X \setminus \{\varepsilon\}$ generate the same monoid we may without loss of generality restrict to the case where X does not contain the empty word.

Consider a Spehner graph $S_R(I) = S_{\mathcal{R}}(X_1, X_2, X_3, X_4)$, where $X_1 = X_2 = X$, $X_3 = X_4 = Y$ and in the set $\mathcal{R}$ we have

$$\begin{aligned} &R_{12} = R, \quad R_{13} = \iota, \quad R_{14} = R, \\ &R_{23} = R, \quad R_{24} = \iota, \quad R_{34} = R. \end{aligned} \tag{4.16}$$

Let us denote the set of closed walks from the state $(\varepsilon, \varepsilon, \varepsilon, \varepsilon)$ back to itself by W_ε. A walk $w = e_1 \cdots e_m \in W_\varepsilon$ corresponds to words $w_1, w_2 \in X^*$ and $w_3, w_4 \in Y^*$ using the notation in page 61. By the relations (4.16) and (4.15), these words satisfy

$$w_1 = w_3, \quad w_2 = w_4, \quad w_1\,R\,w_2 \quad \text{and} \quad w_3\,R\,w_4. \tag{4.17}$$

Let $e_l = u(l)u(l+1)$ be an edge in the walk w such that $u(l)_3 = u(l)_4 = \varepsilon$. For the words in the third and fourth component of the labels, i.e., in the monoid Y^*, we denote

$$\begin{aligned} x' &= x(1)_3 \cdots x(l-1)_3, \\ u &= x(l)_3, \\ x'' &= x(l+1)_3 \cdots x(m)_3, \end{aligned}$$

$$\begin{aligned} y' &= x(1)_4 \cdots x(l-1)_4, \\ v &= x(l)_4, \\ y'' &= x(l+1)_4 \cdots x(m)_4. \end{aligned}$$

By (4.17), we have $x'ux'' \in X^*$ and $y'vy'' \in X^*$ satisfying $x'ux'' \, R \, y'vy''$. Since $u(l)_3 = u(l)_4 = \varepsilon$, we also have $x' \, R \, y'$ by (4.15). Hence, $|x'| = |y'|$ and (u, v) is an R-match for Y over X by (4.9). If $|u| = |v|$, then $(u, v) \in R$ and the match is trivial. Otherwise, (u, v) is a nontrivial R-match.

On the other hand, if (u, v) is an R-match for Y over X it is clear that there is a path $w \in W_\varepsilon$ as above. Thus, in principal we can construct the sets $C^i_{R,X}(Y)$ and $D^i_{R,X}(u, Y)$ by finding all edges such that they belong to some walk $w \in W_\varepsilon(S_R(I))$ and they are of the form

$$(u_1, u_2, \varepsilon, \varepsilon) \xrightarrow{(x_1, x_2, x_3, x_4)} (v_1, v_2, v_3, v_4).$$

In the following algorithms we denote the set of such edges in a walk w by $E(w)$. Actually, there may be an infinite number of walks in W_ε, since a walk may contain closed subwalks $e_k \cdots e_{k'}$ $(u(k) = u(k'+1))$ which can be repeated an arbitrary number of times. Fortunately, we may restrict ourselves to walks $\widetilde{w}$ from $(\varepsilon, \varepsilon, \varepsilon, \varepsilon)$ to $(\varepsilon, \varepsilon, \varepsilon, \varepsilon)$ in $S_R(I)$ which do not contain any closed subwalk twice. Let us denote this finite set by $\widetilde{W}_\varepsilon(S_R(I))$. Repeating closed subwalks is unnecessary since the walks obtained this way will not contain any new vertices or edges and, therefore, no new R-matches for Y over X are found. Hence, we have shown that the following algorithm finds a nontrivial inner R-match for Y over X if $C^i_{R,X}(Y)$ is not empty.

Algorithm 4.4. NONTRIVIALINNERMATCH($S_R(I)$)

`INPUT`: a Spehner graph $S_R(I) = S_R(X, X, Y, Y)$.

1 Find an edge in $E(w)$ where $w \in \widetilde{W}_\varepsilon(S_R(I))$ such that $|x_3| \neq |x_4|$.
2 `IF` there is no such walk
3 `THEN RETURN NULL`
4 `ELSE RETURN` (x_3, x_4).

`OUTPUT`: a nontrivial inner R-match for Y over X, if it exists.

For an implementation of Procedure 4.2 we need also an algorithm which constructs the set $D^i_{R,X}(u, Y)$. In the first three lines Algorithm 4.5 finds all words $v \in Y$ such that (u, v) is a trivial inner R-match for Y over X for the value of the argument u in each recursion. This is clear by the above considerations on the sets $\widetilde{W}_\varepsilon(S_R(I))$ and $E(w)$. If such a word v is not in the set O, the algorithm adds it to O and calls itself recursively in order to

find all words in $D^i_{R,X}(v,Y)$. In each recursive call the set O either grows or the recursion stops. Hence, the algorithm clearly terminates since $O \subseteq Y$ is finite. In the end we obtain $O = D^i_{R,X}(u,Y)$, since by the definition $D^i_{R,X}(u,Y)$ is the union of $D^i_{R,X}(v,Y)$ for all v such that (u,v) is a trivial inner R-match for Y over X.

Algorithm 4.5. DMATCHES$(S_R(I), u, O)$

INPUT: a Spehner graph $S_R(I) = S_R(X,X,Y,Y)$, a word $u \in Y$, a subset $O \subseteq D^i_{R,X}(u,Y)$.

1 FOR EACH $w \in \widetilde{W}_\varepsilon(S_R(I))$
2 FOR EACH edge in $E(w)$ such that
3 $|x_3| = |x_4|$ AND (EITHER $x_3 = u$ OR $x_4 = u$)
4 IF $x_3 \neq u$ THEN set $v \leftarrow x_3$ ELSE set $v \leftarrow x_4$.
5 IF $v \notin O$ THEN
6 Set $O \leftarrow O \cup \{v\}$.
7 RUN DMATCHES$(S_R(I), v, O)$.

OUTPUT: The algorithm makes O equal to $D^i_{R,X}(u,Y)$.

Now we are ready to write an algorithm which finds the base of the inner hull of X using the outlines of Procedure 4.2 and the previous two algorithms.

Algorithm 4.6. INNERHULL(X,R)

INPUT: a finite set of nonempty words X, a similarity relation R.

1 Set $Y \leftarrow X$.
2 Set $G \leftarrow S_R(X,X,Y,Y)$.
3 WHILE NONTRIVIALINNERMATCH$(G) \neq$ NULL DO
4 Set $(x,y) \leftarrow$ NONTRIVIALINNERMATCH(G).
5 Set $k \leftarrow \min(|x|,|y|)$.
6 IF $|x| > |y|$ THEN set $u \leftarrow x$ ELSE set $u \leftarrow y$.
7 Set $O \leftarrow \{u\}$.
8 RUN DMATCHES(G,u,O).
9 FOR EACH $v \in O$
10 Set $Y \leftarrow (Y \setminus \{v\}) \cup \{\mathrm{pref}_k(v), (\mathrm{pref}_k(v))^{-1}v\}$.
11 Set $G \leftarrow S_R(X,X,Y,Y)$.
12 RETURN Y.

OUTPUT: the base of the inner (R,R)-hull of X.

Let us compare this algorithm to the steps of Procedure 4.2. In lines 3–6 the algorithm finds a nontrivial inner R-match $(u, v) \in C^i_{R,X}(Y)$ if it exists. Otherwise the algorithm terminates. This corresponds to step 1 in the procedure. The set $D^i_{R,X}(u, Y)$ is constructed in lines 7 and 8. In lines 9 and 10 we modify Y according to step 3 in Procedure 4.2. The graph $S_R(X, X, Y, Y)$ is updated in order to find $(u, v) \in C^i_{R,X}(Y)$ for the modified Y in the next run of the `WHILE`-loop. Since we have now shown that the algorithm follows the steps of the procedure, we know that it calculates correctly the base of $\widehat{I}_R(X)$ by Theorem 4.19. As in the analysis of Algorithm 4.3 we assume that the input relation R has a representation consisting of at most m pairs of letters and therefore the operation of deciding whether two letters are R-similar can be done in time $\mathcal{O}(m)$. Denote $n = \sum_{x \in X} |x|$. Let us now show that this algorithm works in polynomial time.

First of all, the construction of the graph $G = S_{\mathcal{R}}(X_1, X_2, X_3, X_4)$ takes polynomial time $\mathcal{O}(n^5 m)$. To see this, let us replace the edges where $l > 1$ components of the label vector are nonempty into l different edges where only one component of the label differs from the empty word. In other words, consider a vertex $u = (u_1, u_2, u_3, u_4)$ and let i be the smallest index such that $u_i = \varepsilon$. For any edge leaving from u, we require that $x_j = \varepsilon$ if $j \neq i$. This modification increases the number of vertices $|V|$ and edges $|E|$ but we know that in order to construct the outgoing edges of a vertex we only have to consider labels where exactly one vector component differs from ε. Let us first consider the number of vertices. For $u_i \in \mathrm{Pref}(X_i)$, $1 \leq i \leq 4$, there are at most $1 + \sum_{x \in X_i} |x|$ possibilities. Since $\sum_{y \in Y} |y| \leq n$ by line 10 of the algorithm and in each vertex there are four components, the number of vertices is $\mathcal{O}(n^4)$. In order to find the outgoing edges of u, we take a word $x_i \in X_i$ and compare it to the three components u_j, $j \neq i$ in order to satisfy (4.13). Finding all suitable words x_i needs at most $3n$ comparisons, each taking time $\mathcal{O}(m)$. Since $|V| = \mathcal{O}(n^4)$, the whole construction of the graph can be done in time $O(n^5 m)$.

The algorithm NontrivialInnerMatch(G) takes time $\mathcal{O}(n^5)$, since in order to find the edges in $E(w)$ we have to check the last two components of each vertex and label in G. Moreover, we may use the breadth-first search to confirm that an edge belongs to some walk $w \in \widetilde{W}_\varepsilon(S_R(I))$. Hence, the time needed for that is $|V| + |E| \in \mathcal{O}(n^5)$. In the two `FOR` loops of DMatches$(S_R(I), u, O)$ we similarly go through the graph $S_R(I)$. Hence, one call of Algorithm 4.5 takes time $\mathcal{O}(n^5)$ and there are at most $|Y| \leq n$ number of calls. This increases the total complexity of line 8 to $\mathcal{O}(n^6)$. The `FOR`-loop of InnerHull(X, R) in lines 9 and 10 needs only $\mathcal{O}(n)$ time. Since in the `WHILE`-loop of the hull algorithm at least one word of Y is split into two smaller words, the loop can be executed at most n times. Hence, in each run of the loop we run the algorithm NontrivialInnerMatch(G) in time $\mathcal{O}(n^5)$, DMatches$(S_R(I), u, O)$ in time $\mathcal{O}(n^6)$ and construct the graph

$S_R(X, X, Y, Y)$ in time $\mathcal{O}(n^5 m)$. Therefore, the total time complexity of the InnerHull(X, R) algorithm is $\mathcal{O}(n^6 m + n^7)$.

Theorem 4.24. *The algorithm* InnerHull(X, R) *returns the base of the* (R, R)*-free hull of* X*. If the finite representations of the input relations consist of at most* m *pairs and* $n = \sum_{x \in X} |x|$*, then the time complexity of the algorithm is* $\mathcal{O}(n^6 m + n^7)$*.*

Actually, we may modify the algorithm so that it runs in time $\mathcal{O}(n^6 m)$. Namely, if in the iteration of the `WHILE`-loop where u splits into words u' and u'' we do not split a word $v \in D^i_{R,X}(u, Y)$, then (u', v) is a new nontrivial inner R-match for the modified Y over X and it will be split in some later iteration. Hence, we may simplify the algorithm by deleting line 8.

The algorithms OuterHull(X, R) and FreeHull(X, R) for finding the base of the outer R-hull and the R-free hull of X, respectively, are easily obtained by modifying the Spehner graph $S_{\mathcal{R}}(X_1, X_2, X_3, X_4) = S_R(I)$ used in InnerHull(X, R). For the outer hull, we can use the same algorithm as for the inner hull by deleting the second component $X_2 = X$ and corresponding relations R_{12}, R_{23} and R_{24}. Namely, when searching for outer R-matches for Y over X we need to make sure that only the word w_1 is in X^*. In other words, the Spehner graph needed for OuterHull(X, R) is of the form $S_R(O) = S_{\mathcal{R}}(X, Y, Y)$, where $\mathcal{R}$ consists of relations $R_{12} = \iota$, $R_{13} = R$ and $R_{23} = R$. Thus, the time complexity of OuterHull(X, R) is $\mathcal{O}(n^5 m + n^6)$.

For the free hull, the case is even simpler. We may also delete the first component of $S_R(I)$ and consider only factorizations in Y^*. Hence, the Spehner graph used in this case is of the form $S_{\mathcal{R}}(X, X)$, where $\mathcal{R} = \{R_{12}\} = \{R\}$. This gives us time complexity $\mathcal{O}(n^4 m + n^5)$ for FreeHull(X, R). On the other hand, by Procedure 4.3 another way to construct Base$(\widehat{F}_R(X))$ is to iterate algorithm InnerHull(X, R).

Finally, we give an example of the use of generalized Spehner graphs in the inner hull algorithm.

Example 4.9. Recall Example 4.6, where $X = \{eee, fffi, ggi, hh, i\}$ and $R = \langle (e, f), (f, g), (g, h) \rangle$. Consider the situation in Table 4.2 where $X = \{eee, fff, gg, hh, i\}$ and $Z_1 = \{e, ee, f, ff, gg, hh, i\}$. Figure 4.10 illustrates a part of the Spehner graph $G = S_R(X, X, Z_1, Z_1)$. Four walks, which belong to $\widetilde{W}_\varepsilon(G)$, are represented.

Assume that the algorithm NontrivialInnerMatch(G) chooses the edge leaving from $(\varepsilon, \varepsilon, \varepsilon, \varepsilon)$ with label (eee, eee, ee, e). Hence, it returns the pair $(x_3, x_4) = (ee, e)$. In the algorithm InnerHull(X, R) we therefore run DMatches$(G, ee, \{ee\})$. From the graph we see that the algorithm adds the word ff to O, since an edge with label (eee, fff, ee, ff) belongs to $E(w)$ for a walk $w \in \widetilde{W}_\varepsilon(G)$. Then the algorithm calls recursively

DMATCHES(G, ff, O) which adds gg into O. The word $hh \in D^i_{R,X}(ee, Z_1)$ is found in DMATCHES(G, gg, O). Finally DMATCHES(G, hh, O) does not find any new words belonging to $D^i_{R,X}(ee, Z_1)$ and therefore we have $O = \{ee, ff, gg, hh\}$. The words in O are deleted from $Y = Z_1$ in the `FOR`-loop of INNERHULL(X, R) and replaced by the words e, f, g and h. Hence, INNERHULL(X, R) returns the set $\{e, f, g, h, i\}$ which is denoted by Z_2 in Table 4.2. It is the base of the inner R-hull and the R-free hull of X.

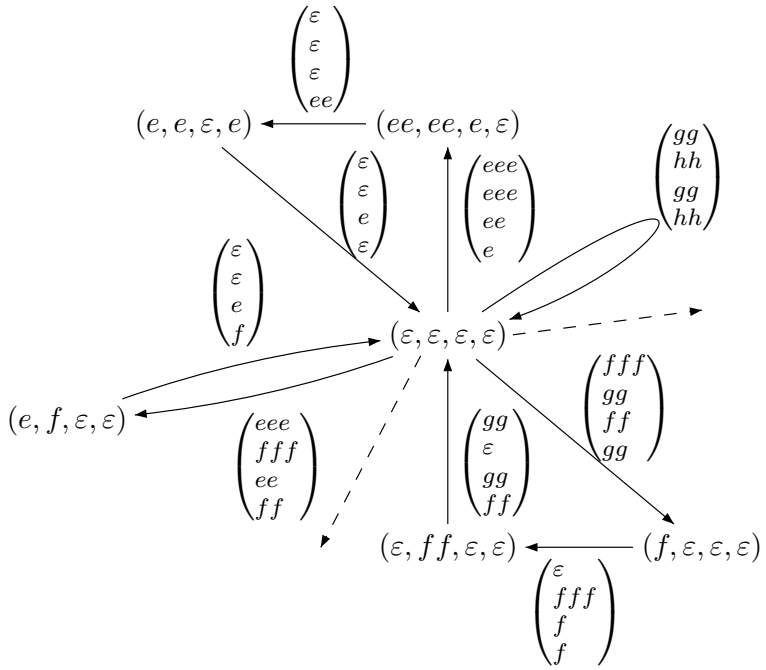


Figure 4.10: A part of the Spehner graph $S_R(X, X, Z_1, Z_1)$.

Chapter 5

Relational Periods

In this chapter we consider variations of the theorem of Fine and Wilf (Theorem 2.5). Section 5.1 begins with definitions of three types of relational periods: global, local and external. The first two are generalizations of the corresponding concepts of partial words. In Section 5.2 we prove various interaction theorems for relational periods. Different interaction types are treated in separate subsections. Typically, we consider words with one pure period and one relational period. The question is whether these two periods induce a third period for long enough words. A summary of the exact length bounds in different interaction cases is given at the end of the section. So called extremal relational Fine and Wilf words are discussed in Section 5.3. These are words of maximal length which do not express period interaction behavior.

5.1 Types of Relational Periods

Recall from Section 2.5 that an integer $p \geq 1$ is a period of a word $x = x_1 \cdots x_n$ if, for all positions i and j congruent modulo p, the letters x_i and x_j are equal. The minimal period of x is denoted by $\pi(x)$. In the sequel these periods are called *pure* as distinct from *relational periods* defined as follows.

Definition 5.1. Let R be a compatibility relation on an alphabet $\mathcal{A}$. For a word $x = x_1 \cdots x_n$, where $x_i \in \mathcal{A}$, an integer $p \geq 1$ is

(*i*) a *global R-period* of x if, for all $i, j \in \{1, 2, \ldots, n\}$, we have

$$i \equiv j \pmod{p} \implies x_i \, R \, x_j;$$

(*ii*) an *external R-period* of x if there exists a word $y = y_1 \cdots y_p$ such that, for all $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, p\}$, we have

$$i \equiv j \pmod{p} \implies x_i \, R \, y_j.$$

In this case, the word y is called an *external word* of x.

(*iii*) a *local R-period* of x if, for all $i \in \{1, 2, \ldots, n-p\}$, we have $x_i \, R \, x_{i+p}$.

These definitions generalize naturally to infinite words. For a word x, *the minimal global* (resp. *external, local*) *R-period* is denoted by $\pi_{R,g}(x)$ (resp. $\pi_{R,e}(x)$, $\pi_{R,l}(x)$). In the sequel, we may omit the subscript R or the argument x if the relation R or the word x is clear from the context. Similarly, if R is understood from the context, then we talk about global, local and external periods. Note that the above mentioned minimal periods may coincide. Next we give an example where all of them are different.

Example 5.1. Let $\mathcal{A} = \{a, b, c, d\}$ and denote $x = babbbcbd$. Let $R = \langle (a,b), (b,c), (c,d), (d,a) \rangle$ be a compatibility relation on the alphabet $\mathcal{A}$. Clearly, the minimal pure period is $\pi(x) = 8$. By the definition of R, we see that 2 is a local R-period of x. Since $(x_7, x_8) = (b,d) \notin R$, 1 is not a local period and therefore, we have $\pi_{R,l}(x) = 2$. Neither 1 nor 2 is an external R-period of x, since otherwise the letter y_1 or respectively y_2 in the external word y is related to every letter of the alphabet, which cannot be the case. Since $y = bab$ satisfies the conditions of an external word in Definition 5.1(ii), we have $\pi_{R,e}(x) = 3$. Furthermore, since $(b,d) \notin R$, we have $\pi_{R,g}(x) > 5$. Indeed, $\pi_{R,g}(x) = 6$, because of the relation $a \, R \, d$. Hence, for a word x, we have

$$\pi = 8 > \pi_g = 6 > \pi_e = 3 > \pi_l = 2.$$

As another example of relational periods we consider periods of partial words.

Example 5.2. Recall partial words from Section 3.2.1. In [6] two types of periods were defined: A partial word w has a *(partial) period p* if, for all $i, j \in D(w)$,

$$i \equiv j \pmod{p} \implies w(i) = w(j).$$

A partial word w has a *local (partial) period p* if

$$i, i+p \in D(w) \implies w(i) = w(i+p).$$

Using the similarity relation $R_\uparrow = \langle \{(\diamond, a) \mid a \in \mathcal{A}\} \rangle$ we see that a period of a partial word w corresponds to a global $R_\uparrow$-period of the companion $w_\diamond$. Similarly, a local partial period corresponds to a local $R_\uparrow$-period. Note that external periods are not very meaningful for partial words. Namely, any integer $p \geq 1$ is an external $R_\uparrow$-period of any word over the extended alphabet $\mathcal{A}_\diamond = \mathcal{A} \cup \{\diamond\}$. Indeed, we may choose $y = \diamond^p$ for an external word. Consequently, for partial words, we always have $\pi_e = 1$.

The next theorem shows how different types of periods are related to each other.

Theorem 5.1. *Every pure period of a word x is a relational (global, external and local) R-period for any compatibility relation R on $\mathcal{A}$. Every global R-period of x is an external R-period of x and a local R-period of x. Thus, for a word x, we always have $\pi(x) \geq \pi_g(x) \geq \max(\pi_e(x), \pi_l(x))$.*

Proof. Let R be a compatibility relation. By reflexivity, $\iota \subseteq R$ and therefore the first statement holds. Note that if $x = x_1 \cdots x_n$ has a period p, then $y = x_1 \cdots x_p$ is an external word of x. Similarly, this choice of y also shows that a global R-period is an external R-period. Clearly, a global period satisfies the definition of a local period. For the minimal periods, these considerations imply the inequalities of the statement. □

Note that every external period is not necessarily a local period and every local period need not be an external period. For instance, in Example 5.1 the minimal local R-period $\pi_l(x)$ is not an external R-period, and furthermore, $\pi_e(x)$ is not a local R-period. There we have $\pi_e(x) > \pi_l(x)$. Next we give an example where $\pi_l(x) > \pi_e(x)$.

Example 5.3. Let $R = \langle (a,b), (b,c), (c,d), (d,a) \rangle$ and let

$$x = adcbccccbd.$$

Consider first the minimal local R-period of x. Since $(x_9, x_{10}) = (x_4, x_2) = (b,d) \notin R$ and 3 is a local R-period, we have $\pi_l = 3$. Since $x_1 = a$, $x_4 = b$, $x_7 = c$ and $x_{10} = d$, there cannot exist any external word $y = y_1y_2y_3$ of length 3. Otherwise, y_1 would be compatible with all letters of the alphabet $\{a,b,c,d\}$. Hence, 3 is not an external R-period. For the same reason, 1 is not an external R-period, but by choosing $y = bc$, we see that $\pi_e = 2$. As noted above, 2 is not a local period. Since $(a,c) \notin R$, the minimal global R-period satisfies $\pi_g > 7$. Actually, $\pi_g = 8$ since $a\,R\,b$. Clearly, $\pi = 10$, and therefore

$$\pi = 10 > \pi_g = 8 > \pi_l = 3 > \pi_e = 2.$$

Next we show that if a similarity relation R is also transitive then all relational periods coincide.

Theorem 5.2. *If a similarity relation R is transitive, and thus an equivalence relation, then $P_g(x) = P_e(x) = P_l(x)$, where $P_g(x)$ (resp. $P_e(x)$, $P_l(x)$) is the set of all global (resp. external, local) R-periods of a word x. Moreover, in this case*

$$\pi_g(x) = \pi_e(x) = \pi_l(x).$$

Proof. Let $x = x_1 \cdots x_n$ be a word of length n and let R be an equivalence relation. By Theorem 5.1, we have $P_g(x) \subseteq P_e(x)$ and $P_g(x) \subseteq P_l(x)$. Consider an external R-period p with an external word $y = y_1 \cdots y_p$. Let

$i \equiv j \pmod{p}$, where $i, j \in \{1, 2, \ldots, n\}$. Then there exists $k \in \{1, 2, \ldots, p\}$ such that $i \equiv k \pmod{p}$ and $j \equiv k \pmod{p}$. Now $x_i \, R \, y_k$ and $x_j \, R \, y_k$ by the definition of an external word. Since R is transitive and symmetric, we have $x_i \, R \, x_j$. Hence, p is a global R-period, and we conclude that $P_e(x) \subseteq P_g(x)$.

Consider then a local R-period q of x. Let $i \equiv j \pmod{q}$, where $i, j \in \{1, 2, \ldots, n\}$. We may suppose that $j = i + kq$, where k is a nonnegative integer. We have

$$x_i \, R \, x_{i+q} \, R \, x_{i+2q} \, R \cdots R \, x_{i+(k-1)q} \, R \, x_{i+kq} = x_j.$$

Since R is transitive, we have $x_i \, R \, x_j$. Thus, q is a global period and we conclude that $P_l(x) \subseteq P_g(x)$. Hence, we have shown that $P_g(x) = P_e(x) = P_l(x)$. For the minimal periods, this clearly implies that $\pi_g(x) = \pi_e(x) = \pi_l(x)$. □

If R is not transitive, local R-periods differ from global and relational periods by the following property.

Lemma 5.1. *If p is a global R-period or an external R-period, then any multiple of p is a global R-period or an external R-period, respectively. This need not be the case for local R-periods.*

Proof. Suppose that p is a global R-period of x and let $i \equiv j \pmod{kp}$, where k is a nonnegative integer. Then clearly $i \equiv j \pmod{p}$ and, by the assumption, $x_i \, R \, x_j$. Hence kp is a global R-period. The proof is similar for external R-periods. Consider then a word $x = abc$ and a relation $R = \langle (a, b), (b, c) \rangle$. The word x has 1 as a local R-period, but 2 is not a local R-period. Thus multiples of local R-periods are not necessarily local R-periods. □

5.2 Variants of the Theorem of Fine and Wilf

In this section we study *interaction properties of periods* with respect to similarity relations. By the interaction property we mean that if a sufficiently long word has two periods then it also has another nontrivial period depending on the original periods. The theorem of Fine and Wilf (Theorem 2.5) is one of the cornerstones of combinatorics on words. In this theorem the derived period is the greatest common divisor of the original periods. Actually, this phenomenon was also the starting point of the study of partial words in the seminal paper of Berstel and Boasson in 1999 [6]. They proved the following variant of the theorem of Fine and Wilf for partial words with one hole.

Theorem 5.3. *Let w be a partial word of length n and suppose that it has local periods p and q. If the set of holes $H(w)$ is a singleton and if $n \geq p + q$, then w is purely* $\gcd(p, q)$*-periodic.*

Furthermore, Berstel and Boasson showed that the bound $p+q$ on the length of the word is sharp. Generalizations for several holes were considered, for example, by Blanchet-Sadri in [12] and Blanchet-Sadri and Hegstrom in [22], where it was shown that local partial periods p and q force a sufficiently long word to have a (global) partial period $\gcd(p,q)$ when certain unavoidable cases (*special words*) are excluded. The bound on the length depends on the number of holes in the word. On the other hand, Shur and Gamzova [72] found bounds for the length of a word with k holes such that (global) partial periods p and q imply a (global) partial period $\gcd(p,q)$. These results of partial words with several holes show that finding simple formulations for the interaction of periods in the case of arbitrary relational periods is not possible except for equivalence relations. Namely, any non-transitive compatibility relation R must have letter relations $(a_1,a_2),(a_2,a_3) \in R$, but $(a_1,a_3) \notin R$ for some letters a_1,a_2,a_3. Then the role of the letter a_2 in R for words over $\{a_1,a_2,a_3\}$ is exactly the same as the role of $\diamond$ for binary partial words and therefore formulations of Fine and Wilf's theorem depend considerably on the number of occurrences of the letter a_2. This indicates that the situation can be very complicated when we consider general non-transitive relations. The following example shows that without any additional assumption we cannot find a general bound for the interaction of relational periods.

Example 5.4. Let $R = \langle (a,b)(b,c) \rangle$ and consider the infinite (not necessarily ultimately periodic) word

$$w = w_1w_2w_3\cdots = acb^{6i_1-2}acb^{6i_2-2}\cdots,$$

where the numbers $i_j \geq 1$ are chosen freely. Now w has global R-periods 2 and 3. Namely,

$$w_1w_3w_5\cdots \in \{a,b\}^*, \quad w_2w_4w_6\cdots \in \{b,c\}^*,$$

and

$$w_1w_4w_7\cdots \in \{a,b\}^*, \quad w_2w_5w_8\cdots \in \{b,c\}^*, \quad w_3w_6w_9\cdots \in \{b\}^*.$$

However, $\gcd(2,3) = 1$ is not a global R-period of the word w. For example, $(w_1,w_2) = (a,c) \notin R$.

Moreover, for $R = \langle (a,b)(b,c) \rangle$, all numbers $2,3,4,\ldots$ are R-periods of the ultimately periodic word $w' = acbbb\cdots$, but 1 is not an R-period of w'. Nonetheless, some interaction results can be obtained. If the relation R is an equivalence relation, the situation reduces to Theorem 2.5.

Theorem 5.4. *Let R be an equivalence relation. If a word x has R-periods p and q and the length of the word is at least $p+q-\gcd(p,q)$, then $\gcd(p,q)$ is an R-period of x. The bound on the length is strict.*

Proof. Let R be an equivalence relation on the alphabet $\mathcal{A}$ and let x be a word over $\mathcal{A}$ with R-periods p and q and of length $n \geq p+q-\gcd(p,q)$. Suppose that $\mathcal{A}$ has m equivalence classes of R and let their set of representatives be $\{a_1,\ldots,a_m\}$. Let $\mathcal{B} = \{b_1,\ldots,b_m\}$ be another alphabet. Consider now a letter-to-letter morphism $\varphi\colon \mathcal{A}^* \to \mathcal{B}^*$, where for every $i \in \{1,2,\ldots,m\}$, each letter belonging to an equivalence class of a_i is mapped to b_i. This mapping is clearly well defined. Then $w = \varphi(x) = w_1 \cdots w_n$ is a word over $\mathcal{B}^*$.

First, let $i,j \in \{1,2,\ldots,n\}$ satisfy $i \equiv j \pmod{p}$. Since $x_i \, R \, x_j$ by the assumption, we have $w_i = \varphi(x_i) = \varphi(x_j) = w_j$ by the definition of the morphism φ. Thus, p is a period of w. Similarly, the word w is q-periodic. By the theorem of Fine and Wilf (Theorem 2.5), we therefore conclude that w is also $\gcd(p,q)$-periodic.

Next, let $i,j \in \{1,2,\ldots,n\}$ satisfy $i \equiv j \pmod{\gcd(p,q)}$. Then $w_i = w_j$ and, by the definition of φ, $x_i = \varphi^{-1}(w_i)$ and $x_j = \varphi^{-1}(w_j)$ belong to the same equivalence class, and hence, $x_i \, R \, x_j$. This means that $\gcd(p,q)$ is a relational R-period of the word x. Of course, the bound $p+q-\gcd(p,q)$ is the best possible, since there are counter examples to the original theorem of Fine and Wilf with length $p+q-\gcd(p,q)-1$ and our statement coincides with Theorem 2.5 by choosing $R = \iota$. □

As was mentioned above, the theorem of Fine and Wilf cannot be generalized to relational periods of a non-transitive compatibility relation unless some restrictions on the number of relations (holes) and exclusions of some special cases are given. Despite this fact, it might be possible to get some new interesting variations of the theorem, for example, by assuming that one of the periods is pure and only the other one is strictly relational. Unfortunately, this restriction seems to be insufficient in the extent that sometimes no finite bound on the length of the word can be obtained for the interaction of periods. For example, there exists an infinite word with a pure period q and a local R-period p such that it does not have a local R-period $\gcd(p,q)$.

Example 5.5. Let $R = \langle (a,b),(b,c) \rangle$. Note that every non-transitive compatibility relation must have a subrelation similar to this one such that a and c are not compatible. Consider an infinite word $x = (bbcab)^\omega$. Clearly, w has a pure period $q = 5$. It also has a local R-period $p = 3$, since the distance of the letters a and c in x cannot be 3. Since $(x_3,x_4) = (a,c) \notin R$, $\gcd(p,q) = 1$ is neither a local R-period nor a global R-period.

In the previous example the local relational period p is too weak to imply the desired interaction result. However, depending on the type of the relational period p, we get diverse results as will be shown in the sequel. For this purpose we define the bound of interaction.

Definition 5.2. Let $P \geq 2$ and $Q \geq 3$ be positive integers with $\gcd(P,Q) = d$ and let t_1 and t_2 be two types of relational R-periods; possibly $t_1 = t_2$. A positive integer $B = B(P,Q)$ is called the *bound of t_1-t_2 interaction for P and Q*, if it satisfies the following conditions:

(i) The bound B is *sufficient*, i.e., for any similarity relation R and for any word w with length $|w| \geq B$ having a pure period Q and a t_1-type R-period P, the number $\gcd(P,Q) = d$ is a t_2-type R-period of w.

(ii) The bound is *strict*, i.e., there exist a similarity relation R and a word w with length $|w| = B - 1$ having a pure period Q and a t_1-type R-period P such that $\gcd(P,Q) = d$ is not a t_2-type R-period of w.

Note that in the definition we exclude trivial cases by assuming that $P \geq 2$ and $Q \geq 3$. Namely, if $Q \leq 2$, then the word contains at most two letters. This is the case of Theorem 5.4, since there are no non-transitive compatibility relations on a binary alphabet. The next lemma shows that it is sufficient to consider cases where $\gcd(P,Q) = 1$.

Lemma 5.2. *Let $P = pd$ and $Q = qd$ be positive integers with $\gcd(P,Q) = d > 1$. If B is the bound of t_1-t_2-interaction for p and q, then Bd is the bound of t_1-t_2-interaction for P and Q.*

Proof. Suppose that a word w has a pure period $Q = qd$ and a relational t_1-type period $P = pd$, where $\gcd(P,Q) = d$. Denote the ith letter of w by w_i and assume that the length of w is at least dB, where $B = B(p,q)$. Let us consider the word

$$w^{(i)} = w_i w_{i+d} \cdots w_{i+k_i d},$$

where $1 \leq i \leq d$ and $k_i = \left\lfloor \frac{|w|-i}{d} \right\rfloor$. Note that the word $w^{(i)}$ has a pure period q and a t_1-type relational period p. Since $|w^{(i)}| \geq B$, $\gcd(p,q) = 1$ must be a t_2-type period for the word $w^{(i)}$ by the definition of $B = B(p,q)$. Since this is true for all $i = 1, 2, \ldots, d$, we conclude that d is a t_2-type relational period of w.

In order to prove that the bound Bd is strict, we give an example of a word u of length $Bd - 1$ such that it has a period Q and an R-period P but no R-period d. Suppose that $v = v_1 v_2 \cdots v_{B-1}$ is a word such that it has a pure period q and a t_1-type period p, but $\gcd(p,q) = 1$ is not a t_2-type relational period of v. By the definition of B, such a word exists. Let a be some letter in the alphabet $\mathcal{A}$ and define the word u by the following formula:

$$u = a^{d-1} v_1 a^{d-1} v_2 \cdots a^{d-1} v_{B-1} a^{d-1}.$$

Now u has a pure period $Q = qd$ and a t_1-type period $P = pd$, but by the properties of v, $\gcd(P,Q) = d$ cannot be a t_2-type R-period of u. □

In the following subsections we consider interaction bounds for different interaction types in more detail.

5.2.1 Global-Global Interaction

Let R be a similarity relation. Here we consider the case where one pure period and one global R-period imply a derived global R-period. The bounds of global-global interaction $B_g(p,q)$ for coprime integers p and q are given in Table 5.1, and we state our first interaction result in the following theorem.

$B_g(p,q)$	$p < q$	$p > q$
p, q odd	$\frac{p+1}{2}q$	$q + \frac{q-1}{2}p$
p odd, q even	$\frac{p+1}{2}q$	$\frac{p+1}{2}q$
p even, q odd	$q + \frac{q-1}{2}p$	$q + \frac{q-1}{2}p$

Table 5.1: Table of bounds $B_g(p,q)$, where $\gcd(p,q) = 1$

Theorem 5.5. *Let p and q be positive integers with $\gcd(p,q) = 1$. The bound of global-global interaction for p and q is $B_g(p,q)$ given in Table 5.1.*

In order to make the proof of this theorem more readable, we divide it into two parts. Theorem 5.5 follows directly from the following two lemmata.

Lemma 5.3. *The bound $B_g(p,q)$ defined in Theorem 5.5 is sufficient.*

Proof. Suppose that a word w has a pure period q and a global R-period p. Assume further that $|w| \geq B_g = B_g(p,q)$. We show that 1 is a global R-period of w. Since the word w is a rational power of a word of length q, there are at most q different letters in w. Hence, it suffices to show that a letter in an arbitrary position $n \in \{1, 2, \ldots, q\}$ is R-compatible with all other letters of w.

We use the following notation. For an integer $n \in \{1, 2, \ldots, q\}$, we define $\tau(n) = \max\{m \mid 1 \leq m \leq |w|,\ m \equiv n \pmod q\}$. Note that if the word w has q different letters, then $\tau(n)$ is the last occurrence of the letter w_n in w. Since w has the global relational period p, it follows that w_n must be related to all letters in the positions

$$S(n) = \left\{ n + ip \;\middle|\; i = 0, 1, \ldots, \left\lfloor \frac{|w| - n}{p} \right\rfloor \right\}$$

and

$$T(n) = \left\{ \tau(n) - ip \;\middle|\; i = 1, 2, \ldots, \left\lfloor \frac{\tau(n)-1}{p} \right\rfloor \right\}.$$

Our aim is to prove that, for all $n \in \{1, 2, \ldots, q\}$, the union $S(n) \cup T(n)$ contains a complete residue system modulo q. This implies that every w_n is R-compatible with all letters w_i for $i = 1, 2, \ldots, q$ and, by the q-periodicity, 1 is an external R-period of w.

Firstly, assume that $\tau(n) \equiv n \pmod{p}$. Then $\tau(n) \equiv n \pmod{pq}$, since $\tau(n) \equiv n \pmod{q}$ and $\gcd(p, q) = 1$. Hence, $|w| \geq pq$ and $S(n)$ contains the set $\{n + ip \mid i = 0, 1, \ldots, q-1\}$, which is a complete residue system modulo q.

Secondly, assume that $\tau(n) \not\equiv n \pmod{p}$ which ensures that the sets $S(n)$ and $T(n)$ are disjoint. Now it suffices to prove that $|S(n) \cup T(n)| \geq q$. Namely, in that case there exists an integer $k \in \{0, q-2\}$ and two sets

$$S'(n) = \{n + ip \mid i = 0, 1, \ldots, k\}$$

and

$$T'(n) = \{\tau(n) - jp \mid j = 1, 2, \ldots, q-k-1\},$$

such that $S'(n) \cup T'(n)$ is a complete residue system modulo q. To justify this, we note that the elements of $S'(n)$ are pairwise incongruent modulo q, since $\gcd(p, q) = 1$ and $k < q$. The same holds for $T'(n)$. Assume next that for some $i \in \{0, 1, 2, \ldots, k\}$ and for some $j \in \{1, \ldots, q-k-1\}$ we have

$$n + ip \equiv \tau(n) - jp \pmod{q}. \tag{5.1}$$

This is true if and only if $(i + j)p \equiv \tau(n) - n \equiv 0 \pmod{q}$, where the second congruence follows from the definition of $\tau(n)$. Since $\gcd(p, q) = 1$, we conclude that (5.1) holds if and only if $i + j \equiv 0 \pmod{q}$. But this is not possible, since

$$0 < i + j \leq k + (q - k - 1) = q - 1 < q.$$

Therefore, the set $S'(n) \cup T'(n)$ contains exactly $(k + 1) + (q - k - 1) = q$ pairwise incongruent elements.

Thus, it remains to prove that if $|w| \geq B_g = B_g(p, q)$, then

$$|S(n) \cup T(n)| = 1 + \left\lfloor \frac{|w| - n}{p} \right\rfloor + \left\lfloor \frac{\tau(n)-1}{p} \right\rfloor \geq q. \tag{5.2}$$

Actually, we show that

$$M(n) := \tau(n) - \left(q - \left\lfloor \frac{B_g - n}{p} \right\rfloor - 1 \right) p > 0. \tag{5.3}$$

Since $M(n)$ is an integer, we have $\tau(n) - 1 \geq (q - \lfloor \frac{B_g - n}{p} \rfloor - 1)p$ and, consequently, $\lfloor \frac{\tau(n)-1}{p} \rfloor \geq q - \lfloor \frac{B_g - n}{p} \rfloor - 1$. Since $|w| \geq B_g$, this implies (5.2). Note that $B_g - n < pq$ and therefore

$$q - \left\lfloor \frac{B_g - n}{p} \right\rfloor - 1 \geq q - (q-1) - 1 = 0.$$

We use the notation $[n]_q$ for the least positive residue of an integer n (mod q), i.e., $[n]_q$ is the positive integer m satisfying $1 \leq m \leq q$ and $m \equiv n$ (mod q). Since $B_g \geq q$, we have

$$\tau(n) \geq \begin{cases} B_g - [B_g]_q + n & \text{if } n \in \{1, 2, \ldots, [B_g]_q\}, \\ B_g - [B_g]_q - q + n & \text{if } n \in \{[B_g]_q + 1, [B_g]_q + 2, \ldots, q\}. \end{cases}$$

Let us consider two cases.

Case 1. Assume first that $B_g = \frac{p+1}{2}q$. Then $[B_g]_q = q$ and we have

$$\begin{aligned} M(n) &> B_g + [B_g]_q + n - \left(q - 1 - \left(\frac{B_g - n}{p} - 1\right)\right)p \\ &= B_g - q + n - qp + B_g - n = 2B_g - (p+1)q \\ &= (p+1)q - (p+1)q = 0. \end{aligned}$$

Case 2. Assume next that $B_g = q + \frac{q-1}{2}p$. Note that

$$\left\lfloor \frac{B_g - n}{p} \right\rfloor = \frac{q-1}{2} + \left\lfloor \frac{q-n}{p} \right\rfloor \geq \frac{q-1}{2},$$

since q is odd and $q \geq n$. If $n \in \{1, 2, \ldots, [B_g]_q\}$, then

$$\begin{aligned} M(n) &\geq B_g - [B_g]_q + n - \left(q - \frac{q-1}{2} - 1\right)p \\ &= q + \frac{q-1}{2}p - [B_g]_q + n - qp + \frac{q-1}{2}p + p \\ &= q - [B_g]_q + n \geq n > 0. \end{aligned}$$

On the other hand, if $n \in \{[B_g]_q + 1, [B_g]_q + 2, \ldots, q\}$, then

$$\begin{aligned} M(n) &\geq B_g - [B_g]_q - q + n - \left(q - \frac{q-1}{2} - 1\right)p \\ &= q + \frac{q-1}{2}p - [B_g]_q - q + n - qp + \frac{q-1}{2}p + p \\ &= n - [B_g]_q > 0. \end{aligned}$$

□

Next we prove that our bound is strict.

Lemma 5.4. *The bound $B_g(p,q)$ defined in Theorem 5.5 is strict.*

Proof. Using the notation of the previous proof, we fix $n = [B_g]_q$. In addition, we define so called *critical positions* $m(p,q) \in \{1,2,\ldots,q\}$ according to Table 5.2. We show that it is possible to construct a word v of length $|v| = B_g - 1$ with a pure period q and a global R-period p such that the letter in the critical position is not related to the letter in the position n. This implies that 1 is not a global R-period of v. Note that all these critical positions are positive integers less than or equal to q. In the sequel we denote critical positions succinctly by m.

$m(p,q)$	$p < q$	$p > q$
p, q odd	$\frac{q-p}{2}$	q
p odd, q even	$\frac{q}{2}$	$\frac{q}{2}$
p even, q odd	q	q

Table 5.2: Table of critical positions $m(p,q)$

Consider now solutions (i,j) for the equation

$$m + iq \equiv n + jq \pmod{p}, \tag{5.4}$$

such that i and j are nonnegative integers. By a *minimal solution* we mean a solution where $\max(n+iq, m+jq)$ is as small as possible. Note that if $i > j$ for some solution, then $m+(i-j)q \equiv n \pmod{p}$ is a smaller solution. Similarly, if $j > i$, then $m \equiv n+(j-i)q \pmod{p}$ is a smaller solution. Thus, a minimal solution is of the form where either $i = 0$ or $j = 0$. Moreover, such a solution is unique. Namely, if (i,j) and (i',j') are distinct minimal solutions, then without loss of generality we may assume that $j = 0$, $i' = 0$, and $m + iq = n + j'q$. In other words, m and n are congruent modulo q, which is a contradiction, since $m \in \{1,2,\ldots,q\} \setminus \{n\}$.

Since $\gcd(p,q) = 1$, we know that $\{m+iq \mid i = 0,1,\ldots,p-1\}$ and $\{n+jq \mid j = 0,1,\ldots,p-1\}$ are complete residue systems modulo p. Hence there exists exactly one $j \in \{0,1,\ldots,p-1\}$ satisfying $m \equiv n + jq \pmod{p}$ and exactly one $i \in \{0,1,\ldots,p-1\}$ satisfying $m + iq \equiv n \pmod{p}$. Furthermore, for $j \in \{1,2,\ldots,p-1\}$, we have

$$m \equiv n + jq \pmod{p} \implies m + (p-j)q = m + pq - jq \equiv n \pmod{p},$$

and $p - j \in \{1,2,\ldots,p-1\}$. Hence, the minimal solution of (5.4) is either of the form $(0,j)$ or $(p-j,0)$.

Now we prove that regardless of the parity of p and q and which of them is greater, the minimal solution is

$$i = 0 \quad \text{and} \quad j = \frac{B_g - n}{q}.$$

Note that, since $B_g < pq$, we have $\frac{B_g - n}{q} \in \{1, 2, \ldots, p-1\}$ in all cases. Consider first those cases of Table 5.1 where $B_g = \frac{p+1}{2}q$ and, consequently, $n = [B_g]_q = q$. Let $j = \frac{B_g - n}{q}$.

Case 1. Let p and q be both odd and $p < q$. In Table 5.2, we have $m = \frac{q-p}{2}$. Now $n + jq = B_g$ and, since q is odd, it follows that

$$(n + jq) - m = \frac{p+1}{2}q - \frac{q-p}{2} = \frac{q+1}{2}p \equiv 0 \pmod{p}.$$

Hence, $(0, \frac{B_g - n}{q})$ is a solution. Furthermore,

$$jq = B_g - n = \frac{p+1}{2}q - q = \frac{p-1}{2}q$$

and

$$m + (p-j)q = m + pq - \frac{p-1}{2}q = m + \frac{p+1}{2}q = m + B_g > B_g.$$

Hence, in the solution $(p - \frac{B_g - n}{q}, 0)$, we have $\max(m + iq, n + jq) > B_g$ whereas in the solution $(0, \frac{B_g - n}{q})$, we have $\max(m + iq, n + jq) = B_g$. Thus, $(0, \frac{B_g - n}{q})$ is the minimal solution.

Case 2. Suppose that p is odd and q is even. By the parity of q, $m = \frac{q}{2}$ is an integer and

$$(n + jq) - m = \frac{p+1}{2}q - \frac{q}{2} = \frac{q}{2}p \equiv 0 \pmod{p}.$$

Hence, $(0, \frac{B_g - n}{q})$ is a solution. As in Case 1, we have

$$m + (p-j)q = m + B_g > B_g,$$

and therefore $(0, \frac{B_g - n}{q})$ is the minimal solution in this case also.

Consider next those cases where $B_g = q + \frac{q-1}{2}p$. According to Table 5.1 and Table 5.2 we have $m = q$ and q is odd. Clearly, $(i, j) = (0, \frac{B_g - n}{q})$ is a solution, since

$$(n + jq) - m = q + \frac{q-1}{2}p - q = \frac{q-1}{2}p \equiv 0 \pmod{p}.$$

As above, we have $m+(p-j)q = m+pq-B_g+n$. By substituting m and B_g we get

$$m+(p-j)q = q+\left(2\cdot\frac{q-1}{2}p+p\right)-\left(q+\frac{q-1}{2}p\right)+n = B_g+(p-q)+n.$$

Case 3. Assume that $p>q$. Then $p-q$ is positive and $m+(p-j)q > B_g$. Thus, $(0,\frac{B_g-n}{q})$ is the smallest solution.

Case 4. Assume that p is even, q is odd and $p<q$. Then $n = q-\frac{p}{2}$. Moreover,

$$m+(p-j)q = B_g+(p-q)+q-\frac{p}{2} = B_g+\frac{p}{2} > B_g,$$

and we conclude that $(0,\frac{B_g-n}{q})$ is the smallest solution also in this final case.

Define now a word w in the three letter alphabet $\{a,b,c\}$ by the rule

$$w = \begin{cases} (b^{m-1}ab^{q-m-1}c)^{\frac{p+1}{2}} & \text{if } B_g = \frac{p+1}{2}q, \\ (b^{n-1}cb^{q-n-1}a)^{\frac{B_g-n}{q}}b^{n-1}c & \text{if } B_g = q+\frac{q-1}{2}p, \end{cases} \tag{5.5}$$

where $m = m(p,q)$ is given by Table 5.2 and $n = [B_g]_q$. Furthermore, let $v = wc^{-1}$ and assume that $R = \langle (a,b),(b,c)\rangle$. Now q is clearly a pure period of v and p is a global R-period. Namely, b is related to each letter in v, and the first occasion where the distance between the letter a and the letter c in w is a multiple of p is the case where a is in the position m and c is in the position B_g. This does not happen in v, since v is one letter shorter. Since $(a,c)\notin R$, 1 is not a global R-period of v. □

Let us consider an example.

Example 5.6. For $p=5$ and $q=7$, the bound for global-global interaction is $B_g(p,q) = \frac{p+1}{2}q = 21$. Hence, any word w such that it has a global R-period 5, a pure period 7 and no relational period $\gcd(p,q)$ is at most of the length $B_g(5,7)-1 = 20$. The situation is illustrated in Figure 5.1. The table with 20 entries represents the word $w = (w_1w_2\cdots w_7)^2w_1w_2\cdots w_6$, which is a rational power of the word $w_1\cdots w_q$, written into p columns. All the letters in the same column are R-related, since p is a global period. The graph with vertices $w_1,\ldots,w_q$ represents all necessary relations in the word. If two vertices occur in the same column of the table, then there is an edge between those vertices. We notice that the graph is almost complete, only the edges (relations) (w_1,w_7) and (w_6,w_7) are missing. Hence, we conclude that $w = (abbbbbc)^2abbbbb$ and $w' = (abbbbac)^2abbbba$ with relation $R = \langle (a,b),(b,c)\rangle$ are words such that they have global R-period 5 and pure period 7, but 1 is not a global R-period. Note that w satisfies the formula given by (5.5). Namely, $m(5,7) = 1$ by Table 5.2. Moreover, from the figure

we see that increasing the length of the word w by 1 is not possible. The next letter is indicated in the table by $\not w_7$. We notice that increasing the length causes w_1, w_6 and w_7 to occur in the same column. Hence, the graph would become perfect (dashed lines) implying that all the letters would be related to each other and, therefore, 1 would be a global R-period.

w_1	w_2	w_3	w_4	w_5
w_6	w_7	w_1	w_2	w_3
w_4	w_5	w_6	w_7	w_1
w_2	w_3	w_4	w_5	w_6
$\not w_7$				

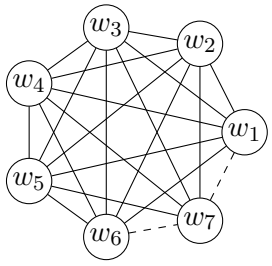


Figure 5.1: Global-global interaction for $p = 5$ and $q = 7$.

5.2.2 Global-Local Interaction

Instead of attaining a global period $\gcd(p, q)$ we loosen our requirements and consider the case where the greatest common divisor becomes a local relational period.

Theorem 5.6. *Let p and q be positive integers with $\gcd(p, q) = 1$. Let k be the smallest integer satisfying $kp \equiv \pm 1 \pmod{q}$. The bound of global-local interaction for p and q is*

$$B_l(p, q) = \begin{cases} q + kp - 1 & \text{if } q \equiv 2 \pmod{p} \text{ and } kp \equiv +1 \pmod{q}, \\ q + kp & \text{otherwise.} \end{cases}$$

As in the previous subsection, we divide the proof into two parts.

Lemma 5.5. *The bound $B_l(p, q)$ defined in Theorem 5.6 is sufficient.*

Proof. Denote $B_l = B_l(p, q)$. Assume that a word w has a pure period q and a global R-period p. We show that 1 is a local R-period of w if $|w| \geq B_l$. Like in the proof of Lemma 5.3 we conclude that there are at most q different letters in w. Hence, the word w has a local R-period 1 if and only if, for all $n = 1, 2, \ldots, q$, we have

$$w_{[n]_q} \, R \, w_{[n+1]_q}, \tag{5.6}$$

where $[n]_q$ denotes the least positive residue of n modulo q. We show that, for each $n \in \{1, 2, \ldots, q\}$, there exist integers $i_n, j_n \in \mathbb{N}$ such that

$$[n]_q + i_n q \equiv [n + 1]_q + j_n q \pmod{p} \tag{5.7}$$

and both sides of the congruence belong to the set $\{1, 2, \ldots, B_l\}$. This implies together with the global period p of w that (5.6) must be satisfied if $|w| \geq B_l$.

Case 1. Assume first that $kp \equiv 1 \pmod{q}$. For each $n \in \{1, 2, \ldots, q-1\}$, choose $j_n = \frac{kp-1}{q}$ and $i_n = 0$. Note that j_n is an integer by the definition of k. Then

$$(n+1) + j_n q = n + 1 + kp - 1 = n + kp \equiv n \pmod{p}.$$

Clearly, both sides of the congruence belong to $\{1, 2, \ldots, B_l\}$. Furthermore, let $j_q = \frac{kp-1}{q} + 1$ and $i_q = 0$. Now

$$1 + j_q q = 1 + kp - 1 + q = q + kp \equiv q \pmod{p}.$$

The left hand side is less than or equal to B_l only if $q \not\equiv 2 \pmod{p}$. However, in this special case where $1 \equiv q - 1 \pmod{p}$, we can choose $i_q = \frac{kp-1}{q}$ and $j_q = 0$ so that

$$q + i_q q = q + kp - 1 \equiv q - 1 \equiv 1 \pmod{p}.$$

Now the left hand side is exactly B_l.

Case 2. Assume that $kp \equiv -1 \pmod{q}$ and, for $n \in \{1, 2, \ldots, q-1\}$, let $i_n = \frac{kp+1}{q}$ and $j_n = 0$. Note that i_n is an integer by the definition of k. Hence,

$$n + i_n q = n + kp + 1 \equiv n + 1 \pmod{p}.$$

Choose furthermore $i_q = \frac{kp+1}{q} - 1$ and $j_q = 0$. Then

$$q + i_q q = q + kp + 1 - q \equiv 1 \pmod{p}.$$

Note that both sides of both congruences belong to the set $\{1, 2, \ldots, B_l\}$.

Hence, we have shown that (5.6) is satisfied for all $n = 1, 2, \ldots, q$, if $|w| \geq B_l$. Therefore, w must have $\gcd(p, q) = 1$ as a local relational period. □

Lemma 5.6. *The bound $B_l(p, q)$ defined in Theorem 5.6 is strict.*

Proof. Denote $B_l(p, q) = B_l$. We prove that there exists a word w of length $B_l - 1$ such that it has a global period p and a pure period q but no local period $\gcd(p, q) = 1$. We show that, at least for one index $n \in \{1, 2, \ldots, q\}$, there is no solution i_n, j_n of (5.7) such that both sides of the equation belong to the set $\{1, 2, \ldots, B_l - 1\}$. Without contradicting the assumption that p is a global period of w we may then assume that $(w_{[n]_q}, w_{[n+1]_q}) \notin R$ and therefore $\gcd(p, q) = 1$ is not a local R-period of w.

Recall that k is the smallest integer satisfying $kp \equiv \pm 1 \pmod{q}$. We begin the proof by showing that

$$k < \frac{q}{2}. \tag{5.8}$$

If $k > \frac{q}{2}$, then also $(q-k)p \equiv -kp \equiv \mp 1 \pmod{q}$. Since $q-k$ is now smaller than k, this is a contradiction. Assume next that $k = \frac{q}{2}$. Then, by the definition of k, $0 \equiv qp = 2kp \equiv \pm 2 \pmod{q}$. We get a contradiction, since $q \geq 3$ by the definition of the interaction bound.

Next we will consider minimal solutions of (5.7). As in the proof of Lemma 5.4, by a minimal solution we mean a solution (i_n, j_n) where $\max([n]_q + i_n q, [n+1]_q + j_n q)$ is as small as possible. Recall that the minimal solution is unique and if $j \in \{1, 2, \ldots, p-1\}$, then the minimal solution is either $(0, j)$ or $(p-j, 0)$. We divide our considerations into three cases and show that in each case there exists some $n \in \{1, 2, \ldots, q\}$ such that the minimal solution (i_n, j_n) of (5.7) satisfies $\max([n]_q + i_n q, [n+1]_q + j_n q) \geq B_l$.

Case 1. Let us first assume that $kp \equiv 1 \pmod{q}$ and $q \not\equiv 2 \pmod{p}$. We consider (5.7) with $n = q$, i.e., the congruence $q + i_q q \equiv 1 + j_q q \pmod{p}$. Note that in the solution $(i_q, j_q) = (0, \frac{kp-1}{q}+1)$ we have $1 + j_q q = q + kp = B_l$. Now assume that the solution $(p - \frac{kp-1}{q} - 1, 0)$ is smaller, i.e., $q + (p - \frac{kp-1}{q} - 1)q < B_l$. Thus,

$$0 < q + kp - \left(q + \left(p - \frac{kp-1}{q} - 1\right) q\right) = 2kp - qp + q - 1 < q - 1, \quad (5.9)$$

where the last inequality follows from (5.8). Since $kp \equiv 1 \pmod{q}$, we have

$$2kp - qp + q - 1 \equiv 1 \pmod{q}. \quad (5.10)$$

Combining (5.9) and (5.10), we conclude that $2kp - qp + q - 1 = 1$. On the other hand,

$$1 = 2kp - qp + q - 1 \equiv q - 1 \pmod{p},$$

which contradicts our assumption. Therefore, the minimal solution is such that $\max(q + i_q q, 1 + j_q q) \geq B_l$.

Thus, let us define a rational power

$$w = (ab^{q-2}c)^{(B_l - 1)/q}$$

in the ternary alphabet $\{a, b, c\}$ with length $B_l - 1$. Let $R = \langle (a, b), (b, c) \rangle$. By the above considerations, the word w has a period q and a global R-period p. However, 1 is not a local R-period of w, since a and c are unrelated.

Case 2. Assume next that $kp \equiv 1 \pmod{q}$ and $q \equiv 2 \pmod{p}$. Consider the congruence

$$(q-1) + i_{q-1} q \equiv q + j_{q-1} q \pmod{p}.$$

Note that in the solution $(i_{q-1}, j_{q-1}) = (0, \frac{kp-1}{q})$ we have $q + j_{q-1} q = q + kp - 1 = B_l$. Moreover, in the solution $(p - \frac{kp-1}{q}, 0)$, we have

$$q - 1 + \left(p - \frac{kp-1}{q}\right) q = q - 1 + qp - kp + 1 \overset{(5.8)}{>} q + kp > B_l.$$

Hence, the minimal solution satisfies $\max(q-1+i_{q-1}q, q+j_{q-1}q) \geq B_l$. In this case, the rational power

$$w = (b^{q-2}ac)^{(B_l-1)/q}$$

and the relation $R = \langle (a,b),(b,c) \rangle$ together with the above calculations show that the bound B_l is strict.

Case 3. Finally assume that $kp \equiv -1 \pmod{q}$. Consider the same congruence as in Case 2. However, note that now $B_l = q + kp$. Now $(i_{q-1}, j_{q-1}) = (\frac{kp+1}{q}, 0)$ is one solution, where $q-1+(\frac{kp+1}{q})q = q+kp = B_l$. In the other solution $(0, p - \frac{kp+1}{q})$, we have

$$q + \left(p - \frac{kp+1}{q}\right) q = q + (q-k)p + 1 \overset{(5.8)}{>} q + kp + 1 > B_l.$$

Hence, the word

$$w = (b^{q-2}ac)^{(B_l-1)/q}$$

with the relation $R = \langle (a,b),(b,c) \rangle$ prove that the bound B_l is strict also in this case. □

Theorem 5.6 follows now directly from Lemma 5.5 and Lemma 5.6. Note that the value of k can be calculated easily using an elementary theorem by Fermat and Euler. Namely, the smallest solution k' of the equation $k'p \equiv 1 \pmod{q}$ is called the *reciprocal* of p modulo q and, by the theorem,

$$k' = [p^{\varphi(q)-1}]_q,$$

where φ is Euler's totient function. Thus, we have $k = \min(k', q-k')$, since $(q-k')p \equiv -1 \pmod{q}$.

Let us now continue to examine the case of Example 5.6 for global-local interactions.

Example 5.7. For $p = 5$ and $q = 7$, we have $k' = [5^{\varphi(7)-1}]_7 = 3 = k$ and $kp = 15 \equiv 1 \pmod{7}$. Since $q = 7 \equiv 2 \pmod{5}$, the bound of global-local interaction is $B_l(p,q) = q + kp - 1 = 21$. Since $B_l(5,7) = B_g(5,7)$, a word w of length $B_l(p,q) - 1 = 20$ with local R-period p and pure period q but not having 1 as a local R-period can be represented in a table form exactly as in Figure 5.1. Note that $B_g(p,q)$ is not equal to $B_l(p,q)$ in general. The word w has a local R-period 1 if and only if $w_{[n]_7} \, R \, w_{[n+1]_7}$ is satisfied for all $n = 1, 2, \ldots, 7$. From the graph of Figure 5.1 we see that the relations $w_6 \, R \, w_7$ and $w_7 \, R \, w_1$ are missing. Hence, the word w could be, for example, $(abbbbac)^2 abbbba$ or $(bbbbbac)^2 bbbbba$ with the relation $\langle (a,b),(b,c) \rangle$. Note that the latter one corresponds to the formula described in Case 2 of the proof of Lemma 5.6.

5.2.3 Global-External Interaction

Under the same assumptions as in the previous section but replacing the local relational periodicity by external periodicity we obtain the next interaction theorem. As before, $[n]_q$ is the least positive residue of an integer $n \pmod q$.

Theorem 5.7. *Let p and q be positive integers with $\gcd(p,q) = 1$. Denote $h = 1 + \lfloor \frac{q}{2} \rfloor p$. The bound of global-external interaction for p and q is*

$$B_e(p,q) = \begin{cases} \min(h + [h]_q - 1, h + (q - [h]_q) + 1) & \text{if } q \text{ is odd,} \\ \max(h, h + [h]_q - (p+1)) & \text{if } q \text{ is even.} \end{cases}$$

The proof of the theorem is divided into two lemmata like in the previous sections.

Lemma 5.7. *The bound $B_e(p,q)$ defined in Theorem 5.7 is sufficient.*

Proof. Assume that a word w has a pure period q and a global R-period p. As in the previous lemmata, the word w is a rational power of a word of length q and therefore contains at most q different letters. If one of the letters, say a, is R-compatible with all the other letters, then the word w also has an external relational period 1. Namely, $y = a$ is an external word of w. On the other hand, if this is not the case and the considered alphabet $\mathcal{A}$ does not contain any extra letters not occurring in w, then 1 is not an external R-period. Hence, the existence of such a letter a is crucial for the bound of global-external interaction. We use the definitions of Lemma 5.3 for $\tau(n)$, $S(n)$ and $T(n)$. As in the proof of Lemma 5.3, it suffices to prove (5.2) in order to assure that w_n is related to all letters occurring in the word and, consequently, that 1 is an external R-period of w. Note that in this proof the integer n is fixed, whereas in the proof of Lemma 5.3 inequality (5.2) was proved for all $n \in \{1, 2, \ldots, q\}$.

Consider first the case where q is odd. Denote $B_e = B_e(p,q)$ and suppose that $|w| \geq B_e = h + [h]_q - 1$, where $h = 1 + \frac{q-1}{2}p$. Then the letter $w_h = w_{[h]_q}$ occurring in the positions h and $[h]_q$ is related to all the other letters. Namely, by the definition of B_e, we have $\tau([h]_q) \geq h$ and

$$|S([h]_q) \cup T([h]_q)| = 1 + \left\lfloor \frac{|w| - [h]_q}{p} \right\rfloor + \left\lfloor \frac{\tau([h]_q) - 1}{p} \right\rfloor \geq 1 + \frac{q-1}{2} + \frac{q-1}{2} = q.$$

Hence, (5.2) is satisfied for $n = [h]_q$. Suppose next that $|w| \geq B_e = h + (q - [h]_q) + 1$. Now the letter in position 1 is related to all other letters. Namely, we have $\tau(1) \geq B_e$ and

$$\left\lfloor \frac{|w| - 1}{p} \right\rfloor \geq \left\lfloor \frac{\tau(1) - 1}{p} \right\rfloor \geq \frac{q-1}{2}.$$

Hence, $|S(1) \cup T(1)| \geq 1 + \frac{q-1}{2} + \frac{q-1}{2} = q$. In other words, (5.2) is satisfied for $n = 1$ in this case.

Let us then assume that q is even. Hence $h = 1 + \frac{q}{2}p$. We note first that $\max(h, h + [h]_q - (p+1)) = h$ if and only if $[h]_q \leq p + 1$. If this is the case, we have

$$\left\lfloor \frac{|w| - [h]_q}{p} \right\rfloor \geq \left\lfloor \frac{\frac{q}{2}p + 1 - [h]_q}{p} \right\rfloor \geq \frac{q}{2} - 1.$$

On the other hand, if $[h]_q > p + 1$, we have

$$\left\lfloor \frac{|w| - [h]_q}{p} \right\rfloor \geq \left\lfloor \frac{\frac{q}{2}p + 1 + [h]_q - (p+1) - [h]_q}{p} \right\rfloor = \frac{q}{2} - 1.$$

Furthermore, $\tau([h]_q) \geq h$ in both cases and

$$\left\lfloor \frac{\tau([h]_q) - 1}{p} \right\rfloor \geq \left\lfloor \frac{\frac{q}{2}p + 1 - 1}{p} \right\rfloor = \frac{q}{2}.$$

Thus, (5.2) is satisfied for $n = [h]_q$. □

Lemma 5.8. *The bound $B_e(p, q)$ defined in Theorem 5.7 is strict.*

Proof. In order to prove that our bound is strict, we show that, for some suitable R, there exists a word w of length $B_e(p, q) - 1$ with a period q and with a global period p such that none of its letters is related to all other letters. We use the notation of Lemma 5.7. It suffices to prove that, for every integer $n \in \{1, 2, \ldots, q\}$, the set $S(n) \cup T(n)$ does not contain a complete residue system (mod q), i.e., (5.2) is not satisfied if $|w| = B_e - 1$. This ensures that it is possible to define a relation R on the alphabet $\mathcal{A} = \{a_1, a_2, \ldots, a_q\}$ in such a way that the rational power

$$w = (a_1 a_2 \cdots a_q)^{\frac{B_e - 1}{q}}$$

has a pure period q and a global R-period p, but it does not have $\gcd(p, q) = 1$ as an external R-period. Namely, a letter w_n is R-compatible only with the letters in the positions $S(n) \cup T(n)$ and none of the q different letters is related to all other letters. We consider four cases:

Case 1. Let q be odd and $B_e = h + [h]_q - 1 = \frac{q-1}{2}p + [h]_q$. Assume that $|w| = B_e - 1 = \frac{q-1}{2}p + [h]_q - 1$. Let $1 \leq n \leq q$ and suppose furthermore that $n = [h]_q + ip + j$, where $i \in \mathbb{Z}$ and $0 \leq j < p - 1$. Now

$$\left\lfloor \frac{|w| - n}{p} \right\rfloor = \left\lfloor \frac{\frac{q-1}{2}p + [h]_q - 1 - ([h]_q + ip + j)}{p} \right\rfloor = \frac{q-1}{2} - i - 1.$$

By the definition of the bound, we have $|w| = B_e - 1 \leq h + q - [h]_q$ where $h + q - [h]_q$ is the first position greater than h such that it is congruent to q

modulo q. Hence, for $l \in \{h, h+1, h+2, \ldots, B_e - 1\}$, this implies $[l]_q \geq [h]_q$. Therefore,

$$\tau(n) = \begin{cases} h + ip + j & \text{if } n \in \{1, 2, \ldots, [B_e]_q - 1\}, \\ h - q + ip + j & \text{if } n \in \{[B_e]_q, [B_e]_q + 1, \ldots, q\} \end{cases} \tag{5.11}$$

and moreover,

$$\left\lfloor \frac{\tau(n) - 1}{p} \right\rfloor \leq \left\lfloor \frac{h + ip + j - 1}{p} \right\rfloor = \left\lfloor \frac{\frac{q-1}{2}p + ip + j}{p} \right\rfloor = \frac{q-1}{2} + i.$$

We conclude that the set $S(n) \cup T(n)$ contains at most $1 + (\frac{q-1}{2} - i - 1) + (\frac{q-1}{2} + i) = q - 1$ elements. Hence, it does not form a complete residue system modulo q.

Case 2. Let q be odd and $B_e = h + (q - [h]_q) + 1 = \frac{q-1}{2}p + q + 2 - [h]_q$. Then $|w| = B_e - 1 = \frac{q-1}{2}p + q + 1 - [h]_q$. As above, denote $n = [h]_q + ip + j$, where $i \in \mathbb{Z}$ and $0 \leq j < p-1$. By the assumption, $h + [h]_q - 1 \geq h + q - [h]_q + 1$ and therefore $2[h]_q \geq q + 2$. Thus, we have

$$\begin{aligned} \left\lfloor \frac{|w| - n}{p} \right\rfloor &= \left\lfloor \frac{\frac{q-1}{2}p + q + 1 - [h]_q - ([h]_q + ip + j)}{p} \right\rfloor \\ &\leq \left\lfloor \frac{\frac{q-1}{2}p - 1 - ip - j}{p} \right\rfloor = \frac{q-1}{2} - i - 1. \end{aligned}$$

Since $|w| = B_e - 1 = h + g - [h]_q$, we have (5.11) by the same reasoning as in Case 1. Hence, $\tau(n) \leq h + ip + j$ and

$$\left\lfloor \frac{\tau(n) - 1}{p} \right\rfloor \leq \frac{q-1}{2} + i.$$

This means that (5.2) is not satisfied and, consequently, $S(n) \cup T(n)$ does not contain a complete residue system modulo q for any $n \in \{1, 2, \ldots, q\}$.

Case 3. Let q be even and $|w| = B_e - 1 = h - 1 = \frac{q}{2}p$. For any $n \in \{1, 2, \ldots, q\}$, we have

$$\left\lfloor \frac{|w| - n}{p} \right\rfloor \leq \frac{q}{2} - 1 \quad \text{and} \quad \left\lfloor \frac{\tau(n) - 1}{p} \right\rfloor \leq \frac{q}{2} - 1.$$

Thus, again (5.2) is not satisfied.

Case 4. Let q be even and $|w| = B_e - 1 = h + [h]_q - (p+1) - 1 = \frac{q}{2}p + [h]_q - p - 1$. As in the previous cases, denote $n = [h]_q + ip + j$, where $i \in \mathbb{Z}$ and $0 \leq j < p - 1$. We have

$$\left\lfloor \frac{|w| - n}{p} \right\rfloor = \left\lfloor \frac{\frac{q}{2}p + [h]_q - p - 1 - ([h]_q + ip - j)}{p} \right\rfloor = \frac{q}{2} - i - 2.$$

Next we prove that, for each $l \in \{h, h+1, \ldots, B_e - 1\}$, we have $[l]_q \geq [h]_q$. Let us assume the contrary. Then, for some $l' \in \{h, h+1, \ldots, B_e - 1\}$, we have $[l']_q = 1$. Consider now the number $l' - \frac{q}{2}p$. On one hand,

$$l' - \frac{q}{2}p \equiv l' - \frac{q}{2}p + qp \equiv 1 + \frac{q}{2}p \equiv [h]_q \pmod{q},$$

and on the other hand,

$$l' - \frac{q}{2}p \leq B_e - 1 - \frac{q}{2}p < [h]_q.$$

This is a contradiction. Hence, $[l]_q \geq [h]_q$ for each $l \in \{h, h+1, \ldots, B_e - 1\}$, which means that (5.11) holds and $\tau(n) \leq h + ip + j$ as in Case 1. Thus,

$$\left\lfloor \frac{\tau(n) - 1}{p} \right\rfloor \leq \left\lfloor \frac{\frac{q}{2}p + 1 + ip + j - 1}{p} \right\rfloor = \frac{q}{2} + i,$$

and $|S(n) \cup T(n)| \leq 1 + (\frac{q}{2} - i - 2) + (\frac{q}{2} + i) = q - 1$ for any $n \in \{1, 2, \ldots, n\}$. □

Let us now consider an example of the global-local interaction.

Example 5.8. As in the previous example, we continue to study interactions of periods $p = 5$ and $q = 7$. Thus, $h = 1 + \lfloor \frac{q}{2} \rfloor p = 16$, $[h]_7 = 2$ and $B_e(p, q) = \min(h + [h]_7 - 1, h + (q - [h]_7) + 1) = \min(17, 22) = 17$. Let w be a word of length $B_e(p, q) - 1 = 16$ with global R-period p and pure period q but not having 1 as an external period. In Figure 5.2, we represent the word w in a table form and the graph of relations as in Example 5.6.

w_1	w_2	w_3	w_4	w_5
w_6	w_7	w_1	w_2	w_3
w_4	w_5	w_6	w_7	w_1
w_2	$\not w_3$			

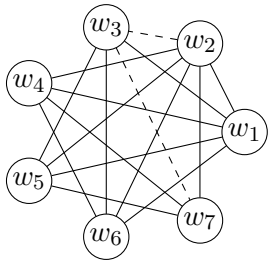


Figure 5.2: Global-external interaction for $p = 5$ and $q = 7$.

The dashed lines in the graph represent the situation where the length of w is increased by 1, i.e., the 17th letter indicated by $\not w_3$ is taken into consideration. Hence, we notice that if $|w| = 16$, then no letter w_i, $i = 1, 2, \ldots, 7$ is necessarily related to all other letters. However, if $|w| = 17$, then $w_2 \, R \, w_3$ and every element w_i becomes connected to w_2. Consequently, w has an external period 1, since in this case $y = w_2$ is an external word of w.

5.2.4 External Interactions

In the previous three sections we found interaction bounds for one pure period and one global relational period. On the other hand, Example 5.5 shows that if we replace the global period by a local period such bounds do not necessarily exist. Is this also true if the global period is replaced by an external period?

Let us assume that a word w has a pure period q and an external period p. Let $y = y_1 \cdots y_p$ be an external word of w, i.e., for every $j \in \{1, 2, \ldots, p\}$, $y_j \, R \, w_i$ whenever $i \equiv j \pmod{p}$. Denote by $\mathrm{Alph}(w)$ the set of the letters occurring in w. The succeeding example shows that some conditions on the letters of the external word are needed in order to get bounds for external-global and external-local interactions.

Example 5.9. Let $R = \langle (a, b), (b, c) \rangle$ be a similarity relation on the three letter alphabet $\{a, b, c\}$. Consider the infinite word $w = (a^{q-1}c)^\omega$ for an integer $q \geq 2$ and choose p such that $\gcd(p, q) = 1$. Clearly any p is an external R-period of w, since b is related to both a and c. However, 1 is neither a global nor a local R-period of w.

Hence, the example implies the following.

Theorem 5.8. *No finite bounds exist for external-global and external-local interactions.*

Because of this, in the formulation of the next theorem we consider only external periods satisfying a special condition.

Definition 5.3. An external period p of a word w is called *holding* if there exists an external word $y = y_1 \cdots y_p$ of w satisfying

$$|\mathrm{Alph}(w) \setminus \mathrm{Alph}(y)| \leq 1. \tag{5.12}$$

By restricting considerations to the holding external periods it is possible to find a bound of interaction.

Theorem 5.9. *Let p and q be positive integers with $\gcd(p, q) = 1$. The bound of external-global interaction $C_g(p, q)$ for a holding external period p and a pure period q is pq. Similarly, the bound of external-local interaction $C_l(p, q)$ for a holding external period p and a pure period q is pq.*

Proof. Suppose that w is of length pq and it has a pure period q and a holding external period p. Let $y = y_1 \cdots y_p$ be an external word of w satisfying (5.12). Consider a letter w_n in position $n \in \{1, 2, \ldots, q\}$. Since q is a period of w, the letter w_n occurs in positions $n + iq$ for $i = 0, 1, \ldots, p-1$. These positions form a complete residue system (mod p), which means that w_n is related to

all letters in Alph(y) by the external period p. By (5.12), there may exist only one letter in Alph(w) such that it does not occur in y. If this letter is w_n, then it is trivially related to itself and therefore to all letters in Alph(w). On the other hand, if $w_n \in \text{Alph}(y)$, then there exists a position k such that $y_k = w_n$. Now y_k is related to letters in positions $k+jp$ for $j = 0, 1, \ldots, q-1$, and these positions form a complete residue system (mod q). Hence, $y_k = w_n$ is related to all letters of w. Since the above considerations hold for all $n = 1, 2, \ldots, q$, all letters in Alph(w) are compatible with all other letters. Hence, 1 is a global and therefore also a local period of w.

Modification of the previous example shows that the bound $C_g(p,q) = C_l(p,q) = pq$ is strict. Assume that R is like in Example 5.9 and

$$w = (a^{q-1}c)^{p-1}a^{q-1}.$$

We may choose $y = b^{p-1}a$. Namely, $y_p = a$ must only be related to letters in positions $p + ip$ for $i = 0, 1, \ldots, q-2$, which are all a's. Hence, w has an external word which satisfies (5.12), but 1 is neither a local nor a global R-period of w. □

For the bound of external-external interaction, additional conditions like (5.12) on the relational period p are not necessary.

Theorem 5.10. *Let p and q be positive integers with* $\gcd(p,q) = 1$. *The bound of external-external interaction for p and q is $C(p,q) = 1 + (q-1)p$.*

Proof. Assume that $y = y_1 \cdots y_p$ is an external word of w. Clearly, if $|w| \geq C(p,q)$, then y_1 is related to all letters in Alph(w). Namely, the set $\{1 + ip \mid i = 0, 1, \ldots, q-1\}$ is a complete residue system (mod q).

In order to prove that this bound is strict, consider the rational power

$$w = (a_1 \cdots a_q)^{(C(p,q)-1)/q}$$

with q different letters $a_1, \ldots, a_q$. Furthermore, let us assume that the alphabet $\mathcal{A}$ under consideration has p extra letters not occurring in w. Suppose that these letters are $y_1, \ldots, y_p$. We define that y_k, where $k \in \{1, 2, \ldots, p\}$, is not related to the letter $a_{[k+(q-1)p]_q}$, but it is related to all letters w_{k+ip} for $i = 0, 1, \ldots, q-2$. Note that the length of w and the assumption that w has q different letters ensures that this is well defined. Hence, $y = y_1 \cdots y_p$ is an external word of w. Furthermore, we may assume that letters in Alph(w) are not compatible with each other. Hence, no letter in the alphabet $\mathcal{A}$ is related to all letters in Alph(w). Therefore, the word w does not have 1 as an external R-period. □

Of course, we may as well restrict our considerations to holding external periods like in Theorem 5.9.

Theorem 5.11. *Let p and q be positive integers with* $\gcd(p,q) = 1$. *Then the bound of external-external interaction for a holding period p and a pure period q is*

$$C_e(p,q) = \begin{cases} (p-1)q+1 & \text{if } p \text{ is even and } q > p, \\ (q-1)p+1 & \text{otherwise.} \end{cases}$$

Theorem 5.11 is a direct consequence of the succeeding Lemmata 5.9 and 5.10.

Lemma 5.9. *The bound $C_e(p,q)$ defined in Theorem 5.11 is sufficient.*

Proof. First of all, Theorem 5.10 implies that the bound $(q-1)p+1$ is sufficient for all external periods. Hence, let us consider the remaining case such that a word w of length $|w| \geq (p-1)q+1$ has a holding external R-period p and a pure period q, where p is even and $q > p$. Let $y = y_1 \cdots y_p$ be an external word satisfying (5.12). Suppose also that $\gcd(p,q) = 1$ is not an external R-period of w.

Since w has a pure period q, it is of the form $(w_1 \cdots w_q)^{p-1} w_1$. Denote $v = v_1 \cdots v_{pq} = (w_1 \cdots w_q)^p$. For $1 \leq i \leq p$, set

$$\mathcal{W}_i = \{w_j \mid j \equiv i \pmod{p}\} \quad \text{and} \quad \mathcal{W} = \mathrm{Alph}(w). \tag{5.13}$$

We also denote

$$\mathcal{V}_i = \{v_j \mid j > (p-1)q+1,\ j \equiv i \pmod{p}\}. \tag{5.14}$$

Note that $\mathcal{W} \setminus \mathcal{W}_i \subseteq \mathcal{V}_i$, since $|v| = pq$. The notation $a\,R\,\mathcal{Y}$ means that the letter a is compatible with all letters in the set $\mathcal{Y}$. For example, the ith letter y_i of the external word y is, by the definition of an external word, compatible with all letters in $\mathcal{W}_i$, i.e., $y_i\,R\,\mathcal{W}_i$. Note that if also $y_i\,R\,\mathcal{V}_i$, then $y_i\,R\,\mathcal{W}$ and y_i is an external word of w, which contradicts the assumption that 1 is not an external period of w. Furthermore, for each $a = v_m \in \mathcal{V}_i$, we have $a = w_{[m]_q+nq}$ for $n = 1, 2, \ldots, p-1$ and $a \in \mathcal{W}_j$ for every $j \neq i$.

We make two observations. Firstly, we conclude that

$$y_i \neq y_j \text{ for } i \neq j. \tag{5.15}$$

Otherwise, we have $\mathcal{V}_i \subseteq \mathcal{W}_j$ and therefore $y_i = y_j\,R\,\mathcal{V}_i$. Hence, $y_i\,R\,\mathcal{W}$ and we get a contradiction the same way as above.

Secondly, we note that

$$\mathcal{V}_i \setminus \{y_i\} \neq \emptyset. \tag{5.16}$$

Namely, if $\mathcal{V}_i = \{y_i\}$, then $y_i\,R\,\mathcal{V}_i$ by the reflexivity of R and, consequently, $y_i\,R\,\mathcal{W}$. As above, this is a contradiction.

Next we show that $w_1 = y_k$ for some $k = 1, 2, \ldots, p$. By the structure of w, we have

$$w_1 \in \mathcal{W}_i, \quad i = 1, 2, \ldots, p. \tag{5.17}$$

If w_1 does not occur in y, then (5.12) implies that $\mathrm{Alph}(y) = \mathcal{W} \setminus \{w_1\}$. Hence, by the definition of an external word, we have $w_1 \, R \, \mathrm{Alph}(y)$. Since R is reflexive, this means that $w_1 \, R \, \mathcal{W}$. Again, we end up in a contradiction.

It also follows that there exists exactly one letter $a \in \mathcal{V}_k$ such that a does not occur in y. Namely, there cannot exist two such letters since this would contradict the holding property (5.12). Suppose next that all letters of $\mathcal{V}_k$ occur in y. By (5.17), we have $y_k = w_1 \, R \, \mathcal{V}_k$, which implies $w_1 \, R \, \mathcal{W}$. This is a contradiction. Hence, we have $a \in \mathcal{V}_k$. Moreover, it holds that

$$a \notin \mathcal{V}_i \text{ for } i \neq k. \tag{5.18}$$

Let us assume, on the contrary, that $a \in \mathcal{V}_i$ for some $i \neq k$. Then $a \in \mathcal{W}_j$ for all $j = 1, 2, \ldots, p$, since $a \in \mathcal{W}_j$ for every $j \neq i$ and $a \in \mathcal{W}_j$ for every $j \neq k$. Thus, $a \, R \, \mathcal{W}$, since $a \, R \, y_j$ for all j and $\mathrm{Alph}(y) = \mathcal{W} \setminus \{a\}$. Again, this contradicts the fact that 1 is not an external R-period of w.

Suppose next that there exists a letter $b \in \mathcal{V}_k$ such that $b = y_i$ for some $i \neq k$. Consider a letter $c \neq b$ belonging to $\mathcal{V}_i$. Note that by (5.16) such c exists and by (5.18) $c \neq a$. Hence, there must exist an index j such that $y_j = c$. Since $b \in \mathcal{V}_k$, we have $b \in \mathcal{W}_l$ for all $l \neq k$, especially for $l = j$. Therefore, $y_j = c \, R \, b$. Since this holds for all letters of $\mathcal{V}_i$, we conclude that $b \, R \, \mathcal{W}$. Again we end up in a contradiction and we may deduce that

$$\mathcal{V}_k = \{a\}. \tag{5.19}$$

Consider now the letter $y_i = x \neq w_1$ in some position $i \neq k$. By (5.16) and by (5.18), we have $\mathcal{V}_i \setminus \{x, a\} \neq \emptyset$. Moreover, there exists at least one letter $z \in \mathcal{V}_i \setminus \{x\}$ such that if $y_j = z$ then $x \in \mathcal{V}_j$. Otherwise, $x \in \mathcal{W}_j$ and $y_j = z \, R \, x$. If $x \notin \mathcal{V}_j$ for any j such that $y_j = z \in \mathcal{V}_i \setminus \{x\}$, then $x \, R \, \mathcal{V}_i$ and, consequently, $x \, R \, \mathcal{W}$, which again implies a contradiction. Suppose next that there exists another letter $z' \in \mathcal{V}_i \setminus \{x\}$ in a position j' of y such that $x \in \mathcal{V}_{j'}$. This implies that $x \in \mathcal{W}_l$ for all $l \neq j'$, especially for $l = j$. Hence, $x \, R \, y_j = z$. Since this holds for all $z \in \mathcal{V}_i \setminus \{x\}$, we have $x \, R \, \mathcal{V}_i$, which is a contradiction. Therefore, the letter z must be unique. In other words, we have

$$\mathcal{V}_i \setminus \{x\} = \{z\}. \tag{5.20}$$

By the equations (5.15), (5.19) and (5.20) we conclude that, for a letter y_i where $i \neq k$, there exists a unique index j such that $\mathcal{V}_i \setminus \{y_i\} = \{y_j\}$ and $\mathcal{V}_j \setminus \{y_j\} = \{y_i\}$. Hence, there must be a partition of integers $\{1, 2, \ldots, p\} \setminus \{k\}$ into pairs, i.e., subsets of cardinality two. Since p is even, this is impossible. Hence, $\gcd(p, q) = 1$ must be an external period of w. □

Lemma 5.10. *The bound $C_e(p,q)$ defined in Theorem 5.11 is strict.*

Proof. We adopt the notation of Lemma 5.9. Recall especially definitions (5.13). For each k, also denote

$$k' = [(q-1)p + k]_q. \tag{5.21}$$

Let $C_e = C_e(p,q)$. In the sequel, we consider four cases. In each case, we show that it is possible to define a relation R, a word w with period q and of length $C_e - 1$ and an external word $y = y_1 \cdots y_p$ of w in such way that no letter in the alphabet $\mathcal{A}$ under consideration is related to all letters in $\mathrm{Alph}(w)$, and in addition, y satisfies $y_i \, R \, \mathcal{W}_i$ for $1 \le i \le p$ and $|\mathrm{Alph}(w) \setminus \mathrm{Alph}(y)| \le 1$. These properties imply that w has a holding external period p, but 1 cannot be an external period.

Case 1. Assume that $q < p$ and q is even. Then $C_e = (q-1)p + 1$. Set $\mathcal{A} = \{a_1, \ldots, a_q\}$ and

$$w = (a_1 \cdots a_q)^{(C_e - 1)/q}. \tag{5.22}$$

Since q is even, we can make a partition P of the set $\{k' \mid k = 1, 2, \ldots, q\} = \{1, 2, \ldots, q\}$ into pairs, i.e., subsets of cardinality two. If m and n belong to the same subset in P, we denote $(m, n) \in P$ and let

$$(a_m, a_n) \notin R. \tag{5.23}$$

Let these be the only R-incompatible pairs. Hence, each letter in $\mathcal{A}$ is R-incompatible with exactly one other letter in $\mathrm{Alph}(w)$.

Taking benefit of the partition P, we set for every $i, j \in \{1, 2, \ldots, q\}$ satisfying $(i', j') \in P$ that

$$y_i = a_{j'} \quad \text{and} \quad y_j = a_{i'}. \tag{5.24}$$

Then $y_i = a_{j'} \, R \, \mathcal{W}_i = \mathcal{A} \setminus \{a_{i'}\}$ for $i \in \{1, 2, \ldots, q\}$. Furthermore, set $y_i = y_{[i]_q}$ for $i = q+1, q+2, \ldots, p$. Note that $\mathcal{W}_i \subseteq \mathcal{W}_{[i]_q}$. Namely, if $i = [i]_q + tq \le p$, then

$$\mathcal{W}_i = \{w_j \mid j \equiv [i]_q + tq \pmod p\} \subseteq \{w_{j-tq} \mid j - tq \equiv [i]_q \pmod p\} = \mathcal{W}_{[i]_q},$$

since q is a period of w. Hence, $y_i \, R \, \mathcal{W}_i$ for all $i = 1, 2, \ldots, p$, and $\mathrm{Alph}(y) = \mathrm{Alph}(w)$.

Case 2. Assume that $q < p$ and q is odd. We have $C_e = (q-1)p + 1$. Let $\mathcal{A} = \{a_1, \ldots, a_q, b, c\}$ and set w as in (5.22). Assume that r,s and t are three different integers in $\{1, 2, \ldots, q\}$, where $s = [(q-1)p + (q+1)]_q$. Since $q - 3$ is even, we can make a partition P of the set $\{1, 2, \ldots, q\} \setminus \{r, s, t\}$ into pairs. Define R-incompatible pairs by (5.23) for indices $\{1, 2, \ldots, q\} \setminus \{r, s, t\}$. Let also $(a_r, a_s), (a_r, b), (a_s, a_t), (a_t, c) \notin R$. Hence, no letter in $\mathcal{A}$ is compatible with all letters of $\mathrm{Alph}(w)$.

Use (5.24) to determine the letters y_i, where $i = 1, 2, \ldots, q$ and $i' \in \{1, 2, \ldots, q\} \setminus \{r, s, t\}$. Furthermore, for $i = 1, 2, \ldots, q$, set

$$y_i = \begin{cases} b & \text{if } i' = r, \\ a_r & \text{if } i' = s, \\ c & \text{if } i' = t. \end{cases}$$

Set also $y_{q+1} = a_t$ so that a_s is the only letter in $\mathrm{Alph}(w)$ not occurring in the external word y. Hence, $|\mathrm{Alph}(w) \setminus \mathrm{Alph}(y)| \leq 1$. For $i = q+2, q+3, \ldots, p$, set $y_i = y_{[i]_q}$ like in Case 1. We may assume that there are no more incompatible pairs than those mentioned above. Therefore, $y_i \, R \, \mathcal{W}_i$ for all i, since $\mathcal{W}_i = \mathrm{Alph}(w) \setminus \{a_{i'}\}$. Especially, $y_{q+1} = a_t \, R \, \mathcal{W}_{q-1}$, where $\mathcal{W}_{q-1} = \mathrm{Alph}(w) \setminus \{a_s\}$.

Case 3. Assume that $q > p$ and p is odd. Then $C_e = (q-1)p + 1$. Let the alphabet be $\mathcal{A} = \{a, a_{q-p+1}, a_{q-p+2}, \ldots, a_q\}$ and set

$$w = (a^{q-p} a_{q-p+1} a_{q-p+2} \cdots a_q)^{(C_e - 1)/q}.$$

Since p is odd, we can partition the set $\{q-p+1, q-p+2, \ldots, q-1\}$ and make $(p-1)/2$ incompatible pairs using (5.23). Additionally, set $(a_q, a) \notin R$. Assume moreover that these are the only R-incompatible pairs. Again, each letter is incompatible with exactly one other letter. Since $i' = q - p + i$ for all $i = 1, 2, \ldots, p$, we may define $y_1 \cdots y_{p-1}$ using (5.24). Furthermore, set $y_p = a$. Now $y_i \, R \, \mathcal{W}_i$ for $i = 1, 2, \ldots, p$. Especially, $y_p = a \, R \, \mathcal{W}_p = \mathcal{A} \setminus \{a_q\}$ and $\mathrm{Alph}(w) \setminus \mathrm{Alph}(y) = \{a_q\}$.

Case 4. Assume that $q > p$ and p is even. We have $C_e = (p-1)q + 1$. Consider a word $(w_1 \cdots w_q)^{p-1}$ where $w_i = w_j$ if $1 \leq i, j \leq q$ and $i \equiv j \pmod{p}$. Assume also that $\mathcal{A} = \mathrm{Alph}(w)$. We make a partition P of the set $\{q-p+1, q-p+2, \ldots, q\}$ into pairs. Note that the set has an even number of elements. Define R-incompatible pairs by (5.23) and let these be the only R-incompatible pairs. Since $\{w_{q-p+1}, w_{q-p+2}, \ldots, w_q\} = \mathrm{Alph}(w) = \mathcal{A}$, no letter is compatible with $\mathrm{Alph}(w)$. As above, $i' = q-p+i$ for all $i = 1, 2, \ldots, p$ and we may define $y_1 \cdots y_p$ using (5.24). Now we have $y_i \, R \, \mathcal{W}_i$ for all i. Moreover, we have $\mathrm{Alph}(y) = \mathrm{Alph}(w)$. □

On the other hand, it might be more interesting to consider the case where the external word of w consists only of letters occurring in w.

Definition 5.4. An external period p of a word w is called *inclusive* if there exists an external word $y = y_1 \cdots y_p$ of w satisfying

$$\mathrm{Alph}(y) \subseteq \mathrm{Alph}(w). \tag{5.25}$$

Using this definition we have one more result concerning the external-external interaction.

Theorem 5.12. *Let p and q be positive integers with* $\gcd(p,q) = 1$. *Then the bound of external-external interaction for an inclusive external period p and a pure period q is*

$$\overline{C}(p,q) = \begin{cases} (q-2)p + (q-1) & \text{if } q \text{ is odd and } q \leq p+1, \\ (q-1)p + 1 & \text{otherwise.} \end{cases}$$

Lemma 5.11 and Lemma 5.12 imply Theorem 5.12.

Lemma 5.11. *The bound $\overline{C}(p,q)$ defined in Theorem 5.12 is sufficient.*

Proof. First of all, Theorem 5.10 implies that the bound $(q-1)p+1$ is sufficient for all external periods. Hence, let us consider the remaining case such that a word w of length $|w| \geq (q-2)p+(q-1)$ has an inclusive external period p and a pure period q, where q is odd and $q \leq p+1$.

We use the notation of Lemmata 5.9 and 5.10. Recall especially that $\mathcal{W}_i = \{w_j \mid j \equiv i \pmod p\}$, $\mathcal{W} = \mathrm{Alph}(w)$ and $k' = [(q-1)p+k]_q$. Hence, by the definition of an external word, $y_k \, R \, \mathcal{W}_k$. Furthermore, set $U = \{1, 2, \ldots, q-1\}$. Note that since $q-1 \leq p$, the set $\mathcal{W}_k$ is defined for all $k \in U$. Moreover, if $k \in U$, then $|w| - k \geq (q-2)p$. Since $\gcd(p,q) = 1$ and q is a pure period of w, this implies that

$$w_m \in \mathcal{W}_k = \left\{ w_{k+ip} \;\middle|\; i = 0, 1, \ldots, \left\lfloor \frac{|w|-k}{p} \right\rfloor \right\} \tag{5.26}$$

for any $m \not\equiv k' \pmod q$. Since $y_k \, R \, \mathcal{W}_k$, it follows that

$$y_k \, R \, w_m \text{ if } m \not\equiv k' \pmod q. \tag{5.27}$$

Next we state three important properties, which will be needed throughout the proof: (i) If $k \in U$ and $y_k = w_{k'}$, then $y_k \, R \, \mathcal{W}$; (ii) If there exist $k, l \in U$ $(k \neq l)$ such that $y_k = y_l$, then $y_k \, R \, \mathcal{W}$; (iii) If there exist $k, l \in U$ $(k \neq l)$ such that $y_l = w_{k'}$ and $y_k \in \mathcal{W} \setminus \{w_{l'}\}$, then $y_k \, R \, \mathcal{W}$.

The first statement follows directly from (5.27), since the similarity relation R is reflexive.

Next, consider the second property. By (5.27), we have $y_k \, R \, (\mathcal{W} \setminus \{w_{k'}\})$ and $y_l \, R \, (\mathcal{W} \setminus \{w_{l'}\})$. Now $k' \neq l'$, since $k, l \in \{1, 2, \ldots, q-1\}$. Hence, $y_k \, R \, w_{l'}$ and $y_l \, R \, w_{k'}$ by (5.27). Since $y_k = y_l$, we have $y_k \, R \, \mathcal{W}$.

Finally, consider (iii). Again, $y_k \, R \, (\mathcal{W} \setminus \{w_{k'}\})$ and $y_l \, R \, (\mathcal{W} \setminus \{w_{l'}\})$. Since $y_k \in \mathcal{W} \setminus \{w_{l'}\}$, we have $y_l = w_{k'} \, R \, y_k$, which implies that $y_k \, R \, \mathcal{W}$. Note that, if $k, l \in U$ $(k \neq l)$, $y_l = w_{k'}$ and $y_k = w_{l'}$, then relations $y_k \, R \, \mathcal{W}_k$ and $y_l \, R \, \mathcal{W}_l$ do not imply $w_{k'} \, R \, w_{l'}$.

If any of the assumptions of (i)–(iii) is satisfied, then the word w necessarily has an external period 1. Namely, $y = y_k$ is an external word of w. Thus, from now on we assume that none of them is satisfied.

Assume first that, at least for one index $k \in U$, the letter in the position k' occurs also in another position $1 \leq n \leq q$. Denote $w_{k'} = w_n = a$. Since $\mathcal{W}_k$ must contain a letter which is in a position congruent to n, we have $a \in \mathcal{W}_k$ and $\mathcal{W}_k = \mathcal{W}$. Thus, $y_k \, R \, \mathcal{W}$ and 1 is an external period of w.

Finally, assume that, for each $k \in U$, the letter $w_{k'}$ occurs only in positions congruent to k' (mod q). This means that all letters $w_{k'}$ $(1 \leq k \leq q)$ are different. Moreover, this implies that $\mathrm{Alph}(y_1 \cdots y_{q-1}) = \mathcal{W} \setminus \{w_{s'}\}$ for some $1 \leq s \leq q$, since $\mathrm{Alph}(y_1 \cdots y_{q-1}) \subseteq \mathcal{W}$ by (5.25) and all letters in $\mathrm{Alph}(y_1 \cdots y_{q-1})$ are different by (ii).

Suppose now that $s = q$. Since $q' \notin \{k' \mid k \in U\}$, we have $y_k \, R \, w_{q'}$ for $1 \leq k \leq q-1$ by (5.27). Since $\mathrm{Alph}(y_1 \cdots y_{q-1}) = \mathcal{W} \setminus \{w_{q'}\}$, it follows that $w_{q'} \, R \, (\mathcal{W} \setminus \{w_{q'}\})$. By the reflexivity of R, we have $w_{q'} \, R \, \mathcal{W}$ and $y = w_{q'}$ is an external word of w.

Moreover, the case $s \neq q$ is impossible. This is based on the fact that $q-1$ is even. Indeed, assume that $y_r = w_{q'}$ for some $r \in U$. Now consider $n \in U \setminus \{r\}$. Since $\mathrm{Alph}(y_1 \cdots y_{q-1}) \subseteq \{w_{k'} \mid 1 \leq k \leq q\}$, we have $y_n = w_{m'}$ for some $m \in \{1, 2, \ldots, q\}$. Since the letters $\{w_{k'} \mid 1 \leq k \leq q\}$ are distinct, the integer m is unique. In addition, $m \neq n$ by (i) and $m \neq q$ by (ii). Furthermore, $y_m = w_{n'}$ by (iii) and therefore $m \neq r$, since $w_{q'} \notin \{w_{k'} \mid k \in U\}$. Thus, $m \in U \setminus \{r, n\}$. Since the set $U \setminus \{r\}$ has odd number $q-2$ elements, there cannot be such unique m for each n. This is a contradiction. Hence, we have shown that $\gcd(p, q) = 1$ is an external period of w. □

Lemma 5.12. *The bound $\overline{C}(p, q)$ defined in Theorem 5.12 is strict.*

Proof. Let p and q be positive integers with $\gcd(p, q) = 1$. Denote $\overline{C} = \overline{C}(p, q)$ and adopt the notation of Lemma 5.9 and Lemma 5.10. In this proof, we want to define a relation R, a word w with period q and of length $\overline{C} - 1$ and an external word $y = y_1 \cdots y_p$ of w in such way that no letter in the alphabet $\mathcal{A}$ under consideration is related to all letters in $\mathrm{Alph}(w)$, and in addition, y satisfies $y_i \, R \, \mathcal{W}_i$ for $1 \leq i \leq p$ and $\mathrm{Alph}(y) \subseteq \mathrm{Alph}(w)$.

Consider first the situation where q is odd and $q < p$. Hence, $\overline{C} = (q-2)p + (q-1)$. We set $\mathcal{A} = \{a_1, \ldots, a_q\}$ and

$$w = (a_1 \cdots a_q)^{(\overline{C}-1)/q}.$$

Case 1. Assume that $q < p$, q is odd and neither $p+1$ nor $p-1$ is divisible by q. Denote

$$\begin{array}{rclcrcl} a & = & a_{[(q-2)p+(q-1)]_q}, & & b & = & a_{[(q-2)p+q]_q}, \\ c & = & a_{[(q-1)p+(q-1)]_q}, & & d & = & a_{[(q-1)p+q]_q}. \end{array}$$

Note that by the above divisibility properties, all these four letters are different. Now $\{a_{i'} \mid i = 1, 2, \ldots, q-2\} = \mathcal{A} \setminus \{c, d\}$. Hence, there exist numbers

$k, l \in \{1, 2, \ldots, q-2\}$ such that $a_{k'} = a$ and $a_{l'} = b$. We make a partition P of the set $\{i' \mid i \in \{1, 2, \ldots, q-2\},\ i \neq l\}$ into pairs. This is possible since the set contains an even number $q-3$ of elements. We use (5.23) to define R-incompatible pairs of P and, in addition, we set $(b, c) \notin R$ and $(b, d) \notin R$. Let these be the only incompatible pairs. Hence, except for b, all other letters are R-incompatible with exactly one other letter. Now consider an external word $y = y_1 \cdots y_p$. For indices $i \in \{1, 2, \ldots, q-2\} \setminus \{l\}$, use (5.24) as before. In addition, set $y_l = c$, $y_{q-1} = y_k$ and $y_q = d$. Moreover, as in Case 1 of Lemma 5.10, set $y_i = y_{[i]_q}$ for $i = q+1, q+2, \ldots, p$. Now

$$\begin{aligned} y_l = c\, R\, \mathcal{W}_l &= \mathcal{A} \setminus \{b\}, \\ y_{q-1} = y_k\, R\, \mathcal{W}_{q-1} &= \mathcal{A} \setminus \{a, c\}, \\ y_q = d\, R\, \mathcal{W}_q &= \mathcal{A} \setminus \{b, d\}, \end{aligned}$$

and $y_i\, R\, \mathcal{W}_i$ by (5.24) for all the other indices $i \in \{1, 2, \ldots, q-2\} \setminus \{l\}$.

Case 2. Assume that $q < p$, q is odd and $p + 1 \equiv 0 \pmod{q}$. We use the same notation as in Case 1. Since $p + 1 \equiv 0 \pmod{q}$, $a = d$. Clearly $b \notin \{c, a\}$. Now $\{a_{i'} \mid i = 1, 2, \ldots, q-2\} = \mathcal{A} \setminus \{c, a\}$. Thus, there does not exist $k \in \{1, 2, \ldots, q-2\}$ such that $a_{k'} = a$, but we have l like in Case 1. Define the relation R and the external word y as in Case 1 except that now $y_{q-1} = b$. Hence, no letter is related to all other letters occurring in w and y is well defined. Namely,

$$y_{q-1} = b\, R\, \mathcal{W}_{q-1} = \mathcal{A} \setminus \{a, c\}.$$

Case 3. Assume that $q < p$, q is odd and $p - 1 \equiv 0 \pmod{q}$. Using the notation of Case 1, we conclude that $b = c$. Clearly $a \notin \{b, d\}$. Now we have $\{a_{i'} \mid i = 1, 2, \ldots, q-2\} = \mathcal{A} \setminus \{b, d\}$. Hence, using the notation of Case 1, there exists k but no l in $\{1, 2, \ldots, q-2\}$. This time we make a partition P of the set $\{i' \mid i = \{1, 2, \ldots, q-2\},\ i \neq k\}$ into subsets of cardinality two. Define R such that it satisfies (5.23) and furthermore that $(a, b) \notin R$ and $(a, d) \notin R$. Assume again that these are the only R-incompatible pairs. In addition to (5.24) set $y_k = b$, $y_{q-1} = b$ and $y_q = a$. Again no letter is compatible with all other letters and y is well defined, since

$$\begin{aligned} y_k = b\, R\, \mathcal{W}_k &= \mathcal{A} \setminus \{a\}, \\ y_{q-1} = b\, R\, \mathcal{W}_{q-1} &= \mathcal{A} \setminus \{a, b\}, \\ y_q = a\, R\, \mathcal{W}_q &= \mathcal{A} \setminus \{b, d\}. \end{aligned}$$

Case 4. Next assume that $q > p + 1$ and p is even. Hence, $\overline{C} = (q-1)p + 1$. Set $\mathcal{A}$ and w as in the previous cases. Make a partition P of the set $\{i' \mid i = 1, 2, \ldots, p\} = \{q-p+1, q-p+2, \ldots, q\}$ into pairs. Define R-incompatible pairs by (5.23) and use (5.24) to define the external word y. Since $q - p \geq 2$, we also set $(a, a_1) \notin R$ for each $a \in \{a_2, \ldots, a_{q-p}\}$. Hence, no

letter is R-compatible with all other letters of $\mathrm{Alph}(w)$. Moreover, $y_i \, R \, \mathcal{W}_i$ for all i.

In all other cases we may use the constructions in Cases 1, 3 and 4 of Lemma 5.10. Note that the external words in these cases satisfy the condition $\mathrm{Alph}(y) \subseteq \mathrm{Alph}(w)$. Note also that if q is odd and $q = p+1$, then $\overline{C}(p,q) = (q-2)p + (q-1) = (p-1)q + 1 = C_e(p,q)$ and the construction in Case 4 is suitable for our purposes. Hence, we have shown that in all cases there exists a word w of length $\overline{C} - 1$ such that it has a pure period q and an external word $y = y_1 \cdots y_p$ but 1 is not an external R-period of w. Moreover, the external word y satisfies (5.25) in every case. □

We end this section by giving an example of the external-external interaction in the holding and inclusive case.

Example 5.10. For $p = 5$ and $q = 7$, we have $C_e(p,q) = (q-1)p + 1 = 31$. Let w be a word of length $C_e(5,7) - 1 = 30$ and with holding external period p and pure period q but without external period 1. As in the examples of the previous sections, we write w in a table; see Figure 5.3. At the top of the table we write an external word $y = y_1 \cdots y_5$ and at the bottom of the table we write the next 5 letters of the ultimately q-periodic word.

y_1	y_2	y_3	y_4	y_5
w_1	w_2	w_3	w_4	w_5
w_6	w_7	w_1	w_2	w_3
w_4	w_5	w_6	w_7	w_1
w_2	w_3	w_4	w_5	w_6
w_7	w_1	w_2	w_3	w_4
w_5	w_6	w_7	w_1	w_2
$\not\wr_3$	$\not\wr_4$	$\not\wr_5$	$\not\wr_6$	$\not\wr_7$

Figure 5.3: External-external interaction for $p = 5$ and $q = 7$.

Note that the interpretation of the table is different than in the case of global period p. Here y_i is related to all letters in the ith column except the entry $\not\wr_{i'}$ in the last row of the table. Note that the external period itself does not force any relations between the letters of the word w. From Figure 5.3 we immediately see that increasing the length of w by 1, the letter y_1 becomes necessarily related to all letters of w. In that case, 1 is an external period with an external word y_1.

Let us now give an example of the word w using the construction given in Case 3 of the proof of Lemma 5.10. First we make a partition of the set $\{q-p+1, q-p+2, \ldots, q-1\} = \{3,4,5,6\}$. Let $P = \{(3,4),(5,6)\}$. Hence, we set $w = (aaa_3a_4a_5a_6a_7)^4aa$ and define $(a_3,a_4),(a_5,a_6) \notin R$ by (5.23) and,

in addition, set $(a, a_7) \notin R$. Assume that these are the only incompatible pairs. For the letters y_i, $i = 1, 2, 3, 4$, of an external word y, we use (5.24). For example, $i = 1$, $i' = [(q-1)p + i]_7 = [31]_7 = 3$, $j = 2$ and $j' = [(q+1)p + j] = [32]_7 = 4$. Thus, $y_1 = a_4$ and $y_2 = a_3$. In addition, setting $y_5 = a$ we get $y = a_4 a_3 a_6 a_5 a$. Hence, w has an holding external period 5 by (5.12). Moreover, $\overline{C}(5, 7) = 31$ and the external word y of w satisfies also (5.25). The word w, the external word y, and relations between the letters are represented in Figure 5.4.

a_4	a_3	a_6	a_5	a
a	a	a_3	a_4	a_5
a_6	a_7	a	a	a_3
a_4	a_5	a_6	a_7	a
a	a_3	a_4	a_5	a_6
a_7	a	a	a_3	a_4
a_5	a_6	a_7	a	a
$\not{a}_3$	$\not{a}_4$	$\not{a}_5$	$\not{a}_6$	$\not{a}_7$

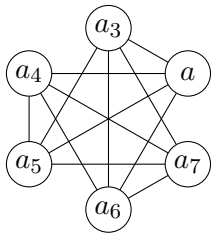


Figure 5.4: Example of a 7-periodic word with holding external word of length 5 but without external relational period 1.

5.2.5 Local Interactions

Despite the negative result in Example 5.5 there exist interaction bounds for some integers p and q also in the case where p is local. If no bound $B(p, q)$ of interaction for p and q exists, we set $B(p, q) = \infty$.

Theorem 5.13. *Let p and q be positive integers with* $\gcd(p, q) = 1$. *Then the bound of local-local interaction for p and q is*

$$D_l(p, q) = \begin{cases} p + q & \text{if } p - 1 \equiv 0 \pmod{q} \text{ or } p + 1 \equiv 0 \pmod{q}, \\ \infty & \text{otherwise.} \end{cases}$$

Proof. Let w be a word of length $D_l = D_l(p, q)$ with a pure period q and a local R-period p. Suppose that $\gcd(p, q) = 1$. Assume first that $p + 1 \equiv 0 \pmod{q}$. By the periodicity assumption, we then have

$$w_i \, R \, w_{i+p} = w_{i-1}$$

for all $i = 2, 3, \ldots, q$, and $w_1 \, R \, w_{1+p} = w_q$. Since q is a period of w, 1 is a local R-period of w. On the other hand, if we set $R = \langle (a, c), (b, c) \rangle$, the word

$$w = (c^{q-2} ab)^{(p+q-1)/q}$$

has a pure period q and a local R-period p. However, $\gcd(p,q) = 1$ is not a local R-period of w, since $(w_{q-1}, w_q) \notin R$. Note that in order to check that w has a local period p, it suffices to ensure that the distance from any occurrence of a to any occurrence of b is not p. By the length of w, this holds. Namely, the only position i such that $w_i \in \{a,b\}$ and $i+p \leq |w|$ is the first occurrence of the letter a. We have $a = w_{q-1}\, R\, w_{q-1+p} = w_{q-2} = c$. Moreover, let $kq = p+1$ for some positive k. Then the only position i such that $w_i \in \{a,b\}$ and $i-p > 0$ is the last occurrence of b. Hence, this position is kq and $b = w_{kq}\, R\, w_{kq-p} = w_1 = c$.

Assume next that $p - 1 \equiv 0 \pmod{q}$. Now $w_i\, R\, w_{i+p} = w_{i+1}$ for all $i = 1, 2, \ldots, q$. As above, this means that w has a local R-period 1. Our bound is strict, since setting again $R = \langle (a,c), (b,c) \rangle$, the word

$$w = (ac^{q-2}b)^{(p+q-1)/q}$$

has a pure period q and a local R-period p. However, $(w_q, w_{q+1}) \notin R$ and 1 is not a local R-period. Again the length of w ensures that a and b do not have to be related. As above, we have to check the first occurrence of a and the last occurrence of b. Assume that $p - 1 = kq$ for some positive k. We have $a = w_1\, R\, w_{1+p} = w_2 = c$ and $b = w_{(k+1)q}\, R\, w_{(k+1)q-p} = w_{q-1} = c$.

Finally, assume that q does not divide $p-1$ nor $p+1$. Then $i+p \not\equiv i+1 \pmod{q}$ and $i+p \not\equiv i-1 \pmod{q}$. Thus, if $R = \langle (a,c), (b,c) \rangle$, then the infinite word

$$w = (abc^{q-2})^{\omega}$$

has a pure period q and a local R-period p, but clearly 1 is not a local R-period of w. □

The next theorem shows that local R-periods are weak also when considering other types of interaction.

Theorem 5.14. *Let p and q be positive integers with $\gcd(p,q) = 1$. The bounds $D_e(p,q)$ of local-external interaction and $D_g(p,q)$ of local-global interaction do not exist, except for $q = 3$, in which case $D_e(p,q) = p+2$ and $D_g(p,q) = p+3$.*

Proof. As usual, denote $D_e(p,q) = D_e$ and $D_g(p,q) = D_g$. Consider first the case where $q = 3$. Assume that a word w has a pure period 3 and a local R-period p. Recall that $[n]_q$ is the least positive residue of an integer $n \pmod{q}$. If $|w| \geq p+2$, then

$$w_1\, R\, w_{[1+p]_3} \text{ and } w_2\, R\, w_{[2+p]_3}. \tag{5.28}$$

Since $\gcd(p,q) = 1$, the letter $w_{[1+p]_3}$ is equal to w_2 or w_3 and $w_{[2+p]_3}$ is equal to w_3 or w_1, respectively. This implies that in the first case $y = w_2$ is

an external word of w, whereas in the second case we may choose $y = w_1$. If $|w| \geq p + 3$, then in addition to (5.28), we have $w_3 \, R \, w_{[3+p]_3}$ where $w_{[3+p]_3}$ is equal to either w_1 or w_2. Hence, we must have $w_i \, R \, \mathrm{Alph}(w)$ for $i = 1, 2, 3$ and therefore 1 is a global R-period of w. On the other hand, $u = (abc)^{\lfloor (p+1)/3 \rfloor}$ with $R = \langle (a, w_{[1+p]_3}) \rangle$ and $v = (abc)^{\lfloor (p+2)/3 \rfloor}$ with $S = \langle (a, w_{[1+p]_3}), (b, w_{[2+p]_3}) \rangle$ show that the bounds D_e and D_g are strict for $q = 3$.

Otherwise, let $q \geq 4$. Consider a four letter alphabet $\{a, b, c, d\}$ and set $R = \langle (a, b), \, (b, c), (c, d), (d, a) \rangle$. Define an infinite word $w = (w_1 \cdots w_q)^\omega$ in the following way. Set

$$w_1 = a, \;\; w_{[1+p]_q} = b, \;\; w_{[1+2p]_q} = c \text{ and } w_{[1+ip]_q} = d$$

for $i = 3, 4, \ldots, q-1$. Now, by the definition of R, $w_i \, R \, w_{i+p}$ for all $i = 1, 2, \ldots, q$. Hence, p is a local R-period of w. However, 1 is neither an external nor a global R-period, since no letter is compatible with all the other letters. Hence, $D_e = D_g = \infty$. □

5.2.6 Summary of Bounds

In order to have a clearer picture of the different variants of Fine and Wilf's theorem represented in the previous sections, we summarize the bounds in Table 5.3.

By Theorem 5.1, a global period is a stronger attribute than the other periods, and therefore

$$B_g(p, q) \geq B_e(p, q) \text{ and } B_g(p, q) \geq B_l(p, q)$$

for every p and q. Observe also that B-bounds (B_g, B_e and B_l) are in many cases smaller than the other bounds. On the other hand, if we compare the bounds of global-external and global-local interaction we see, for example, that

$$\begin{array}{rcrcrcl} B_e(5,9) & = & 23 & > & 19 & = & B_l(5,9), \\ B_e(4,7) & = & 15 & = & 15 & = & B_l(4,7), \\ B_e(3,5) & = & 8 & < & 10 & = & B_l(3,5). \end{array}$$

This indicates the incomparability of the external relational period and the local relational period, which was already seen in Examples 5.1 and 5.3 with respect to minimal periods. However, in some sense the local period seems to be the weakest. In the case where p is an external period, we get interaction bounds, at least, if we assume extra conditions. In the case of a local period p, bounds usually do not exist. As an example, we give Table 5.4 which summarizes interaction bounds for $p = 6$ and $q = 7$.

interaction type	bound
global-global	$B_g = \begin{cases} \frac{p+1}{2}q & \text{if } (p < q \text{ and } p \text{ is odd}) \\ & \text{or } (p > q \text{ and } q \text{ is even}), \\ q + \frac{q-1}{2}p & \text{otherwise.} \end{cases}$
global-external	$B_e = \begin{cases} \min(h + [h]_q - 1, h + (q - [h]_q) + 1) & \text{if } q \text{ is odd,} \\ \max(h, h + [h]_q - (p+1)) & \text{if } q \text{ is even.} \end{cases}$
global-local	$B_l = \begin{cases} q + kp - 1 & \text{if } q \equiv 2 \pmod{p} \\ & \text{and } kp \equiv +1 \pmod{q}, \\ q + kp & \text{otherwise.} \end{cases}$
holding external-global	$C_g = pq$
holding external-external	$C_e = \begin{cases} (p-1)q + 1 & \text{if } p \text{ is even and } q > p, \\ (q-1)p + 1 & \text{otherwise.} \end{cases}$
holding external-local	$C_l = pq$
external-global	∞
inclusive external-external	$\overline{C} = \begin{cases} (q-2)p + (q-1) & \text{if } q \text{ is odd and } q \leq p+1 \\ (q-1)p + 1 & \text{otherwise.} \end{cases}$
external-external	$C = 1 + (q-1)p$
external-local	∞
local-global	$D_g = \begin{cases} p+3 & \text{if } q = 3 \\ \infty & \text{otherwise} \end{cases}$
local-external	$D_e = \begin{cases} p+2 & \text{if } q = 3 \\ \infty & \text{otherwise} \end{cases}$
local-local	$D_l = \begin{cases} p+q & \text{if } p - 1 \equiv 0 \pmod{q} \\ & \text{or } p + 1 \equiv 0 \pmod{q}, \\ \infty & \text{otherwise.} \end{cases}$

Table 5.3: Interaction bounds for p and q, where $\gcd(p, q) = 1$, $h = 1 + \lfloor q/2 \rfloor p$ and k is the smallest integer such that $kp \equiv \pm 1 \pmod{q}$.

t_1 \ t_2	global	external	local
global	25	22	13
holding external	42	36	42
inclusive external		36	
external	∞	37	∞
local	∞	13	∞

Table 5.4: Interaction bounds for $p = 6$ and $q = 7$.

5.3 Extremal Words

In this section we study words which demonstrate that the bound of global-global interaction is strict. The study of these words originates from the standard Fine and Wilf case, where for coprime periods p and q, the non-constant words of maximal length $p+q-2$ have very interesting properties. Such words are called *extremal Fine and Wilf words*. In 1994 de Luca and Mignosi [32] showed that the set of all factors of these words coincides with the set of factors of Sturmian words. Furthermore, the extremal words are palindromes and unique up to renaming of letters. The theorem of Fine and Wilf for more that two periods was investigated in several papers; see [26, 28, 50]. In 2003 Tijdeman and Zamboni [77] gave a fast algorithm to count an extremal word (and its length) for an arbitrary number of periods. Moreover, they showed that such word with periods $p_1, \ldots, p_r$ and without period $\gcd(p_1, \ldots, p_r)$ containing a maximal number of distinct letters is uniquely determined up to renaming of letters and it is a palindrome.

Here we consider relational extremal Fine and Wilf words in the case of global-global interaction. We prove that under some natural constraints, the structure of such words of maximal length is unique up to renaming of letters. These extremal words are over a ternary alphabet and the relation is necessarily similar to the compatibility relation of partial words. Furthermore, we consider their palindromic properties. Recall that $B_g(p, q)$ is the bound of global-global interaction for p and g.

Definition 5.5. For integers $p \geq 2$ and $q \geq 3$ satisfying $\gcd(p, q) = 1$, we define the set of *extremal relational Fine and Wilf words* $FW(p, q)$. A word w is in $FW(p, q)$ if $|w| = B_g(p, q) - 1$ and there exists a similarity relation R such that w has a global R-period p and a pure period q but $\gcd(p, q) = 1$ is not an R-period of w.

Denote by R_w the similarity relation with minimal number of pairs of letters such that $w \in FW(p, q)$ has an R_w-period p. Note that the relation

R_w is well defined: For each letter a occurring in w, let I_a be the set of positions i such that $w_i = a$. Consider the set of letters $\mathcal{B}_a$ in the positions $\{j \mid \exists\ i \in I_a : i \equiv j \pmod{p}\}$. The letter a must be R-compatible with the letters in $\mathcal{B}_a$. All other pairs involving a are inessential. In other words, $a\ R_w\ b \iff b \in \mathcal{B}_a$.

Note also that by the q-periodicity, only q different letters can occur in $FW(p,q)$. Moreover, both bounds $B_g(p,q) = \frac{p+1}{2}q$ and $B_g(p,q) = q + \frac{q-1}{2}p$ given in Table 5.1 are greater than $p+q-1$, which implies that the words must have at least three letters. Indeed, words over a binary alphabet $\{a,b\}$ with a relational R-period p and a pure period q and of length greater than $p+q-2$, are either unary by the theorem of Fine and Wilf or $a\,R\,b$ holds. In both cases, $\gcd(p,q) = 1$ is a relational period. Therefore, for $w \in FW(p,q)$, we have

$$3 \leq |\mathrm{Alph}(w)| \leq q,$$

where $\mathrm{Alph}(w)$ denotes the set of all letters occurring in w. In general $w \in FW(p,q)$ is not unique, not even up to renaming of letters.

Example 5.11. Consider the set $FW(3,7)$. For $p = 3$ and $q = 7$, we have the following bound

$$B(p,q) = \frac{p+1}{2}q = 14.$$

Hence, the length of the words in $FW(3,7)$ is 13. For a ternary alphabet $\{a,b,c\}$ and the relation $R = \langle (a,b),(b,c)\rangle$, we notice that $u = babbabcbabbab$ is in $FW(3,7)$. On the other hand, for the alphabet $\{a,b,c,d\}$, we have $v = abcacadabcaca \in FW(3,7)$ with the relation

$$R_v = \langle (a,b),(a,c),(a,d),(b,c),(c,d)\rangle.$$

Even if we restrict our considerations to words having the smallest possible number of different letters, we do not have uniqueness. For example, in addition to u, $w = babbbbcbabbbb \in FW(3,7)$.

Despite the previous examples, we show that all words in $FW(p,q)$ share in some sense a unique structure. We require the following definitions.

Definition 5.6. Let R be a similarity relation on $\mathcal{A}^*$. We say that two letters a and b are *relationally isomorphic*, or more precisely, *R-isomorphic* if, for each letter $x \in \mathcal{A}$, we have

$$a\,R\,x \iff b\,R\,x.$$

A letter a is *relationally universal*, or more precisely, *R-universal* if $a\,R\,x$ for all $x \in \mathcal{A}$.

In the sequel we consider words in $FW(p,q)$ which do not have any distinct relationally isomorphic letters and the number of occurrences of a relationally universal letter is minimal. This restriction is justified, since these words are a sort of template for other extremal relational Fine and Wilf words. Namely, all the words in $FW(p,q)$ can be obtained up to renaming of letters from the word w described in the next theorem by two operations, namely changing some symbols to universal symbols and replacing a letter with a new letter which is R_w-isomorphic to the original one. In this respect, $w \in FW(p,q)$ with no distinct R_w-isomorphic letters and with minimal number of occurrences of an R_w-universal letter can be called *minimal*. As above, we use the notation $[n]_q$ for the least positive residue of an integer $n \pmod q$. For simplicity, denote also $B = B_g(p,q)$. We have the following theorem.

Theorem 5.15. *Let w be a word in $FW(p,q)$ with no distinct R_w-isomorphic letters and with minimal number of occurrences of an R_w-universal letter. This word is unique up to renaming of letters. Furthermore, w is of the form uc^{-1}, where*

$$u = \begin{cases} \left(\left(b^{[B]_p-1}ab^{p-[B]_p}\right)^{\lfloor \frac{q}{p} \rfloor} b^{q-1-\lfloor \frac{q}{p} \rfloor p}c\right)^{\frac{p+1}{2}} & \text{if } B = \frac{p+1}{2}q \text{ and } p < q, \\ \left(b^{[B]_p-1}ab^{q-1-[B]_p}c\right)^{\frac{p+1}{2}} & \text{if } B = \frac{p+1}{2}q \text{ and } p > q, \\ (b^{[B]_q-1}cb^{q-[B]_q-1}a)^{\frac{B-[B]_q}{q}} b^{[B]_q-1}c & \text{otherwise,} \end{cases}$$

and the relation is $R_w = \langle (a,b),(b,c) \rangle$.

Proof. Consider a word v with a pure period q and an R-period p. Hence v is determined by its prefix of length q and the total length of the word. Let m and n be distinct integers in $\{1,2,\ldots,q\}$. As in the proof of Lemma 5.4, we consider the minimal solution (i,j) of the equation (5.4):

$$m + iq \equiv n + jq \pmod p.$$

If there exists a solution such that $\max(m+iq, n+jq) \le |v|$, then $v_m \, R \, v_n$ by the periods p and q. Recall that in the minimal solution, either $i = 0$ or $j = 0$.

By the definition of the bound $B_g = B_g(p,q)$, there exists a minimal solution satisfying $\max(m+iq, n+jq) \le B_g$ for each m and n. On the other hand, for some m' and n', there must be a minimal solution with $\max(m'+iq, n'+jq) = B_g$, since B_g is strict. Without loss of generality, we may assume that $i = 0$ and $n' + jq = B_g$. This implies that

$$n' = [B_g]_q \quad \text{and} \quad m' \equiv B_g \pmod p.$$

Consider now a word w in $FW(p,q)$ with no distinct R_w-isomorphic letters and with minimal number of occurrences of an R_w-universal letter.

Let us denote by U those integers in the set $\{1, 2, \ldots, q\} \setminus \{[B_g]_q\}$ that are not congruent to B_g modulo p. Note that U is not empty. The above considerations imply that a letter in a position belonging to U is related to all letters occurring in the word. Hence, denoting the R_w-universal letter by b, we have $w_i = b$ for all $i \in U$.

Let us now consider the position $[B_g]_q$. If $w_{[B_g]_q} = b$, then letters in the positions $m \equiv B_g \pmod{p}$ are R_w-compatible with all the letters in w, i.e., with each other and with the universal letter b. Thus, $\gcd(p, q) = 1$ is a global R_w-period. This is a contradiction. Hence, the letter in position $[B_g]_q$ is different from b, say $w_{[B_g]_q} = c$. Since $\gcd(p, q) = 1$ is not an R_w-period, there must exist a letter a in some of the positions $m \equiv B_g \pmod{p}$ such that $(a, c) \notin R_w$. If a position m is such that the minimal solution of (5.4) for all $n \in \{1, 2, \ldots, q\}$ satisfies $\max(m + iq, n + jq) \leq |w|$, then the letter w_n is related to all the letters in $\mathrm{Alph}(w)$, i.e., $w_m = b$. If this is not the case, then the smallest solution of (5.4) for m and $n = [B_g]_q$ must satisfy $\max(m + iq, [B_g]_q + jq) > |w|$. Since in w there is a minimal number of occurrences of the universal letter, this means that $w_m \neq b$. More precisely, $w_m \, R_w \, w_n$ for $n \in \{1, 2, \ldots, q\} \setminus \{[B_g]_q\}$ and $(w_m, w_{[B_g]_q}) \notin R_w$. Since w does not have any distinct R_w-isomorphic letters, we may set $w_m = a$ for all such positions m. This shows us that all the letters w_l, $l = 1, 2, \ldots, q$, are determined by the minimal solutions of (5.4), and the word w is a unique word over a three letter alphabet.

In order to find all occurrences of the letter a, we must determine which of the positions $1 \leq m \leq q$ satisfying $m \equiv B_g \pmod{p}$ do not have a solution (i, j) for

$$m + iq \equiv [B_g]_q + jq \pmod{p} \tag{5.29}$$

such that $\max(m + iq, [B_g]_q + jq) \leq B_g - 1$. Thus, we need to consider minimal solutions. As in the proof of Lemma 5.4, for a fixed m, there exists a unique solution $(0, j)$ of (5.29) such that $0 \leq j < pq$. Moreover, either this solution or the solution $(p - j, 0)$ is the minimal one.

Consider first those cases of Table 5.1 where $B_g = \frac{p+1}{2}q$ and assume that $m \equiv B_g \pmod{p}$. For a solution $(i, j) = (0, \frac{p-1}{2})$, we have $[B_g]_q + jq = q + \frac{p-1}{2}q = B_g$. For the other solution $(p - j, 0)$, we have

$$m + (p - j)q = m + pq - \frac{p-1}{2}q = \frac{p+1}{2}q + m = B_g + m.$$

Hence, for the minimal solution, we have $\max(m + iq, [B_g]_q + jq) > B_g - 1$. This proves that a letter in a position $1 \leq m \leq q$ satisfying $m \equiv B_g \pmod{p}$ is not universal, i.e., it must be the letter a. Note that if $B_g = \frac{p+1}{2}q$ and $p > q$, then q is even by Table 5.1 and $B_g \equiv \frac{q}{2} \pmod{p}$. Thus, $m \in \{1, 2, \ldots, q\}$, $m \equiv B_g \pmod{p}$ really exists.

Consider next those cases where $B_g = q + \frac{q-1}{2}p$. Now $m = q - kp$ for some $k = 0, 1, \ldots, \lfloor \frac{q}{p} \rfloor$ and $(i, j) = (0, \frac{B_g - [B_g]_q}{q})$ is a solution where $[B_g]_q + jq = B_g$. For the other solution $(p - j, 0)$, we have

$$\begin{aligned} m + (p-j)q &= m + pq - B_g + [B_g]_q = q - kp + pq - q - \frac{q-1}{2}p + [B_g]_q \\ &= B_g + [B_g]_q + (p-q) - kp. \end{aligned}$$

If $p > q$, then $k = 0$, $p - q > 0$ and $m + (p-j)q > B_g - 1$. If $p < q$, then p is even by Table 5.1. Hence, $[B_g]_q = q - \frac{p}{2}$. We get $m + (p-j)q = B_g + \frac{p}{2} - kp > B_g - 1$ if and only if $k = 0$. Thus, the only position $m \in \{1, 2, \ldots, q\} \setminus \{[B_g]_q\}$ where the minimal solution satisfies $\max(m + iq, [B_g]_q + jq) > B_g - 1$ is $m = q$. In other words, $w_m = a$ if and only if $m = q$.

The preceding calculations show that the word w must be of the form given in the statement of theorem and, furthermore, $R_w = \langle (a, b), (b, c) \rangle$. □

Note that the relation $R_w = \langle (a, b), (b, c) \rangle$ in Theorem 5.15 which was used in defining the minimal extremal words in $FW(p, q)$ corresponds to the compatibility relation of partial words.

As in the case of normal extremal Fine and Wilf words [32, 77], the minimal extremal relational Fine and Wilf words given in Theorem 5.15 have nice palindromic properties. A word $w = w_1 \cdots w_n$ is a *palindrome* if $w = \overline{w}$, where $\overline{w} = w_n w_{n-1} \cdots w_1$. A generalization of palindromic words are so called pseudo-palindromic words.

Definition 5.7. Let $\varphi \colon \mathcal{A} \to \mathcal{A}$ be a morphism satisfying $\varphi^2 = \mathrm{id}$. A word $w = w_1 \cdots w_n$ is a *φ-pseudo-palindrome* if $w = \varphi(\overline{w})$ for $\overline{w} = w_n w_{n-1} \cdots w_1$.

For more information on palindromes and pseudo-palindromes, see [4, 31]. As a final result of extremal Fine and Wilf words we prove the following palindromic properties.

Theorem 5.16. *Let $w \in \mathcal{A}^*$, where $\mathcal{A} = \{a, b, c\}$, belong to $FW(p, q)$ with no distinct R_w-isomorphic letters and with minimal number of occurrences of an R_w-universal letter. Let $R_w = \langle (a, b), (b, c) \rangle$. If $B_g(p, q) = \frac{p+1}{2}q$, then w is a palindrome. Otherwise, it is a φ-pseudo-palindrome, where $\varphi \colon \mathcal{A} \to \mathcal{A}$ is defined by $\varphi(a) = c$ and $\varphi(b) = b$.*

Proof. The word w is given by the formula of Theorem 5.15. Consider first $w \in FW(p, q)$ such that $B_g(p, q) = \frac{p+1}{2}q$. Suppose that $w_m = a$. By Theorem 5.15, $m = n + iq$ for some i and $1 \leq n < q$ satisfying $n \equiv B_g \pmod{p}$. Since $B_g \equiv 0 \pmod{q}$, $w_{B_g - n - iq} = w_{q-n}$ by the period q. Since $n \equiv B_g \pmod{p}$, we have $q - n \equiv q - B_g + pq = \frac{p+1}{2}q = B_g \pmod{p}$. This means that $w_{B_g - m} = w_{q-n} = a$.

Then consider occurrences of c in w. Suppose now that $w_m = c$. By Theorem 5.15, $m \equiv 0 \pmod{q}$. Since $B_g = \frac{p+1}{2}q$, also $B_g - m \equiv 0 \pmod{q}$.

This implies that $w_{B_g-m} = c$ and we have shown that $w_m = w_{B_g-m} = w_{|w|+1-m}$ if $w_m = a$ or $w_m = c$. Hence, this is true also for $w_m = b$ and the word w is a palindrome.

Next consider $w \in FW(p,q)$ such that $B_g(p,q) = q + \frac{q-1}{2}p$. By Theorem 5.15, we know that $w_m = a$ if and only if $m \equiv 0 \pmod{q}$ and $w_m = c$ if and only if $m \equiv B_g \pmod{q}$. Hence, for $w_m = a$, we have $B_g - m \equiv B_g \pmod{q}$ and therefore $w_{B_g-m} = c$. On the other hand, if $w_m = c$, then $B_g - m \equiv 0 \pmod{q}$ and $w_{B_g-m} = a$. Thus, $w_m = \varphi(w_{B_g-m}) = \varphi(w_{|w|+1-m})$, i.e., w is a φ-pseudo-palindrome. □

We finish this section by giving some examples of relational extremal Fine and Wilf words demonstrating also the palindromic properties discussed above.

Example 5.12. We showed already in Example 5.6 that the word $w = (abbbbac)^2abbbba$ with the relation $R = \langle (a,b),(b,c)\rangle$ is a word of maximal length such that w has a global R-period 5 and a pure period 7, but 1 is not a global R-period. Note that in $w_1 \cdots w_7$ the letter c occurs in the position $[B_g]_7 = 7$ and the letter a occurs exactly in positions 1 and 6, which are the positions congruent to $B_g = 21$ modulo 5. Hence, w is of the form uc^{-1} given in Theorem 5.15. Moreover, this word is clearly a palindrome.

The word w is minimal and therefore acts as a template for other words in $FW(5,7)$. For example, replace the letter w_2 by d and w_6 by e. From Figure 5.1 we clearly see that $w_2 = d$ must be R_w-isomorphic to b. In other words, it must be R-universal. Similarly, $w_6 = e$ must be R_w-isomorphic to a. Hence, we have $v = (adbbbec)^2adbbbe \in FW(5,7)$ with the relation $R_v = \langle \Omega_{\mathcal{A}} \setminus \{(a,c),(c,a),(e,c),(c,e)\}\rangle$. Note that this word is neither a palindrome nor a pseudo-palindrome.

As an example of a pseudo-palindrome, we consider a minimal word $w' \in FW(7,5)$. Now $B_g(7,5) = q + \frac{q-1}{2}p = 19$, $[B_g(7,5)]_5 = 4$ and $(B_g(7,5) - [B_g(7,5)]_5)/q = 3$. By the formula of Theorem 5.15, we have $w' = (bbbca)^3bbb$, which is a φ-pseudo palindrome for the morphism $\varphi\colon \{a,b,c\}^* \to \{a,b,c\}^*$ such that $\varphi(a) = c$ and $\varphi(b) = b$.

Chapter 6

Conclusions

In this thesis we introduced the notion of a similarity relation on words induced by a reflexive and symmetric relation on letters and, as a motivation for the research, we proposed several applications arising from computer science and molecular biology. We especially emphasized the connection between similarity relations and partial words. The thesis was divided into two parts dealing with independent topics, both of which are classical in the theory of combinatorics on words.

In the first part of the thesis, we revisited the theory of variable length codes. We generalized codes to relational (R, S)-codes and showed that the well-known Sardinas–Patterson algorithm can be extended to test whether a given set of words is an (R, S)-code or not. Furthermore, we analyzed coding properties of relational codes by inventing algorithms for finding maximal alteration relations and minimal fidelity relations. Moreover, an NP-complete problem related to relational codes was given. Many classical notions and results concerning unique factorization and free word monoids were generalized for similarity relations. As an example, we may here mention the unique factorization extensions of submonoids, stability, hulls, Schützenberger's criterion and Tilson's closure result. Using these notions we were able to prove our main result, a defect theorem formulated for (R, S)-unique factorization hulls. Moreover, a cumulative defect theorem of (R, S)-free hulls and a defect theorem of partial words were obtained as a corollary. We also modified Spehner graphs for similarity relations and showed how they can be used for implementations of algorithms for finding hulls and testing (R, S)-codes.

The second part of the thesis was devoted to interaction properties of periods with respect to similarity relations. We defined global, external and local relational periods and proved several relational variations of the famous theorem of Fine and Wilf. In the end, we also exposed some properties of relational extremal Fine and Wilf words.

Research on similarity relations offers a wide range of possible research

topics in the future. It is an interesting question to consider how different theorems of word combinatorics can be generalized when the accuracy of the initial data (words) is obscured by a similarity relation. The research on partial words has a similar flavour and, indeed, in some cases replacing the compatibility of partial words by a more general similarity relation does not induce any constituent difficulties. On the other hand, there are theorems like the theorem of Fine and Wilf and critical factorization theorem, where the number of holes in the partial word plays an essential role. These theorems can not be easily generalized for arbitrary similarity relations, on the contrary, other kind of formulations are needed.

As an example, we mention one intriguing question for future studies. This is the attempt to generalize the theory of Morse and Hedlund for relational periodicity. The problem is to define a concept of subword complexity in such a way (if it exists) that relationally ultimately periodic words are exactly those with bounded complexity. Here all three period types have distinct roles and open up various alternatives for the theorem and complexity function itself.

Bibliography

[1] B. Adamczewski and Y. Bugeaud. On the complexity of algebraic numbers. I. Expansions in integer bases. *Ann. of Math. (2)*, 165(2):547–565, 2007.

[2] B. Adamczewski, Y. Bugeaud, and F. Luca. Sur la complexité des nombres algébriques. *C. R. Math. Acad. Sci. Paris*, 339(1):11–14, 2004.

[3] S. I. Adian. *The Burnside problem and identities in groups*, volume 95 of *Ergebnisse der Mathematik und ihrer Grenzgebiete [Results in Mathematics and Related Areas]*. Springer-Verlag, Berlin, 1979. Translated from the Russian by John Lennox and James Wiegold.

[4] V. Anne, L. Q. Zamboni, and I. Zorca. Palindromes and pseudo-palindromes in episturmian and pseudo-episturmian infinite words. In S. Brlek and C. Reutenauer, editors, *Proceedings of Words 2005*, volume 36 of *Publications du LACIM*, pages 91–100, 2005.

[5] J. Baylis. *Error-Correcting Codes: A Mathematical Introduction*. Chapman and Hall Mathematics Series. Chapman & Hall, London, 1998.

[6] J. Berstel and L. Boasson. Partial words and a theorem of Fine and Wilf. *Theoret. Comput. Sci.*, 218(1):135–141, 1999. WORDS (Rouen, 1997).

[7] J. Berstel and J. Karhumäki. Combinatorics on words—a tutorial. *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS*, 79:178–228, 2003.

[8] J. Berstel and D. Perrin. *Theory of Codes*, volume 117 of *Pure and Applied Mathematics*. Academic Press Inc., Orlando, FL, 1985.

[9] J. Berstel, D. Perrin, J.-F. Perrot, and A. Restivo. Sur le théorème du défaut. *J. Algebra*, 60(1):169–180, 1979.

[10] F. Blanchet-Sadri. A periodicity result of partial words with one hole. *Comput. Math. Appl.*, 46(5-6):813–820, 2003.

[11] F. Blanchet-Sadri. Codes, orderings, and partial words. *Theoret. Comput. Sci.*, 329(1-3):177–202, 2004.

[12] F. Blanchet-Sadri. Periodicity on partial words. *Comput. Math. Appl.*, 47(1):71–82, 2004.

[13] F. Blanchet-Sadri. Primitive partial words. *Discrete Appl. Math.*, 148(3):195–213, 2005.

[14] F. Blanchet-Sadri. *Algorithmic Combinatorics on Partial Words*. Chapman & Hall/CRC Press, Boca Raton, FL, 2007.

[15] F. Blanchet-Sadri and A. R. Anavekar. Testing primitivity on partial words. *Discrete Appl. Math.*, 155(3):279–287, 2007.

[16] F. Blanchet-Sadri, D. D. Blair, and R. V. Lewis. Equations on partial words. In R. Královič and P. Urzyczyn, editors, *MFCS 2006, 31st International Symposium on Mathematical Foundations of Computer Science, Stará Lesná, Slovakia, August 28-September 1, 2006*, volume 4162 of *Lecture Notes in Comput. Sci.*, pages 167–178. Springer-Verlag, Berlin, Heidelberg, 2006.

[17] F. Blanchet-Sadri, N. C. Brownstein, and J. Palumbo. Two element unavoidable sets of partial words. In T. Harju, J. Karhumäki, and A. Lepistö, editors, *DLT 2007, 11th International Conference on Developments in Language Theory, Turku, Finland, July 3-6, 2007*, volume 4588 of *Lecture Notes in Comput. Sci.*, pages 96–107. Springer-Verlag, Berlin, Heidelberg, 2007.

[18] F. Blanchet-Sadri and A. Chriscoe. Local periods and binary partial words: an algorithm. *Theoret. Comput. Sci.*, 314(1-2):189–216, 2004.

[19] F. Blanchet-Sadri, K. Corcoran, and J. Nyberg. Fine and Wilf's periodicity result on partial words and consequences. In *LATA 2007 1st International Conference on Language and Automata Theory and Applications, Tarragona, Spain, March 29-April 4, 2007*.

[20] F. Blanchet-Sadri and S. Duncan. Partial words and the critical factorization theorem. *J. Combin. Theory Ser. A*, 109(2):221–245, 2005.

[21] F. Blanchet-Sadri, J. D. Gafni, and K. H. Wilson. Correlations on partial words. In W. Thomas and P. Weil, editors, *STACS 2007, 24th International Symposium on Theoretical Aspects of Computer Science, Aachen, Germany, February 22-24, 2007*, volume 4393 of *Lecture Notes in Comput. Sci.*, pages 97–108. Springer-Verlag, Berlin, Heidelberg, 2007.

[22] F. Blanchet-Sadri and R. A. Hegstrom. Partial words and a theorem of Fine and Wilf revisited. *Theoret. Comput. Sci.*, 270(1-2):401–419, 2002.

[23] F. Blanchet-Sadri and D. K. Luhmann. Conjugacy on partial words. *Theoret. Comput. Sci.*, 289(1):297–312, 2002.

[24] F. Blanchet-Sadri and M. Moorefield. Pcodes of partial words, October 2005. Manuscript, available at `http://www.uncg.edu/mat/pcode/40paper.pdf`.

[25] A. Carpi. On the repetition threshold for large alphabets. In R. Královič and P. Urzyczyn, editors, *MFCS 2006, 31st International Symposium on Mathematical Foundations of Computer Science, Stará Lesná, Slovakia, August 28-September 1, 2006*, volume 4162 of *Lecture Notes in Comput. Sci.*, pages 226–237. Springer-Verlag, Berlin, Heidelberg, 2006.

[26] M. G. Castelli, F. Mignosi, and A. Restivo. Fine and Wilf's theorem for three periods and a generalization of Sturmian words. *Theoret. Comput. Sci.*, 218(1):83–94, 1999. WORDS (Rouen, 1997).

[27] C. Choffrut and J. Karhumäki. Combinatorics of words. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 1*, pages 329–438. Springer, Berlin, 1997.

[28] S. Constantinescu and L. Ilie. Generalised Fine and Wilf's theorem for arbitrary number of periods. *Theoret. Comput. Sci.*, 339(1):49–60, 2005.

[29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, MA, second edition, 2001.

[30] M. Crochemore and W. Rytter. *Jewels of Stringology*. World Scientific Publishing Co. Inc., River Edge, NJ, 2003. Text Algorithms.

[31] A. de Luca and A. De Luca. Pseudopalindrome closure operators in free monoids. *Theoret. Comput. Sci.*, 362(1-3):282–300, 2006.

[32] A. de Luca and F. Mignosi. Some combinatorial properties of Sturmian words. *Theoret. Comput. Sci.*, 136(2):361–385, 1994.

[33] A. Ehrenfeucht, T. Harju, I. Petre, D. M. Prescott, and G. Rozenberg. *Computation in Living Cells–Gene Assembly in Ciliates*. Natural Computing Series. Springer-Verlag, Berlin, Heidelberg, 2004.

[34] A. Ehrenfeucht and G. Rozenberg. Elementary homomorphisms and a solution of the D0L sequence equivalence problem. *Theoret. Comput. Sci.*, 7(2):169–183, 1978.

[35] S. Ferenczi and C. Mauduit. Transcendence of numbers with a low complexity expansion. *J. Number Theory*, 67(2):146–161, 1997.

[36] N. J. Fine and H. S. Wilf. Uniqueness theorems for periodic functions. *Proc. Amer. Math. Soc.*, 16:109–114, 1965.

[37] M. J. Fischer and M. S. Paterson. String-matching and other products. In R. Karp, editor, *Complexity of Computation (Proc. SIAM-AMS Appl. Math. Sympos., New York, 1973)*, pages 113–125. SIAM–AMS Proc., Vol. VII. Amer. Math. Soc., Providence, R. I., 1974.

[38] Yu. V. Gamzova. Statistical patterns of the interaction of periods of partial words. *Diskretn. Anal. Issled. Oper. Ser. 1*, 11(4):20–35, 2004.

[39] M. R. Garey and D. S. Johnson. *Computers and intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., San Francisco, Calif., 1979.

[40] L. J. Guibas and A. M. Odlyzko. Periods in strings. *J. Combin. Theory Ser. A*, 30(1):19–42, 1981.

[41] V. Halava, T. Harju, and L. Ilie. Periods and binary words. *J. Combin. Theory Ser. A*, 89(2):298–303, 2000.

[42] V. Halava, T. Harju, and T. Kärki. Interaction properties of relational periods. Technical Report 798, TUCS Turku Centre for Computer Science, Turku, Finland, December 2006.

[43] V. Halava, T. Harju, and T. Kärki. The theorem of Fine and Wilf for relational periods. Technical Report 786, TUCS Turku Centre for Computer Science, Turku, Finland, October 2006.

[44] V. Halava, T. Harju, and T. Kärki. Relational codes of words. *Theoret. Comput. Sci.*, 389(1-2):237–249, 2007.

[45] V. Halava, T. Harju, and T. Kärki. Defect theorems with compatibility relations. *Semigroup Forum*, 76(1):1–24, 2008.

[46] V. Halava, T. Harju, T. Kärki, and L. Q. Zamboni. Relational Fine and Wilf words. In P. Arnoux, N. Bédaride, and J. Cassaigne, editors, *Proceedings of WORDS 2007*, pages 159–167. IML, 2007. Also available at `http://www.tucs.fi/research/series/` as a technical report 839 of TUCS Turku Centre for Computer Science.

[47] T. Harju and J. Karhumäki. On the defect theorem and simplifiability. *Semigroup Forum*, 33(2):199–217, 1986.

[48] T. Harju and J. Karhumäki. Many aspects of defect theorems. *Theoret. Comput. Sci.*, 324(1):35–54, 2004.

[49] T. Head and A. Weber. Deciding multiset decipherability. *IEEE Trans. Inform. Theory*, 41(1):291–297, 1995.

[50] J. Justin. On a paper by Castelli, Mignosi, Restivo. *Theor. Inform. Appl.*, 34(5):373–377, 2000.

[51] T. Kärki. Compatibility relation on codes and free monoids. In *Proceedings of XIth Mons Days of Theoretical Computer Science, Rennes, 30 August - 2 September, 2006*, pages 237–243. IFSIC/IRISA, 2006. To appear in Theor. Inform. Appl.

[52] T. Kärki. Transcendence of numbers with an expansion in a subclass of complexity $2N + 1$. *Theor. Inform. Appl.*, 40(3):459–471, 2006.

[53] T. Kärki. Spehner graphs for similarity relations. In *Proceedings of Workshop on Algorithms on Words, Turku, 28-30 March, 2007*. Department of Mathematics, University of Turku, Finland, 2007.

[54] G. Kucherov, L. Noé, and M. Roytberg. A unifying framework for seed sensitivity and its application to subset seeds. *J. Bioinformatics and Computational Biology*, 4(2):553–570, 2006.

[55] M. Linna. The decidability of the D0L prefix problem. *Internat. J. Comput. Math.*, 6(2):127–142, 1977/78.

[56] G. Lischke. Restorations of punctured languages and similarity of languages. *MLQ Math. Log. Q.*, 52(1):20–28, 2006.

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

[58] M. Lothaire. *Algebraic Combinatorics on Words*, volume 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2002.

[59] M. Lothaire. *Applied Combinatorics on Words*, volume 105 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2005.

[60] B. Ma, J. Tromp, and M. Li. Patternhunter: Faster and more sensitive homology search. *Bioinformatics*, 18(3):440–445, 2002.

[61] M. Morse and G. A. Hedlund. Symbolic dynamics II. Sturmian trajectories. *Amer. J. Math.*, 62:1–42, 1940.

[62] J. Moulin-Ollagnier. Proof of Dejean's conjecture for alphabets with $5, 6, 7, 8, 9, 10$ and 11 letters. *Theoret. Comput. Sci.*, 95(2):187–205, 1992.

[63] D. W. Mount. *Bioinformatics: Sequence and Genome Analysis.* Cold Spring Harbor Laboratory Press, New York, 2001.

[64] An. A. Muchnik. The definable criterion for definability in Presburger arithmetic and its applications. *Theoret. Comput. Sci.*, 290(3):1433–1444, 2003.

[65] S. Muthukrishnan and H. Ramesh. String matching under a general matching relation. *Inform. and Comput.*, 122(1):140–148, 1995.

[66] M. Rodeh. A fast test for unique decipherability based on suffix trees. *IEEE Trans. Inform. Theory*, 28(4):648–651, 1982.

[67] A. Salomaa. *Formal languages.* Academic Press (Harcourt Brace Jovanovich Publishers), New York, 1973. ACM Monograph Series.

[68] A. A. Sardinas and C. W. Patterson. A necessary and sufficient condition for the unique decomposition of coded messages. *IRE Internat. Conv. Rec.*, 8:104–108, 1953.

[69] M.-P. Schützenberger. Une théorie algébrique du codage. *C. R. Acad. Sci. Paris*, 242:862–864, 1956.

[70] A. L. Semenov. The Presburger nature of predicates that are regular in two number systems. *Sibirsk. Mat. Ž.*, 18(2):403–418, 479, 1977.

[71] A. M. Shur and Yu. V. Gamzova. Periods' interaction property for partial words. In T. Harju and J. Karhumäki, editors, *Proceedings of WORDS'03*, volume 27 of *TUCS General Publication*, pages 75–82. Turku Centre for Computer Science, Turku, 2003.

[72] A. M. Shur and Yu. V. Gamzova. Partial words and the period interaction property. *Izv. Ross. Akad. Nauk Ser. Mat.*, 68(2):191–214, 2004.

[73] J. C. Spehner. Présentations et présentations simplifiables d'un monoïde simplifiable. *Semigroup Forum*, 14(4):295–329, 1977.

[74] G. Sutton and I. Dew. Shotgun fragmment assembly. In I. Rigoutsos and G. Stephanopoulos, editors, *Systems Biology, Volume 1: Genomics*, pages 79–117. Oxford University Press, New York, 2006.

[75] A. Thue. Über unendliche Zeichenreihen. *Norske Vid. Skrifter I Mat.-Nat. Kl., Christiania*, 7:1–22, 1906.

[76] A. Thue. Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen. *Norske Vid. Skrifter I Mat.-Nat. Kl., Christiania*, 1:1–67, 1912.

[77] R. Tijdeman and L. Q. Zamboni. Fine and Wilf words for any periods. *Indag. Math. (N.S.)*, 14(1):135–147, 2003.

[78] B. Tilson. The intersection of free submonoids of a free monoid is free. *Semigroup Forum*, 4:345–350, 1972.

Index

Turku Centre for Computer Science
TUCS Dissertations

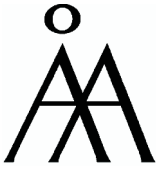
63. **Tommi Meskanen**, On the NTRU Cryptosystem
64. **Saeed Salehi**, Varieties of Tree Languages
65. **Jukka Arvo**, Efficient Algorithms for Hardware-Accelerated Shadow Computation
66. **Mika Hirvikorpi**, On the Tactical Level Production Planning in Flexible Manufacturing Systems
67. **Adrian Costea**, Computational Intelligence Methods for Quantitative Data Mining
68. **Cristina Seceleanu**, A Methodology for Constructing Correct Reactive Systems
69. **Luigia Petre**, Modeling with Action Systems
70. **Lu Yan**, Systematic Design of Ubiquitous Systems
71. **Mehran Gomari**, On the Generalization Ability of Bayesian Neural Networks
72. **Ville Harkke**, Knowledge Freedom for Medical Professionals – An Evaluation Study of a Mobile Information System for Physicians in Finland
73. **Marius Cosmin Codrea**, Pattern Analysis of Chlorophyll Fluorescence Signals
74. **Aiying Rong**, Cogeneration Planning Under the Deregulated Power Market and Emissions Trading Scheme
75. **Chihab BenMoussa**, Supporting the Sales Force through Mobile Information and Communication Technologies: Focusing on the Pharmaceutical Sales Force
76. **Jussi Salmi**, Improving Data Analysis in Proteomics
77. **Orieta Celiku**, Mechanized Reasoning for Dually-Nondeterministic and Probabilistic Programs
78. **Kaj-Mikael Björk**, Supply Chain Efficiency with Some Forest Industry Improvements
79. **Viorel Preoteasa**, Program Variables – The Core of Mechanical Reasoning about Imperative Programs
80. **Jonne Poikonen**, Absolute Value Extraction and Order Statistic Filtering for a Mixed-Mode Array Image Processor
81. **Luka Milovanov**, Agile Software Development in an Academic Environment
82. **Francisco Augusto Alcaraz Garcia**, Real Options, Default Risk and Soft Applications
83. **Kai K. Kimppa**, Problems with the Justification of Intellectual Property Rights in Relation to Software and Other Digitally Distributable Media
84. **Dragoş Truşcan**, Model Driven Development of Programmable Architectures
85. **Eugen Czeizler**, The Inverse Neighborhood Problem and Applications of Welch Sets in Automata Theory
86. **Sanna Ranto**, Identifying and Locating-Dominating Codes in Binary Hamming Spaces
87. **Tuomas Hakkarainen**, On the Computation of the Class Numbers of Real Abelian Fields
88. **Elena Czeizler**, Intricacies of Word Equations
89. **Marcus Alanen**, A Metamodeling Framework for Software Engineering
90. **Filip Ginter**, Towards Information Extraction in the Biomedical Domain: Methods and Resources
91. **Jarkko Paavola**, Signature Ensembles and Receiver Structures for Oversaturated Synchronous DS-CDMA Systems
92. **Arho Virkki**, The Human Respiratory System: Modelling, Analysis and Control
93. **Olli Luoma**, Efficient Methods for Storing and Querying XML Data with Relational Databases
94. **Dubravka Ilić**, Formal Reasoning about Dependability in Model-Driven Development
95. **Kim Solin**, Abstract Algebra of Program Refinement
96. **Tomi Westerlund**, Time Aware Modelling and Analysis of Systems-on-Chip
97. **Kalle Saari**, On the Frequency and Periodicity of Infinite Words
98. **Tomi Kärki**, Similarity Relations on Words: Relational Codes and Periods

Turku
Centre *for*
Computer
Science

Joukahaisenkatu 3-5 B, 20520 Turku, Finland | www.tucs.fi

University of Turku

- Department of Information Technology
- Department of Mathematics



Åbo Akademi University

- Department of Information Technologies

Turku School of Economics

- Institute of Information Systems Sciences

ISBN 978-952-12-2027-2
ISSN 1239-1883

Tomi Kärki

Similarity Relations on Words: Relational Codes and Periods

Tomi Kärki

Similarity Relations on Words: Relational Codes and Periods