On Special *k*-Spectra, *k*-Locality, and Collapsing Prefix Normal Words

Pamela Fleischmann

Dissertation
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
(Dr. rer. nat.)
der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel
eingereicht im Jahr 2020

Kiel Computer Science Series (KCSS) 2021/3 dated <++YYYY++>-<++MM++>-<++DD++>

URN:NBN <++URN-NBN++>
ISSN 2193-6781 (print version)
ISSN 2194-6639 (electronic version)

Electronic version, updates, errata available via https://www.informatik.uni-kiel.de/kcss

The author can be contacted via pamela@tompeteder.org

Published by the Department of Computer Science, Kiel University

Dependable Systems

Please cite as:

⊳ Pamela Fleischmann. *On Special k-Spectra, k-Locality, and Collapsing Prefix Normal Words* Number 2021/3 in Kiel Computer Science Series. Department of Computer Science, 2021. Dissertation, Faculty of Engineering, Kiel University.

```
@book{Fleischmann2021,
  author    = {Pamela Fleischmann},
  title     = {On Special $k$-Spectra, $k$-Locality, and Collapsing Prefix Normal Words},
  publisher = {Department of Computer Science, Kiel University},
  year      = {2021},
  number    = {2021/3},
  doi       = {10.21941/kcss/2021/3},
  series    = {Kiel Computer Science Series},
  note      = {Dissertation, Faculty of Engineering,
               Kiel University.}
}
```

About this Series

The Kiel Computer Science Series (KCSS) covers dissertations, habilitation theses, lecture notes, textbooks, surveys, collections, handbooks, etc. written at the Department of Computer Science at Kiel University. It was initiated in 2011 to support authors in the dissemination of their work in electronic and printed form, without restricting their rights to their work. The series provides a unified appearance and aims at high-quality typography. The KCSS is an open access series; all series titles are electronically available free of charge at the department's website. In addition, authors are encouraged to make printed copies available at a reasonable price, typically with a print-on-demand service.

Please visit `http://www.informatik.uni-kiel.de/kcss` for more information, for instructions how to publish in the KCSS, and for access to all existing publications.

1. Gutachter:	Prof. Dr. Dirk Nowotka Christian-Albrechts-Universität Kiel
2. Gutachter:	Prof. Dr. Tero Harju Turku University Turku
3. Gutachter:	Prof. Dr. Richard Weidmann Christian-Albrechts-Universität Kiel

Datum der mündlichen Prüfung: 17.12.2020

Zusammenfassung

Das Gebiet der Wortkombinatorik wurde 1906 von Axel Thue begründet und umfasst inzwischen viele verschiedene Unterbereiche. In dieser Arbeit werden die Bereiche der gestreuten Faktoren, die eine Repräsentation nicht vollständiger Informationen darstellen, sowie zwei Maße auf Wörtern, genauer die Lokalität und die Präfixnormalität, welche Anwendungen im Pattern Matching haben, untersucht.

Der erste Teil beschäftigt sich mit gestreuten Faktoren (auch bekannt als Teilwörter oder Teilsequenzen): Ein Wort u ist ein gestreuter Faktor eines Wortes w, wenn u aus w durch das Löschen von Buchstaben entsteht. Formal heißt das, dass u ein gestreuter Faktor von w ist, wenn möglicherweise leere Wörer $u_1, u_2, \dots, u_n$ und $v_0, v_1, \dots, v_n$ existieren, so dass $u = u_1 u_2 \cdots u_n$ und $w = v_0 u_1 v_1 u_2 v_2 \cdots u_n v_n$ gelten. In diesem Bereich untersuchen wir als erstes die Menge aller k-langen gestreuten Faktoren, das sogenannte k-Spektrum $\mathrm{ScatFact}_k(w)$, eines Wortes w. Wir beweisen zunächst Aussagen über $\mathrm{ScatFact}_k(w)$ für binäre, schwach-0-balancierte und schwach-c-balancierte Wörter. Dies sind binäre Wörter, in denen ein Buchstabe c-mal häufiger vorkommt als der andere. Insbesondere beschäftigen wir uns mit der Frage, welche Kardinalitäten die k-Spektren für eine gegebene Länge der gestreuen Faktoren haben können, wenn w ein beliebiges schwach-0-balanciertes Wort der Länge $2k$ bzw. ein schwach-c-balanciertes Wort der Länge $2k - c$ ist. Aufbauend auf diesen Erkenntnissen untersuchen wir die Wörter genauer, deren k-Spektrum alle Wörter der Länge k enthält. Diese Wörter nennen wir k-universelle Wörter. In diesem Zusammenhang präsentieren wir auch einen Algorithmus der in optimaler Zeit entscheidet, ob die k-Spektren zweier Wörter gleich sind bei gegebenem k. Neben diversen Ergebnissen zu k-universellen Wörtern, liegt ein Schwerpunkt auf der Berechnung des Universalitätsindizes von Wiederholungen von Wörtern. Hierfür wird der Begriff der Universalität zur zirkulären Universalität erweitert. Als letztes beschäftigen wir uns in diesem Teil der Arbeit mit dem Rekonstruktionsproblem für gestreute

Faktoren. Dieses Problem ist folgendermaßen definiert: finde die minimale Information, z.B. in Form von Multimengen von gestreuten Faktoren einer bestimmten Länge oder die Anzahl der Vorkommen bestimmter gestreuter Faktoren einer gegebenen Menge, die nötig ist, um ein Wort eindeutig zu bestimmen. Wir zeigen zunächst, dass ein Wort $w \in \{\mathtt{a}, \mathtt{b}\}^*$ eindeutig rekonstruierbar ist mit Hilfe der Anzahl der Vorkommen von höchstens $\min(|w|_\mathtt{a}, |w|_\mathtt{b}) + 1$ gestreuten Faktoren der Form $\mathtt{a}^i\mathtt{b}$ - hierbei ist $|w|_\mathtt{a}$ die Anzahl der Vorkommen von $\mathtt{a}$ in w. Dieses Ergebnis verallgemeinern wir auf Alphabete der Form $\{1, \ldots, q\}$, in dem wir zeigen, dass höchstens $\sum_{i=1}^{q-1} |w|_i \, (q - i + 1)$ gestreute Faktoren nötig sind, um w eindeutig zu rekonstruieren. Beide Ergebnisse sind besser als die bisher bekannten oberen Schranken. Außerdem behandeln wir auch die zeitliche Komplexität der vorgestellten Algorithmen.

Im zweiten Teil der Arbeit stehen Muster (pattern) im Vordergrund und das damit verbundene Pattern Matching Problem. Muster sind Wörter, die neben Buchstaben auch Variablen enthalten. Insbesondere beschäftigen wir uns mit dem Maß der Lokalität: ein Muster heißt k-lokal, wenn beim Markieren des Musters in jedweder Möglichkeit nie mehr als k markierte Blöcke entstehen. Als erstes zeigen wir, dass das Problem, die Lokalität eines Musters zu bestimmen, NP-vollständig ist. Danach stellen wir eine Reihe von Ergebnissen vor, die sich mit der Lokalität von Wiederholungen und Palindromen beschäftigen. Dieses Kapitel beenden wir mit dem für uns interessanten Ergebnis, dass das Matching Problem NP-schwer *werden kann*, wenn Muster gematcht werden, die Wiederholungen von Mustern sind, deren Matching-Problem effizient möglich ist. Dies zeigen wir anhand von regulären Mustern.

Im letzten Teil der Arbei untersuchen wir präfixnormale Wörter. Dies sind binäre Wörter, bei denen jedes Präfix mindestens soviele Einsen enthält wie der Faktor derselben Länge mit den meisten Einsen. Das Problem, den Index der präfixnormalen Äquivalenzrelation zu bestimmen, welches als erstes von Fici und Lipták 2011 vorgestellt wurde, ist noch offen. In dieser Arbeit betrachten wir zwei Aspekte dieses Problem. Wir untersuchen präfixnormale Palindrome und sogenannte kollabierende Wörter, die eine Erweiterung der verlängerungskritischen Wörter sind. Wir beweisen für beide Teile Charakterisierungen und zeigen den Zusammenhang zwischen diesen Klassen von Wörtern. Darauf aufbauend zeigen wir, dass offene Probleme im Zusammenhang mit Präfixnormalität auf Teilprobleme reduziert werden können.

Abstract

The domain of *Combinatorics on Words*, first introduced by Axel Thue in 1906, covers by now many subdomains. In this work we are investigating scattered factors as a representation of non-complete information and two measurements for words, namely the locality of a word and prefix normality, which have applications in pattern matching.

In the first part of the thesis we investigate scattered factors: A word u is a scattered factor of w if u can be obtained from w by deleting some of its letters. That is, there exist the (potentially empty) words $u_1, u_2, \ldots, u_n$, and $v_0, v_1, \ldots, v_n$ such that $u = u_1 u_2 \cdots u_n$ and $w = v_0 u_1 v_1 u_2 v_2 \cdots u_n v_n$. First, we consider the set of length-k scattered factors of a given word w, called the k-spectrum of w and denoted by $\mathrm{ScatFact}_k(w)$. We prove a series of properties of the sets $\mathrm{ScatFact}_k(w)$ for binary weakly-0-balanced and, respectively, weakly-c-balanced words w, i.e., words over a two-letter alphabet where the number of occurrences of each letter is the same, or, respectively, one letter has c occurrences more than the other. In particular, we consider the question which cardinalities $n = |\mathrm{ScatFact}_k(w)|$ are obtainable, for a positive integer k, when w is either a weakly-0-balanced binary word of length $2k$, or a weakly-c-balanced binary word of length $2k - c$. Second, we investigate k-spectra that contain all possible words of length k, i.e., k-spectra of so called k-universal words. We present an algorithm deciding whether the k-spectra for given k of two words are equal or not, running in optimal time. Moreover, we present several results regarding k-universal words and extend this notion to circular universality that helps in investigating how the universality of repetitions of a given word can be determined. We conclude the part about scattered factors with results on the reconstruction problem of words from scattered factors that asks for the minimal information, like multisets of scattered factors of a given length or the number of occurrences of scattered factors from a given set, necessary to uniquely determine a word. We show that a word $w \in \{\mathtt{a}, \mathtt{b}\}^*$ can be reconstructed from the number of occurrences of at

most $\min(|w|_{\mathtt{a}}, |w|_{\mathtt{b}}) + 1$ scattered factors of the form $\mathtt{a}^i\mathtt{b}$, where $|w|_{\mathtt{a}}$ is the number of occurrences of the letter $\mathtt{a}$ in w. Moreover, we generalise the result to alphabets of the form $\{1, \ldots, q\}$ by showing that at most $\sum_{i=1}^{q-1} |w|_i \,(q - i + 1)$ scattered factors suffices to reconstruct w. Both results improve on the upper bounds known so far. Complexity time bounds on reconstruction algorithms are also considered here.

In the second part we consider patterns, i.e., words consisting of not only letters but also variables, and in particular their locality. A pattern is called k-local if on marking the pattern in a given order never more than k marked blocks occur. We start with the proof that determining the minimal k for a given pattern such that the pattern is k-local is NP-complete. Afterwards we present results on the behaviour of the locality of repetitions and palindromes. We end this part with the proof that the matching problem becomes also NP-hard if we do not consider a regular pattern - for which the matching problem is efficiently solvable - but repetitions of regular patterns.

In the last part we investigate prefix normal words which are binary words in which each prefix has at least the same number of 1s as any factor of the same length. First introduced in 2011 by Fici and Lipták, the problem of determining the index (amount of equivalence classes for a given word length) of the prefix normal equivalence relation is still open. In this paper, we investigate two aspects of the problem, namely prefix normal palindromes and so-called collapsing words (extending the notion of critical words). We prove characterizations for both the palindromes and the collapsing words and show their connection. Based on this, we show that still open problems regarding prefix normal words can be split into certain subproblems.

Acknowledgements

With a short tail wagging a dog can express more emotion than some people with hours of talk. (Louis Armstrong)

This work would not have been possible without the help, encouragement, support and advise of many people whom I would like to thank at this place.

First of all, I thank my supervisor, Dirk Nowotka, for his faith in me, such that I was able to choose my research topics and to discover my passion for scattered factors. Without his support and guidance, this would not have been possible. I am also grateful that I got the opportunity to gain insights into industry projects and their coordination and organisation.

Second, my gratitude is by Florin Manea. Thanks a lot of for all the discussions, for your advise, for insights to the world of academia, and for not always finishing everything in the very last minute.

Thanks to my current and former colleagues from the Dependable Systems Group in Kiel, who worked with me and questioned me, laughed and cursed with me, or simply drank a good glass of wine with me: Mitja Kulczynski, Thorsten Ehlers, Philipp Sieweck, Karoliina Lehtinen, Max Jonas Friese, Joel Day, Danny Bøgsted Poulsen.

Furthermore, I would like the members of my examining committee, Tero Harju, Richard Weidmann, Klaus Jansen, and Thomas Wilke for their efforts.

Science is collaboration; I would like to thank my co-authors that were not mentioned so far, Marie Lejeune, Michel Rigo, Stefan Siemer, Maria Kosche, Tore Koß, Pascal Ochem, Malte Skambath, and Max Bannach. Moreover, a special thank to my Words Nerds; working on words can be fun in a relaxed environment when finding the secrets of the nasty words is the only goal. Especially, I would like to mention Laura Barker, Katharina Harwardt, Judith Wiedenbeck, Lukas Haschke, Cedric Tsatia Tsida, and Yannik Eikmeier, who took the extra time to convert their work into a paper.

Since working in a university is not only science, but also bureaucracy and organisation, I would like to thank Ina Pfannenschmidt, Frank Huch, Anne Bock, Brigitte Scheidemann, and Michael Hanus for their support and help in working with the international students.

Finally, I express my deep gratitude to Andreas, who listened to me, bore my mood when the technology did not work as I would have liked it to, encouraged me to keep on when the normal chaos went south, and prepared dinner when the day was again longer then intended. Moreover, I have to thank Neska for showing me what really is important in life and her "do not complain just do it"-way. With the beginning of my PhD studies, Alfons came into my life, and even though he is the most stupid dog I ever had, he taught me that it is sometimes okay to be afraid of the results of one's own actions.

tl;dr
Thanks to all of you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The domain of *Combinatorics on Words* dates back to Axel Thue in 1906 [109]. A word is a sequence of elements, called letters, from a given set, called alphabet and denoted by Σ, X, or $\mathcal{A}$, depending whether it is finite, infinite, or arbitrary in size. Since then the investigation of words led its an own research field with many well established subdomains. In this work we investigate three different parts of the domain, namely scattered factors, k-local words, and prefix normal words.

Given a word w, a scattered factor (also called scattered subword, or simply subword in the literature) is a word obtained by removing one or more parts (called factors) from w. More formally, u is a scattered factor of w if there exist $u_1, \ldots, u_n \in \Sigma^*$, $v_0, \ldots, v_n \in \Sigma^*$ such that $u = u_1 u_2 \cdots u_n$ and $w = v_0 u_1 v_1 u_1 \cdots u_n v_n$. Consequently, a scattered factor of w can be thought of as a representation of w in which some parts are missing, e.g., *do take tart* is a scattered factor of *do not take the sausage from the table* (ignoring the spaces). As such, there is considerable interest in the relationship of a word and its scattered factors from both a theoretical and practical point of view. For an introduction to the study of scattered factors, see Chapter 6 of [81]. On the one hand, it is easy to imagine how, in any situation where discrete, linear data is read from an imperfect input – such as when sequencing DNA or during the transmission of a digital signal – scattered factors form a natural model, as multiple parts of the input may be missed, but the rest will remain unaffected and in-sequence. For instance, various applications and connections of this model in verification are discussed in [111, 55] within a language theoretic framework, while applications of the model in DNA sequencing are discussed in [34] in an algorithmic framework. On the other hand, from a more algebraic perspective, there have been efforts to bridge the gap between the non-commutative field of combinatorics on words with traditional commutative mathematics via Parikh matrices (cf., e.g., [90, 104, 106]) which are algebraic structures in which the number of specific scattered factors occurring in a word are stored. In an algorithmic framework, scattered factors play an important role in many classical problems, e.g., the longest common subsequence and the shortest common supersequence problems [83, 9], and the string-to-string correction problem [110]. More recently, scattered factors appear in applicative works related to bioinformatics [34]. This versatility of scattered factors is also highlighted by the many contexts in which this concept appears. For instance, in [111, 55, 74], various theories of logic were developed around the notion of scattered factors which are analysed mostly with automata theory tools and discussed in connection to applications in formal verification.

The set (or also in some cases, multi-set) of scattered factors of a word w, denoted $\mathrm{ScatFact}(w)$ is typically exponentially large in the length of w, and contains a lot of redundant information in the sense that, for $k' < k \leqslant |w|$, a word of length k' is a scattered factor of w if and only if it is a scattered factor of a scattered factor of w of length k. This has led to the idea of k-spectra: the set of all length-k scattered factors of a word. For example, the 3-spectrum of the word $\mathtt{ababbb}$ is the set $\{\mathtt{aab}, \mathtt{aba}, \mathtt{abb}, \mathtt{bab}, \mathtt{bbb}\}$. Note that unlike some literature, we do not consider the k-spectra to be the multi-set of scattered factors in the present work, but rather ignore the multiplicities when talking about k-spectra. This distinction is non-trivial as there are significant variations on the properties based on these different definitions (cf., e.g., [85]). Also, the notion of k-spectra is closely related to the classical notion of factor complexity of words, which counts, for each positive integer k, the number of distinct factors of length k of a word. Here, the cardinality of the k-spectrum of a word gives the number of the word's distinct scattered factors of length k. The study of a word's scattered factors of a fixed length has its roots in [108], where the relation $\sim_k$ (called Simon congruence) defines the congruence of words that have the same full k-spectra.

One of the most fundamental questions about k-spectra of words, and indeed sets of scattered factors in general, is that of recognition: given a set S of words (of length k), is S the k-spectrum of some word? In general, it remains a long standing goal of the theory of scattered factors to give a *nice* descriptive characterisation of scattered factor sets (and k-spectra), and to better understand their structure, cf. [81, 108].

In the current work in Section 3.1, we consider k-spectra in the restricted setting of a binary alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. For such an alphabet, we can always identify the natural number $c \in \mathbb{N}_0$ which describes how weakly balanced a word is: c is the difference between the amount of letters $\mathtt{a}$ and $\mathtt{b}$. Thus, it seems natural to categorise all words over Σ according to this difference: a binary word where one letter has exactly c more occurrences than the other one is called weakly-c-balanced. In Section 3.1 the cardinalities of k-spectra of weakly-c-balanced words of length $2k - c$ are investigated. Our first results concern the minimal and maximal cardinality the k-spectrum $\mathrm{ScatFact}_k$ might have. We show that the cardinality ranges for weakly-0-balanced words between $k + 1$ and 2^k, and determine exactly for which words of length $2k$ these values are reached. In the case of weakly-c-balanced words, we are able to replicate the result regarding the minimal cardinality of $\mathrm{ScatFact}_k$, but the case of maximal cardinality is more complicated. To this end, it seems that the words containing many alternations between the two letters of the alphabet have larger sets $\mathrm{ScatFact}_k$. Therefore, we first investigate the scattered factors of the words which are prefixes of $(\mathtt{ab})^\omega$ and give a precise description of all scattered factors of any length of such words. In particular, we do not compute only the cardinality of $\mathrm{ScatFact}_k(w)$, for all such words w, but also describe a way to obtain directly the respective scattered factors without repetitions. We use this to describe exactly the sets $\mathrm{ScatFact}_i$ for the word $(\mathtt{ab})^{k-c}\mathtt{a}^c$, which seems a good candidate for a weakly-c-balanced word with many distinct scattered factors.

In Section 3.2 we investigate in more detail the maximum cardinality of a k-spectrum for a given $k \in \mathbb{N}$, i.e., we are interested in a special congruence class w.r.t. $\sim_k$: the class of words which have the largest possible k-spectrum. A word w is called *k-universal* if its k-spectrum contains all words of length k over a given alphabet. That is, k-universal words are those words that are as rich as possible in terms of scattered factors of length k (and, consequently, also scattered factors of length at most k): the restriction of their downward closure to words of length k contains all possible words of the respective length, i.e., is a *universal* language. Thus, $w = \mathtt{aba}$ is not 2-universal since $\mathtt{bb}$ is not a scattered factor of w, while $w' = \mathtt{abab}$ is 2-universal. Calling the words *universal* whose k-spectra contain all possible words of length k is rooted in formal language theory. The classical universality problem (cf., e.g., [58]) is whether a given language L (over an alphabet Σ) is equal to Σ^*, where L can be given, e.g., as the language accepted by an automaton. A variant of this problem, called length universality, asks, for a natural number ℓ and a language L (over Σ), whether L contains all strings of length ℓ over Σ. See [53] for a series of results on this problem and a discussion on its motivation, and [96, 73, 53] and the references therein for more results on the universality problem for various types of automata. The universality problem was also considered for words [88, 10] and, more recently, for partial words [18, 54] w.r.t. their factors. In this context, the question is to find, for a given ℓ, words w over an alphabet Σ, such that each word of length ℓ over Σ occurs exactly once as a contiguous factor of w. De Bruijn sequences [10] fulfil this property, and have been shown to have many applications in various areas of computer science and combinatorics, see [18, 54] and the references therein. As such, our study of scattered factor-universality is related to, and motivated by, this well developed and classical line of research.

While $\sim_k$ is a well studied congruence relation from language theoretic, combinatorial, or algorithmic points of view (see [108, 81, 45] and the references therein), the study of universality w.r.t. scattered factors seems to have been mainly carried out from a language theoretic point of view. In [63] as well as in [65, 64] the authors approach, in the context of studying the height of piecewise testable languages, the notion of ℓ-rich words, which coincides with the ℓ-universal words we define here; we will discuss the relation between these notions, as well as our preference to talk about universality rather than richness, later in this work. A combinatorial study of scattered factor universality was started in [26], where a simple characterisation of k-universal binary words was given. In the combinatorics on words literature, more attention was given to the so called binomial complexity of words, i.e., a measure of the multiset of scattered factors that occur in a word, where each occurrence of such a factor is considered as an element of the respective multiset (see, e.g., [102, 49, 78, 77]). As such, it seemed interesting to us to continue the work on scattered factor universality: try to understand better (in general, not only in the case of binary alphabets) their combinatorial properties, but, mainly, try to develop an algorithmic toolbox around the concept of (k-)universal words. One of the main tools we are going to use is the arch factorisation introduced by Hebrard [57] which is recalled in the preliminaries. There we also explain in detail the connection to richness introduced in [63].

We start Section 3.2 with one of our main results: testing whether two words have the same full k-spectrum (set of all scattered factors up to length k), for given $k \in \mathbb{N}$, can be done in linear time. Our result works under the common assumption that the input alphabet is an integer alphabet (or that it can be sorted in linear time) and improves the results of [45]. Afterwards we prove that the arch factorisation can be computed in time linear w.r.t. the word-length and, thus, we can also determine whether a given word is k-universal. Afterwards, we provide several combinatorial results on k-universal words (over arbitrary alphabets); while some of them follow in a rather straightforward way from the seminal work of Simon [108], other require a more involved analysis. One such result is a characterisation of k-universal words. Moreover, we investigate the similarities and differences of the universality if a word w is repeated or w^R and $\pi(w)$ resp. are appended to w, for a morphic permutation π of the alphabet. As consequences, we get a linear run-time algorithm for computing a minimal length scattered factor of ww that is not a scattered factor of w. This approach works for arbitrary alphabets, while, e.g., the approach of [57] only works for binary alphabets. We finish this section by analysing the new notion of k-circular universality, connected to the universality of word repetitions. We finish this section with considering the problem of modifying the universality of a word by repeated concatenations or deletions. Motivated by the fact that, in general, starting from an input word w, we could reach larger sets of scattered factors of fixed length by iterative concatenations of w, we show that, for a word w over Σ and a positive integer k, we can compute efficiently the minimal ℓ such that w^ℓ is k-universal. This result is extensible to sets of words. Finally, the shortest prefix or suffix we need to delete to lower the universality index (the maximal k such that a word is k-universal) of a word to a given number can be computed in linear time. Interestingly, in all of the algorithms where we are concerned with reaching k-universality we never effectively construct a k-universal word (which would take exponential time, when k is given as input via its binary encoding, and would have been needed in order to solve these problems using, e.g., [45, 34]). Our algorithms run in polynomial time w.r.t. $|w|$, the length of the input word, and $\log_2(k)$, the size of the representation of the number k.

The last section of Chapter 3 is dedicated to the aforementioned reconstruction problem. The general scheme for a so-called *reconstruction problem* is the following one: given a sufficient amount of information about substructures of a hidden discrete structure, can one uniquely determine this structure? In particular, what are the fragments about the structure needed to recover it all. For instance, a square matrix of size at least 5 can be reconstructed from its principal minors given in any order [87].

In graph theory, given some subgraphs of a graph (these subgraphs may contain some common vertices and edges), can one uniquely rebuild the original graph? Given a finite undirected graph $G = (V, E)$ with n vertices, consider the multiset made of the n induced subgraphs of G obtained by deleting exactly one vertex from G. In particular, one knows how many isomorphic subgraphs of a given class appear. Two graphs leading to the same multiset (generally called a *deck*) are said to be *hypomorphic*. A conjecture due to Kelly and Ulam states that two hypomorphic graphs with at least three vertices are isomorphic [70, 92]. A similar conjecture in terms of edge-deleted subgraphs has been proposed by Harary [56]. These conjectures are known to hold true for several families of graphs. A finite word can be seen as an edge- or vertex-labeled linear tree. So variants of the graph reconstruction problem can be considered and are of independent interest. Participants of the Oberwolfach meeting on Combinatorics on Words in 2010 [7] gave a list of 18 important open problems in the field. Amongst them, the twelfth problem is stated as *reconstruction from subwords of given length*.

For natural numbers k, n, words of length n over a given alphabet are said to be *k-reconstructible* whenever the multiset of scattered factors of length k (or *k-deck*) uniquely determines any word of length n. Notice that the definition requires multisets to store the information how often a scattered factor occurs in the words. For instance, the scattered factor ba occurs three times in baba which provides more information for the reconstruction than the mere fact that ba is a scattered factor. The challenge is to determine the function $f(n) = k$ where k is the least integer for which words of length n are k-reconstructible. This problem has been studied by several authors and one of the first traces goes back to 1973 [62]. Results in that direction have been obtained by Schützenberger (with the so-called *Schützenberger's Guessing game*) and Simon [108]. They show that words of length n sharing the same multiset of scattered factors of length up to $\lfloor n/2 \rfloor + 1$ are the same. Consequently, words of length n are $(\lfloor n/2 \rfloor + 1)$-reconstructible. In [72], this upper bound has been improved: Krasikov and Roditty have shown that words of length n are k-reconstructible for $k \geqslant \lfloor \frac{16\sqrt{n}}{7} \rfloor + 5$. On the other hand Dudik and Schulmann [32] provide a lower bound: if words of length n are k-reconstructible, then $k \geqslant 3^{(\sqrt{2/3}-o(1))\log_3^{1/2} n}$. Bounds were also considered in [86]. Algorithmic complexity of the reconstruction problem is discussed, for instance, in [30]. Note that the different types of reconstruction problems have application in philogenetic networks, see, e.g., [59], or in the context of molecular genetics [35] and coding theory [79].

Another motivation, close to combinatorics on words, stems from the study of k-binomial equivalence of finite words and k-binomial complexity of infinite words (see [102] for more details). Given two words of the same length, they are k-binomially equivalent if they have the same multiset of scattered factors of length k, also known as *k-spectrum* ([6], [85], [103]). Given two words x and y of the same length, one can address the following problem: decide whether or not x and y are k-binomially equivalent? A polynomial time decision algorithm based on automata and a probabilistic algorithm have been addressed in [49]. A variation of our work would be to find, given k and n, a minimal set of scattered factors for which the knowledge of the number of occurrences in x and y permits to decide k-binomial equivalence.

Over an alphabet of size q, there are q^k pairwise distinct length-k factors. If we relax the requirement of only considering scattered factors of the same length, another interesting question is to look for a minimal (in terms of cardinality) multiset of scattered factors to reconstruct a word entirely. Let the *binomial coefficient* $\binom{u}{x}$ be the number of occurrences of x as a scattered factor of u. The general problem addressed in Section 3.3 is therefore the following one: For a given alphabet Σ and a natural number n, find a minimal number of k words $u_1, \ldots, u_k$ (not necessarily of the same length) such that no two words of length n over Σ have the same length-k vector of coefficients $\left[\binom{w}{u_1}, \ldots, \binom{w}{u_k}\right]$ and, thus, uniquely determine w by knowing the binomial coefficients of these k given words. In this new context, we naturally look for a value of k less than the upper bound for k-reconstructibility.

In Section 3.3, we recall first the use of Lyndon words in the context of reconstructibility. A word w over a totally ordered alphabet is called *Lyndon word* if it is the lexicographically smallest amongst all its rotations, i.e., $w = xy$ is smaller than yx for all non trivial factorisations $w = xy$. Every binomial coefficient $\binom{w}{x}$ for arbitrary words w and x over the same alphabet can be deduced from the values of the coefficients $\binom{w}{u}$ for Lyndon words u that are lexicographically less than or equal to x. Thus, we are considering an alphabet equipped with a total order on the letters. Words of the form $\mathtt{a}^n\mathtt{b}$ with letters $\mathtt{a} < \mathtt{b}$ and a natural number n are a special form of Lyndon words, the so-called *right-bounded-block* words. We consider the reconstruction problem from the information given by the occurrences of right-bounded-block words as scattered factors of a word of length n. In Section 3.3.1 we show how to reconstruct a word uniquely from $m+1$ binomial coefficients of right-bounded-block words where m is the minimum number of occurrences of $\mathtt{a}$ and $\mathtt{b}$ in the word. We also prove that this is less than the upper bound given in [72]. In Section 3.3.2 we reduce the problem for arbitrary finite alphabets $\{1, \ldots, q\}$ to the binary case. Here we show that at most $\sum_{i=1}^{q-1} |w|_i \, (q-i+1) \leqslant q|w|$ binomial coefficients suffice to uniquely reconstruct w with $|w|_i$ being the number of letter i in w. Again, we compare this bound to the best known one for the classical reconstruction problem (from words of a given length). In the last subsection we also propose several results of algorithmic nature regarding the efficient reconstruction of words from given scattered factors.

A special case of scattered factors are the *projections*. While a scattered factor is obtainable by deleting arbitrary parts of the word, for projections all letters but the ones of a given set are deleted. Taking $w = \mathtt{banana}$ as example and the set $\Delta = \{\mathtt{a}, \mathtt{b}\}$, the projection of w on Δ is $w_\Delta = \mathtt{baaa}$. This kind of scattered factors are investigated in the context of pattern matching in the domain of k-locality. The locality number is rather new and we shall discuss it in more detail. A word is k-local if there exists an order of its symbols such that, if we *mark* the symbols in the respective order (which is called a *marking sequence*), at each stage there are at most k contiguous blocks of marked symbols in the word. This k is called the *marking number* of that marking sequence. The *locality number* of a word is the smallest k for which that word is k-local, or, in other words, the minimum marking number over all marking sequences. For example, the marking sequence $\sigma = (\mathtt{x}, \mathtt{y}, \mathtt{z})$ marks $\alpha = \mathtt{xyxyzxz}$ as follows (marked blocks are illustrated by overlines): $\mathtt{xyxyzxz}$, $\overline{\mathtt{x}}\mathtt{y}\overline{\mathtt{x}}\mathtt{yz}\overline{\mathtt{x}}\mathtt{z}$, $\overline{\mathtt{xyxy}}\mathtt{z}\overline{\mathtt{x}}\mathtt{z}$, $\overline{\mathtt{xyxyzxz}}$; thus, the marking number of σ is 3. In fact, all marking sequences for α have a marking number of 3, except $(\mathtt{y}, \mathtt{x}, \mathtt{z})$, for which it is 2: $\mathtt{x}\overline{\mathtt{y}}\mathtt{x}\overline{\mathtt{y}}\mathtt{zxz}$, $\overline{\mathtt{xyxy}}\mathtt{z}\overline{\mathtt{x}}\mathtt{z}$, $\overline{\mathtt{xyxyzxz}}$. Thus, the locality number of α, denoted by $\mathrm{loc}(\alpha)$, is 2.

The locality number has applications in pattern matching with variables [27]. A *pattern* is a word that consists of *terminal symbols* (e. g., $\mathtt{a}, \mathtt{b}, \mathtt{c}$), treated as constants, and *variables* (e. g., $x_1, x_2, x_3, \ldots$). A pattern is mapped to a word by substituting the variables by strings of terminals. For example, $x_1 x_1 \mathtt{bab} x_2 x_2$ can be mapped to $\mathtt{acacbabcc}$ by the substitution $(x_1 \mapsto \mathtt{ac}, x_2 \mapsto \mathtt{c})$. Deciding whether a given pattern matches (i. e., can be mapped to) a given word is one of the most important problems that arise in the study of patterns with variables (note that the concept of patterns with variables is part of several different domains like combinatorics on words (word equations [67], unavoidable patterns [80]), pattern matching [1], language theory [2], learning theory [2, 36, 91, 97, 69, 40], database theory [4], as well as in practice, e.g., extended regular expressions with backreferences [48, 50, 105, 51], used in programming languages like Perl, Java, Python, etc.). Unfortunately, the *matching problem* is NP-complete [2] in general (it is also NP-complete for strongly restricted variants [41, 39] and also intractable in the parameterised setting [42]).

As demonstrated in [98], for the matching problem a paradigm shift yields a very promising algorithmic approach. More precisely, any class of patterns with bounded treewidth (for suitable graph representations) can be matched in polynomial-time. However, computing (and therefore algorithmically exploiting) the treewidth of a pattern is difficult (see the discussion in [39, 98]), which motivates more direct string parameters that bound the treewidth and are simple to compute (virtually all known structural parameters that lead to tractability [27, 39, 98, 107] are of this kind (the efficiently matchable classes investigated in [28] are one of the rare exceptions)). This also establishes an interesting connection between ad-hoc string parameters and the more general (and much better studied) graph parameter treewidth. The locality number is a simple parameter directly defined on words, it bounds the treewidth and the corresponding marking sequences can be seen as instructions for a dynamic programming algorithm. However, compared to other *tractability parameters*, it seems to cover best the treewidth of a word, but whether it can be efficiently computed is unclear. For Loc, the problem to determine the locality number, only exact exponential-time algorithms are known and whether it can be solved in polynomial-time, or whether it is at least fixed-parameter tractable is mentioned as open problems in [27]. In Chapter 4 it is shown that Loc is NP-complete. The treewidth approach was investigated in [15]: there the reductions from MinCutwidth to MinLoc and from MinLoc to MinPathwidth are plugged together. By this, a reduction is obtained which transfers approximation results from MinPathwidth to MinCutwidth, which yields an $\mathcal{O}(\sqrt{\log(\mathsf{opt})}\log(n))$-approximation algorithm for MinCutwidth. This improves, to our knowledge for the first time since 1999, the best approximation for Cutwidth from [76]. In this work only the combinatorial results of k-local words are investigated like the locality of palindromes and repetitions.

The last chapter of this work is dedicated to so called prefix normal words - a generalisation of abelian equivalence. Two words are called abelian equivalent if the amount of each letter is identical in both words, e.g., *rotor* and *torro* are abelian equivalent albeit *banana* and *ananas* are not. Abelian equivalence has been studied with various generalisations and specifications such as abelian-complexity, k-abelian equivalence, avoidability of (k-)abelian powers and many more (cf., e.g., [16, 22, 33, 24, 71, 95, 101, 100]). The number of occurrences of each letter is captured in the Parikh vector (also known as Parikh image or Parikh mapping) ([93]): given a lexicographical order on the alphabet, the i^{th} component of this vector is the amount of the i^{th} letter of the alphabet in a given word. Parikh vectors have been studied in [25, 66, 89] and are generalised to Parikh matrices for saving more information about the word (cf. eg., [90, 104]).

A recent generalisation of abelian equivalence, for words over the binary alphabet $\{0, 1\}$, is prefix normal equivalence [44]. Two binary words are prefix normal equivalent if their maximal numbers of 1s in any factor of length n are equal for all $n \in \mathbb{N}$. [14] showed that this relation is indeed an equivalence relation and, moreover, that each class contains exactly one uniquely determined representative - called a *prefix normal word*. A word w is said to be prefix normal if the prefix of w of any length has at least the number of 1s as any of w's factors of the same length. For instance, the word 110101 is prefix normal but 101101 is not, witnessed by the fact that 11 is a factor but not a prefix. Both words are prefix normal equivalent. In addition to being representatives of these equivalence classes, prefix normal words are also of interest since they are connected to Lyndon words, in the sense that every prefix normal word is a pre-necklace [44]. Furthermore, as shown in [44], the indexed jumbled pattern matching problem (see, e.g., [11, 12, 75]) is connected to prefix normal forms: if the prefix normal forms are given, the indexed jumbled pattern matching problem can be solved in linear time $\mathcal{O}(n)$ of the word length n. The best known algorithm for this problem has a run-time of $\mathcal{O}(n^{1.864})$(see [17]). Consequently, there is also an interest in prefix normal forms from an algorithmic point of view. An algorithm for the computation of all prefix normal words of length n in run-time $\mathcal{O}(n)$ per word is given in [19]. [3] showed that the number of prefix normal words of length n is $2^{n-\Theta(\log^2(n))}$ and the class of a given prefix normal word contains at most $2^{n-O(\sqrt{n\log(n)})}$ elements. A closed formula for the number of prefix normal words is still unknown. In "OEIS" [61] the number of prefix normal words of length n (A194850), a list of binary prefix normal words (A238109), the number of prefix normal palindromes of length n (A308465), and the maximum size of a class of binary words of length n having the same prefix normal form (A238110), can be found. An extension to infinite words is presented in [20].

In Chapter 5 we investigate two conspicuities mentioned in [44, 13]:

palindromes and extension-critical words. Generalising the result of [13] we prove that prefix normal palindromes play a special role since they are not prefix normal equivalent to any other word. Since not all palindromes are prefix normal, as witnessed by 101101, determining the number of prefix normal palindromes is an (unsolved) sub-problem. We show that solving this sub-problem brings us closer to determining the index, i.e., the number of equivalence classes w.r.t. a given word length, of the prefix normal equivalence relation. Moreover, we give a characterisation based on the maximum-ones function for prefix normal palindromes. The notion of extension-critical words is based on an iterative approach: compute the prefix normal words of length $n+1$ based on the prefix normal words of length n. A prefix normal word w is called extension-critical if $w1$ is not prefix normal. For instance, the word 101 is prefix normal but 1011 is not and, thus, 101 is called extension-critical. This means that all non-extension-critical words contribute to the class of prefix normal words of the next word-length. We investigate the set of extension-critical words by introducing an equivalence relation *collapse*, grouping all extension-critical words that are prefix normal equivalent w.r.t. length $n+1$. Finally, we prove that (prefix normal) palindromes and the collapsing relation (extensional-critical words) are related. In contrast to [44], we work with suffix-normal words (least representatives) instead of prefix-normal words. We prove that both notions lead to the same results.

Structure of this work. In Chapter 2, the basic definitions and notions are presented; first, we give the general definitions and afterwards the specific preliminaries for scattered factors, k-locality, and prefix normal words. In Chapter 3, we present the results on scattered factors. This Chapter covers the topics of weakly c-balanced words, scattered factor universality, and the reconstruction from right-bounded block words. In Chapter 4 we first present the NP-completeness proof of k-locality. Afterwards the combinatorial results in this domain are presented. Finally, in Chapter 5 we present the results on prefix normal words, especially on prefix normal palindromes and the collapsing relation.

All results that are contributions of my co-authors in the respective papers are marked with $*$.

Chapter 2

Preliminaries

In this chapter the basic definitions and notions from the domain of combinatorics on words will be introduced divided into parts that reflect the different subdomains *scattered factors, patterns and k-locality,* and *prefix normal words*. For further fundamental information regarding the domain of combinatorics on words the reader may consult [81, 80]. Definitions belonging to single proofs or concepts are given directly in the according context within the next chapters.

2.1 General Definitions

Let $\mathbb{N}$ be the set of natural numbers and $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. By $\mathbb{N}_{>k}$ ($\mathbb{N}_{\geqslant k}$ resp.) denote the set of natural numbers greater than (or greater than and equal to resp.) $k \in \mathbb{N}$. Let $[n]$ denote the set $\{1, \ldots, n\}$ and $[n]_0 = [n] \cup \{0\}$ for an $n \in \mathbb{N}$ and abbreviate $[n]\backslash[k-1]$ and $[n]\backslash[k]$ by $[n]_{\geqslant k}$ and $[n]_{>k}$, respectively.

An *alphabet* $\mathcal{A}$ is a set of symbols and $\mathcal{A}^*$ denotes the free monoid over $\mathcal{A}$ with the neutral element ε called the *empty word* under concatenation. The free semigroup over $\mathcal{A}$ is $\mathcal{A}^+ = \mathcal{A}^* \backslash \{\varepsilon\}$. A *word* is an element of $\mathcal{A}^*$, i.e., a finite sequence of letters from $\mathcal{A}$. The set of all infinite words $\mathtt{a}_1\mathtt{a}_2\ldots$ with $\mathtt{a}_i \in \mathcal{A}$ is denoted by $\mathcal{A}^\omega$. In this work finite and infinite alphabets will be distinguished by the notion: Σ will always represent a finite alphabet and the elements are called *letters*, whereas potentially infinite alphabets are denoted by X and the elements are called *variables*. Assume, if both alphabets are considered in a setting, that they are disjoint, i.e., $\Sigma \cap X = \emptyset$. For given Σ and X, words α in $(X \cup \Sigma)^*$ are called *patterns* and they are called *terminal-free* in the case $\alpha \in X^*$. Let $\textsc{Pat}_\Sigma = (X \cup \Sigma)^*$

be the set of patterns and define $\text{PAT} = \bigcup_{\Sigma} \text{PAT}_{\Sigma}$ for a given set of variables X. Variable-free patterns, i.e., patterns $\alpha \in \Sigma^*$, are called *terminal-words* or simply words.

The *length* of a word $w \in \mathcal{A}^*$ is denoted by $|w|$. For $k \in \mathbb{N}$ set $\mathcal{A}^{\leqslant k} = \{w \in \mathcal{A}^* || w| \leqslant k\}$ and $\mathcal{A}^k = \{w \in \mathcal{A}^* || w| = k\}$. A word $u \in \mathcal{A}^*$ is a *factor* of $w \in \mathcal{A}^*$ if $w = xuy$ for some $x, y \in \mathcal{A}^*$. In the case $x = \varepsilon$, u is called a *prefix* of w, and called a *suffix* of w in the case $y = \varepsilon$. Factors, prefixes, and suffixes are called *proper* if they are neither the empty word nor the complete word itself. Let $\text{Fact}(w)$, $\text{Pref}(w), \text{Suff}(w)$ denote the set of w's factors, prefixes, and suffixes respectively. Moreover, for all sets, the indexes $k, \leqslant k, < k, \geqslant k, > k$ denote a length restriction of the contained words, e.g., $\text{Fact}_{\leqslant k}(w) = \{v \in \text{Fact}(w) | \, |v| \leqslant k\}$. Moreover, $\text{PropFact}(w), \text{PropPref}(w)$, and $\text{PropSuff}(w)$ denote the sets of proper factors, prefixes, and suffixes respectively. For better readability, we are using *dots* to highlight a factor which is of interest in the given context. For instance, $\mathtt{b.an.an.a}$ is for highlighting that the factor $\mathtt{an}$ occurs twice in the word $\mathtt{banana}$.

The i^{th} letter of $w \in \mathcal{A}^*$ is denoted by $w[i]$ for $i \in [|w|]$ and set $w[i..j] = w[i]w[i+1] \cdots w[j]$ for $1 \leqslant i \leqslant j \leqslant |w|$. By convention, $w[i..j] = \varepsilon$, if $i > j$. By $|w|_{\mathtt{a}}$ denote the number of occurrences of the letter $\mathtt{a} \in \mathcal{A}$ in $w \in \mathcal{A}^*$.

Since for patterns two disjoint alphabets are necessary, the access of the symbols of a word $w \in (X \cup \Sigma)^*$ is split into

$$\text{alph}(w) = \{\mathtt{a} \in \Sigma | \, |w|_{\mathtt{a}} > 0\} \text{ and } \text{var}(w) = \{\mathtt{x} \in X | \, |w|_{\mathtt{x}} > 0\}.$$

Thus, all symbols in the word $w \in (X \cup \Sigma)^*$ are given by $\text{alph}(w) \cup \text{var}(w)$, and $\text{var}(w) = \emptyset$ holds for (terminal) words.

The reverse of $w = w[1] \cdots w[n] \in \Sigma^*$ is defined by $w^R = w[n] \cdots w[1]$. A *palindrome* is a word w with $w = w^R$. For a word $u \in \mathcal{A}^*$ we define $u^0 = \varepsilon, u^{i+1} = u^i u$, for $i \in \mathbb{N}$. A word $w \in \mathcal{A}^*$ is called *power* (repetition) of a word $u \in \mathcal{A}^*$, if $w = u^t$ for some $t \in \mathbb{N}_{>1}$.

Given a word $w \in \Sigma^*$ and a property $P : \Sigma \to \{0, 1\}$, a *P-block* in w is a factor u of w such that all of u's letters fulfil P and u is not contained in a larger factor v of w also fulfilling P. Formally, $u \in \text{Fact}(w)$ is a P-block if $P(u[i]) = 1$ for all $i \in [|u|]$ and for all $v \in \text{Fact}(w)$ with $u \in \text{PropFact}(v)$ there exists $j \in [|v|]$ with $P(v[j]) = 0$. For convenience the defining property will be abbreviated if it is clear from the context: For instance the word abaaabaabb has three a-blocks and three b-blocks (instead of defining a property $P_{\mathtt{a}}$ with $P_{\mathtt{a}}(u[i]) = 1$ iff $u[i] = \mathtt{a}$ and speaking of three $P_{\mathtt{a}}$-blocks). Let $[x]^{\flat}$ denote an x-block for $x \in \mathcal{A}$ and $x^{\flat}$ similarly a block that contains at least one x. The first notation seems artificial since the empty block only corresponds to an inserted ε in the word, but it allows to define classes of words and patterns: if w is in $[\mathtt{a}]^{\flat}\mathtt{b}[\mathtt{a}]^{\flat}$ then w is defined by the regular expression $\{\mathtt{a}\}^*\mathtt{b}\{\mathtt{a}\}^*$. In the context of k-locality this definition will be refined such that the benefits become clearer.

A word $u \in \mathcal{A}^*$ is a *conjugate* of a word $w \in \mathcal{A}^*$ if there exist $x, y \in \mathcal{A}^*$ with $w = xy$ and $u = yx$. If Σ has a total order $<$ - that is implicitly extended to the lexicographical order $<$ on Σ^* - Lyndon words are defined as follows: a word $w \in \Sigma^*$ is called *Lyndon word* (or simply *Lyndon*) if w is the strictly lexicographically smallest word amongst all its conjugates, i.e., w is lexicographically smaller than yx for all factorisations $w = xy$ for $x, y \in \Sigma^+$. A special form of Lyndon words are the *right-bounded block* words that are of the form $\mathtt{a}^{\ell}\mathtt{b}$ for $\mathtt{a}, \mathtt{b} \in \Sigma$ with $\mathtt{a} < \mathtt{b}$, and $\ell \in \mathbb{N}$.

For $\ell, n \in \mathbb{N}$ and $w_1, \ldots, w_n \in \mathcal{A}^*$, define $\langle w_1, \ldots, w_n \rangle_{\ell}$ as the set of all words $w = x_1 \cdots x_{\ell}$ with $x_i \in \{w_1, \ldots, w_n\}$. Let $\langle w_1, \ldots, w_n \rangle = \bigcup_{\ell \in \mathbb{N}} \langle w_1, \ldots, w_n \rangle_{\ell}$.

Over a binary alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ a word $w \in \Sigma^*$ is called *weakly c-balanced* if $||w|_{\mathtt{a}} - |w|_{\mathtt{b}}| = c$, i.e., the difference between the amount of a and b in w is $c \in \mathbb{N}_0$. For instance, abaa is weakly 2-balanced, aba is weakly 1-balanced, while abbaba is weakly 0-balanced. Let Σ^*_{wzb} denote the set of weakly 0-balanced words.

A function $f : \mathcal{A}_1^* \to \mathcal{A}_2^*$ for alphabets $\mathcal{A}_1, \mathcal{A}_2$ is called a *morphism* if $f(uv) = f(u)f(v)$ for $u, v \in \mathcal{A}_1^*$. Notice that it suffices to define morphisms on the letters of $\mathcal{A}_1$ to define it on longer words.

Finally, since some of the results are of algorithmic nature, we introduce the necessary setting. The computational model we use is the standard unit-cost RAM with logarithmic word size: for an input of size n, each memory word can hold $\log n$ bits. Arithmetic and bitwise operations with numbers in $[n]$ are, thus, assumed to take constant time. In the upcoming algorithmic problems, we assume that the processed words are sequences of integers (called letters or symbols, each fitting in a constant number of memory words). In general, after a linear time preprocessing, we can assume that for an input word of length n over an alphabet Σ, its letters are in the set $\{1, 2, ..., |\Sigma|\}$ (where, clearly, $|\Sigma| \leqslant n$). For a more detailed discussion see, e.g., [23].

2.2 Definitions Regarding Scattered Factors

In this subsection the main definitions surrounding the domain of scattered factors (also known as (scattered) subwords and subsequences) are given. In contrast to factors, scattered factors are not necessarily consecutive parts of the words: a scattered factor can be obtained by deleting some letters of a given word w and concatenating the remaining parts in order. Thus, every factor is especially a scattered factor. In the context of scattered factors only finite alphabets are going to be considered and, consequently, Σ will be the only alphabet.

2.2.1 Definition. A word $v = v[1] \cdots v[k] \in \Sigma^k$, $k \in \mathbb{N}_0$, is a *scattered factor* of a word $w \in \Sigma^*$ if there exist $k+1$, not necessarily distinct and possibly empty, words $x_1, \ldots, x_{k+1} \in \Sigma^*$ such that $w = x_1 v[1] \cdots x_k v[k] x_{k+1}$. Let $\mathrm{ScatFact}(w)$ be the set of all scattered factors of w and define $\mathrm{ScatFact}_k(w)$ ($\mathrm{ScatFact}_{\leqslant k}$) as the set of all scattered factors of w of length (up to) $k \in \mathbb{N}$. A word $u \in \Sigma^*$ is a *common scattered factor* of two words $w, v \in \Sigma^*$, if $u \in \mathrm{ScatFact}(w) \cap \mathrm{ScatFact}(v)$. Accordingly, the word u is an *discommon scattered factor* of w and v if u is a scattered factor of exactly one of them.

Regarding scattered factors it is an important difference whether the multiplicity of the scattered factor in a word is taken into account (cf., e.g., [85]). Notice that the scattered factor set in Definition 2.2.1 is an ordinary set without multiplicities.

The scattered factor sets $\mathrm{ScatFact}_{\leqslant k}(w)$ and $\mathrm{ScatFact}_k(w)$ are also known as *full k-spectrum* and, respectively, *k-spectrum* of a word $w \in \Sigma^*$ (see [6], [85], [103]). Obviously the k-spectrum is empty for $k > |w|$ and contains exactly w's letters for $k = 1$ and only w for $k = |w|$.

A special kind of scattered factors is given by the projections. A word u is the projection of a word w on $\Delta \subseteq \Sigma$ if u is obtained by removing all letters from w belonging to $\mathrm{alph}(w)\backslash\Delta$. For instance, `bnn` is the projection of `banana` on $\{\mathtt{b},\mathtt{n}\}$.

2.2.2 Definition. A word $v = v_1 \cdots v_k \in \Sigma^*$ is a *projection* w_Δ of a word $w \in \Sigma^*$ on $\Delta \subseteq \Sigma$ if $v \in \mathrm{ScatFact}(w) \cap \Delta^*$ and there exist $x_1 \cdots x_{k+1} \in (\mathrm{alph}(w)\backslash\Delta)^*$ with $w = x_1 v_1 \cdots x_k v_k x_{k+1}$.

Alternatively, projections can be defined by erasing morphisms. Let $h_\Delta : \Sigma^* \to \Delta^*$ be the morphism defined by $h(\mathtt{a}) = \mathtt{a}$ for all $\mathtt{a} \in \Delta$ and $h(\mathtt{b}) = \varepsilon$ for $\mathtt{b} \in \Sigma\backslash\Delta$. Then w_Δ is defined as $h(w)$.

Simon [108] defines the congruence $\sim_k$ where $u \sim_k v$ if $\mathrm{ScatFact}_{\leqslant k}(u) = \mathrm{ScatFact}_{\leqslant k}(v)$ for $u, v \in \Sigma^*$ and $k \in \mathbb{N}_0$. Since the k-spectrum is a subset of Σ^k for $k \in \mathbb{N}_0$, the questions arise for which words $u \in \Sigma^*$, $\mathrm{ScatFact}_k(u) = \Sigma^k$ holds, and of which form u is. In [64] the notion of richness in the context of scattered factors is introduced: a word $u \in \Sigma^*$ is called *rich* if $\mathrm{alph}(u) = \Sigma$ holds. Notice that this is equivalent to $\mathrm{ScatFact}_1(u) = \Sigma$. This definition is extended to *ℓ-richness*: a word is ℓ-rich if it is a concatenation of ℓ possibly different rich (w.r.t. the same given alphabet Σ) words. The concatenation of ℓ rich words is also fundamental in the *arch factorisation* introduced by Hebrard [57]. Since this factorisation is fundamental and the basis for most of the results in Section 3.2, it is recalled here.

2.2.3 Definition ([57])**.** For $w \in \Sigma^*$ the *arch factorisation* of w is given by

$$w = \mathrm{ar}_w(1) \cdots \mathrm{ar}_w(k) r(w)$$

for a $k \in \mathbb{N}_0$ with

▷ $\mathrm{alph}(\mathrm{ar}_w(i)) = \Sigma$,

▷ $\mathrm{ar}_w(i)[|\mathrm{ar}_w(i)|] \notin \mathrm{alph}(\mathrm{ar}_w(i)[1..|\mathrm{ar}_w(i)| - 1]$ for all $i \in [n]$, and

▷ $\mathrm{alph}(r(w)) \subset \Sigma$.

The words $\mathrm{ar}_w(i)$ are called *arches* of w and $r(w)$ is called the *rest*. Define the word containing the unique last letters of the arches by

$$m(w) = \mathrm{ar}_w(1)[|\mathrm{ar}_w(1)|] \cdots \mathrm{ar}_w(k)[|\mathrm{ar}_w(k)|].$$

As by the notion stressed, the focus each in the arch factorisation and in the factorisation into rich words is on decomposing the word into specific factors. Taking the aforementioned perspective of asking whether the set of scattered factors of a given length $k \in \mathbb{N}$ equals Σ^k leads to the following notion.

2.2.4 Definition. A word $s \in \Sigma^*$ is called *k-universal,* for $k \in \mathbb{N}_0$, if $\text{ScatFact}_k(w) = \Sigma^k$. For convenience a word is simply called *universal* if it is 1-universal. Define the *universality-index* $\iota(w)$ of $w \in \Sigma^*$ as the largest k such that w is k-universal.

In the context of Simon congruence also the following definition is helpful: it defines a representative amongst all words with the same k-spectrum for a given $k \in \mathbb{N}$.

2.2.5 Definition. The *shortlex normal form* of a word $w \in \Sigma^*$ w.r.t. $\sim_k$, where Σ is an ordered alphabet, is the shortest word u with $u \sim_k w$ which is also lexicographically smallest (w.r.t. the given order on Σ) amongst all words $v \sim_k w$ with $|v| = |u|$.

It is a simple observation that a word is ℓ-rich if and only if it is ℓ-universal and a rich-factorisation, i.e., the factorisation of an ℓ-rich word into ℓ rich words, can also be efficiently obtained. Nevertheless, we will use the name of ℓ-universal word rather than ℓ-rich word, as richness is used as a name also for other properties of words, such as the property of a word of length n to have $n+1$ distinct palindromic factors, see, e.g., [31, 82] and the references therein. As w is ℓ-universal iff w is the concatenation of $\ell \in \mathbb{N}$ universal words it follows immediately that, if w is over the ordered alphabet $\Sigma = \{1 < 2 < \ldots < \sigma\}$ and it is ℓ-universal then its shortlex normal form w.r.t. $\sim_\ell$ is $(1 \cdot 2 \cdots \sigma)^\ell$ (as this is the shortest and lexicographically smallest ℓ-universal word).

2.2.6 Remark. Notice that k-universality is always w.r.t. a given alphabet Σ: the word $\texttt{abcba}$ is 1-universal for $\Sigma = \{\texttt{a}, \texttt{b}, \texttt{c}\}$ but 2-universal for $\Sigma \backslash \{\texttt{c}\}$. If it is clear from the context, we omit explicit mention of Σ, otherwise we state it.

The following observation leads to the next definition: considering the word $w = \texttt{abc}$ we notice that it is 1-universal and w^s is s-universal for all $s \in \mathbb{N}$; on the other hand $v = \texttt{ababcc}$ which is also 1-universal behaves differently on repeating it - v^2 is 3-universal.

2.2.7 Definition. A word $w \in \Sigma^*$ is called *k-circular universal* if a conjugate of w is k-universal. Define the *circular universality index* $\zeta(w)$ of w as the largest k such that w is k-circular universal. Again we call a word simply circular-universal if it is 1-circular universal.

In the previous example v is 2-circular universal since v is a conjugate of $\mathtt{abccab}$. Another natural generalisation of universality is to consider sets of words. The words $w_1 = \mathtt{ab}, w_2 = \mathtt{bc}, w_3 = \mathtt{ca}$ over $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$ are not 1-universal but, e.g., $w_1 w_2$ is 1-universal and $w_1 w_3 w_2$ is 2-universal.

2.2.8 Definition. Let $n \in \mathbb{N}$. The set $S = \{w_1, \ldots, w_n | \, w_i \in \Sigma^*, i \in [n]\}$ is called *universal* if there exists a $u \in \langle w_1, \ldots, w_n \rangle$ which is universal. Moreover, S is called k-universal if there exist $p \in \mathbb{N}_0$ and $u_1, \ldots, u_p \in \langle w_1, \ldots, w_n \rangle$ such that $u_1 \cdots u_p$ is k-universal.

2.2.9 Remark. It is worth noting that, unlike the case of factor universality of words and partial words [88, 10, 18, 54], in the case of scattered factor universality of words it does not make sense to try to identify a k-universal word $w \in \Sigma^*$, for $k \in \mathbb{N}_0$, such that each word from Σ^k occurs *exactly once* as scattered factor of w. Indeed, if $|w| \geqslant k + |\Sigma|$ then there exists a word from Σ^k which occurs at least twice as a scattered factor of w. Moreover, the shortest word which is k-universal has length $k|\Sigma|$, because each k-universal word must have $\mathtt{a}^k$ as a scattered factor, for all $\mathtt{a} \in \Sigma$. As $k|\Sigma| \geqslant k + |\Sigma|$ for $k, |\Sigma| \in \mathbb{N}_{\geqslant 2}$, all k-universal words have scattered factors occurring more than once: there exists $i, j \in [|\Sigma| + 1]$ such that $w[i] = w[j]$ and $i \neq j$. Then $w[i]w[|\Sigma| + 2..|\Sigma| + k]$ and $w[j]w[|\Sigma| + 2..|\Sigma| + k]$ are both scattered factors of w and $w[i]w[|\Sigma| + 2..|\Sigma| + k] = w[j]w[|\Sigma| + 2..|\Sigma| + k]$.

For the last definition of this subsection the multiplicities of scattered factors are of interest. Here the notion of the binomial coefficient of two words is important.

2.2.10 Definition. For $u, v \in \Sigma^*$, the number of different occurrences of v as a scattered factor of u is denoted by $\binom{u}{v}$.

2.2.11 Remark. Notice that $|w|_{\mathtt{x}} = \binom{w}{\mathtt{x}}$ for all $\mathtt{x} \in \Sigma$.

2.3 Definitions Regarding *k*-Locality

In this subsection the main definitions regarding patterns and especially the notion of *k*-locality will be given. Since patterns are investigated, both a finite alphabet Σ and an potentially infinite alphabet X are considered.

The locality of a word is a measurement of how the position of letters are w.r.t. each other, e.g., in $\mathtt{abcbabcab}$ the letter $\mathtt{b}$ only occurs next to $\mathtt{a}$. The locality can be investigated for (terminal) words as well as for patterns. For this reason the more general definition for patterns are introduced before the peculiarities for (terminal) words are enlarged upon.

2.3.1 Definition. For a pattern $\alpha \in \textsc{Pat}_\Sigma$, a *substitution* is a morphism $h : \mathrm{var}(\alpha) \to \Sigma^*$. This notion can be extended to $\mathrm{var}(\alpha) \cup \mathrm{alph}(\alpha)$ by $h(\mathtt{a}) = \mathtt{a}$ for $\mathtt{a} \in \Sigma$. The substitution is called *non-erasing* if it maps into Σ^+ and *erasing* otherwise. The *pattern language* $L(\alpha)$ of α is defined by $\{h(\alpha) \mid h \text{ is a substitution for } \alpha\}$.

In this work only non-erasing substitutions are considered.

2.3.2 Definition. The *matching problem*, denoted by $\textsc{Match}$, is to decide for a given pattern α and word w, whether there exists a substitution h with $h(\alpha) = w$. For any $P \subseteq \textsc{Pat}$, the *matching problem for P* is to decide for a given pattern $\alpha \in P$ and word w, whether there exists a substitution h with $h(\alpha) = w$.

Regarding patterns the focus of the measurement *locality* is in solving the pattern matching problem efficiently for subclasses of patterns. Therefore, the letters are not taken into account. The following definition captures the structure of the variables within a pattern by removing the letters.

2.3.3 Definition. Let $\beta \in (X \cup \Sigma)^*$. The *skeleton* of β is the (unique and terminal-free) pattern $\alpha = \beta_X$, i.e., the projection of β on $\mathrm{var}(\beta)$.

The relative positions of letters is defined via a *marking* process. Given a sequence of all variables, the variables are substituted one by one by a *marked* version of this variable. Thus, variables can be distinguished by how many occurrences are neighbours to already marked variables.

2.3.4 Definition. Let $\overline{X} = \{\overline{x} \mid x \in X\}$ be the set of *marked variables*. For the skeleton α of a pattern $\beta \in (X \cup \Sigma)^*$, a *marking sequence* of the variables occurring in β, is an enumeration $x_1, x_2, \ldots, x_{|\operatorname{var}(\beta)|}$ of $\operatorname{var}(\beta)$. A variable x_i is called *marked at point* $k \in \mathbb{N}$ (both in β and α) if $i \leqslant k$. Moreover, we define α_k, *the marked skeleton of* β *at point* k, as the string obtained from α by replacing all x_i with $i \leqslant k$ by $\overline{x_i}$. A *marked block* in α_k is a block u with the property $P_{\text{block}} : X \cup \overline{X} \to \{0, 1\}$ with $P_{\text{block}}(y) = 1$ iff $y \in \overline{X}$.

Marking the skeleton $\alpha = y_1 \cdots y_k \in X^*$ of a pattern β with a marking sequence σ induces a sequence of natural numbers $b = (b_1, \ldots, b_\ell)$ where b_i is the number of marked blocks at stage i for all $i \in [\ell]$ and $k, \ell \in \mathbb{N}$. This sequence is called *blocksequence*.

Notice that in the context of k-locality, repetitions of letters in a word are of minor interest: if a word w contains an $\mathtt{a}$-block for an $\mathtt{a} \in \Sigma$ all these $\mathtt{a}$ are marked simultaneously when $\mathtt{a}$ is marked. This observation leads to the following definition.

2.3.5 Definition. For $w = x_1^{k_1} x_2^{k_2} \cdots x_\ell^{k_\ell} \in (\Sigma \cup X)^*$ with $k_i, \ell \in \mathbb{N}$, $i \in [\ell]$, the *print* of w is defined by $x_1 \cdots x_\ell$ assumed that $x_j \neq x_{j+1}$ for $j \in [\ell - 1]$. The print is also called *condensed form* of w and a word is called *condensed* if it is its own print.

The idea of marking non-terminalfree patterns (patterns with letters) is to mark the letters implicitly if they are neighbouring marked variables. For this reason the notations $[x]^\flat$ and $x^\flat$ are extended in this setting to $(\{x\} \cup \Sigma)^*$ and $(\{x\} \cup \Sigma)^+$ resp.

Using the idea of a marking sequence, we can now define the k-locality of a pattern.

2.3.6 Definition. Let $\beta \in (X \cup \Sigma)^*$ be pattern with skeleton $\alpha \in \Sigma^\ell$, for $\ell \in \mathbb{N}$, and σ be a marking sequence for β. Then α is *k-local* w.r.t. σ for $k \in \mathbb{N}_0$ if for all $i \leqslant \ell$, α_i, the marked skeleton of α at point i, has at most k marked blocks. The *locality-number* of α is the smallest $k \in \mathbb{N}$ such that there exists a marking sequence σ and α is k-local w.r.t. σ. Let $\operatorname{loc}_\sigma(\alpha)$ denote the k such that α is k-local w.r.t. σ and let $\operatorname{loc}(\alpha)$ denote the locality number of α. A pattern is called *strictly* k-local if it is k-local but not $(k-1)$-local. Let $\textsc{Pat}_{k\text{-loc}}$ denote the class of k-local patterns. The minimal k such that a pattern β is k-local is called the *locality number* of β.

The definition of k-locality is based generally on patterns that may contain letters as well as variables. In specific, the definition is reduced to the skeleton of a pattern, whereas the possible infinity of the alphabet X is not taken into account: the locality is a property of a single finite pattern that contains only finitely many different variables. Thus, also words $w \in \Sigma^*$ have a locality-number if the letters in Σ are marked (substituted by copies from $\overline{\Sigma}$) instead of the variables.

2.4 Definitions Regarding Prefix Normal Words

Following [44], for prefix normal words only binary alphabets are considered, namely $\Sigma = \{0, 1\}$ with the fixed lexicographic order induced by $0 < 1$ on Σ. In analogy to binary numbers we call a word $w \in \Sigma^n$ *odd* if $w[n] = 1$ and *even* otherwise.

For a function $f : [n] \to \Delta$, for $n \in \mathbb{N}_0$, and an arbitrary alphabet Δ the concatenation of the images defines a finite word $\text{serialize}(f) = f(1)f(2)\cdots f(n) \in \Delta^*$. Since serialize is bijective, serialize(f) is identified with f and, thus, f is used in both cases (as long as it is clear from the context). This definition allows us to access f's *reversed function* $g : [n] \to \Delta; k \mapsto f(n-k+1)$ easily by f^R.

2.4.1 Definition. The *maximum-ones function* is defined for a word $w \in \Sigma^*$ by

$$f_w : [|w|]_0 \to [|w|]_0;\ k \mapsto \max\left\{\, |v|_1 \mid v \in \text{Fact}_k(w) \right\},$$

giving for each $k \in [|w|]_0$ the maximum number of 1s occuring in a factor of length k. Likewise the *prefix-ones and suffix-ones functions* are defined by

$$p_w : [|w|]_0 \to [|w|]_0; k \mapsto |\,\text{Pref}_k(w)|_1 \quad \text{and}$$
$$s_w : [|w|]_0 \to [|w|]_0; k \mapsto |\,\text{Suff}_k(w)|_1.$$

2.4.2 Definition. Two words $u, v \in \Sigma^n$ are called *prefix normal equivalent* ($u \equiv_n v$) if $f_u = f_v$ holds and the equivalence class of v is denoted by $[v]_\equiv = \{u \in \Sigma^n | \, u \equiv_n v\}$. A word $w \in \Sigma^*$ is called *prefix (suffix) normal* iff $f_w = p_w$ ($f_w = s_w$ resp.) holds. Let $\sigma(w) = \sum_{i \in [n]} f_w(i)$ denote the *maximal-one sum* of $w \in \Sigma^n$.

2.4.3 Remark. Notice that $s_w = p_{w^R}$, $f_w = f_{w^R}$, $p_w(i), s_w(i) \leqslant f_w(i)$ hold for all $i \in \mathbb{N}_0$. By $p_{w^R} = s_w$ and $f_w = f_{w^R}$, it follows immediately that a word $w \in \Sigma^*$ is prefix normal iff its reversal is suffix normal.

The authors of [44] showed that for each word $w \in \Sigma^*$ there exists exactly one $w' \in [w]_\equiv$ that is prefix normal - the prefix normal form of w. We introduce the concept of *least representative*, which is the lexicographically smallest element of a class and, thus, also unique.

2.4.4 Definition. A word $w \in \Sigma^n$ is called the *least-representative* of the class $[w]_\equiv$ if all other elements in $[w]_\equiv$ are lexicographically larger, i.e., $w \leqslant v$ for all $v \in [w]_\equiv$.

As mentioned in [14] palindromes play a special role: immediately by $w = w^R$ for $w \in \Sigma^*$, we have $p_w = s_w$, i.e., palindromes are the only words that can be prefix and suffix normal. Notice that not all palindromes are prefix normal as witnessed by 101101.

2.4.5 Definition. A palindrome is called *prefix normal palindrome* if it is prefix normal. Let $\mathrm{NPal}(n)$ denote the set of all prefix normal palindromes of length $n \in \mathbb{N}$ and set $\mathrm{npal}(n) = |\mathrm{NPal}(n)|$. Let $\mathrm{Pal}(n)$ be the set of all palindromes of length $n \in \mathbb{N}$.

word length	prefix normal palindromes
1	$0, 1$
2	$0^2, 1^2$
3	$0^3, 101, 1^3$
4	$0^4, 1001, 1^4$
5	$0^5, 10001, 10101, 11011, 1^5$
6	$0^6, 100001, 110011, 1^6$

Table 2.1. Prefix normal palindromes.

Chapter 3

Scattered Factors

In this chapter we are investigating three topics regarding scattered factors. As a reminder $u \in \Sigma^*$ is a scattered factor of $w \in \Sigma^*$ if there exist $v_1, \ldots, v_{|u|+1} \in \Sigma^*$ with $w = v_1 u[1] v_2 u[2] \cdots v_{|u|} u[|u|] v_{|u|+1}$. Two long outstanding problems are determining the index of Simon congruence and the reconstruction problem for scattered factors. In the first two sections we tackle the first problem from a new perspective: instead of investigating the index itself we explore which cardinalities of k-spectra are possible and which words lead to which cardinalities of k-spectra. This approach can be seen as a bridge since only words leading to k-spectra with the same cardinality may fall in the same equivalence class. In the second section we look deeper into the maximal possible cardinality a k-spectrum may have, i.e., the scattered factor universality. We finish this chapter with a new approach for the reconstruction problem: while so far only words of the same length has been investigated regarding the problem, we allow words of different length - namely right-bounded block words - for reconstructing a word uniquely.

The following three well-known properties of scattered factors are mentioned before the subsections since they are of general interest. It is worth noting that if u is a scattered factor of w, and v is a scattered factor of u, then v is a scattered factor of w. Additionally, notice two important symmetries regarding k-spectra. For $w \in \Sigma^*$ and a morphic permutation f on Σ (i.e., a renaming) we have

$$\begin{aligned} \mathrm{ScatFact}(w^R) &= \{u^R \mid u \in \mathrm{ScatFact}(w)\} \text{ and} \\ \mathrm{ScatFact}(f(w)) &= \{f(u) \mid u \in \mathrm{ScatFact}(w)\}. \end{aligned}$$

Thus, from a structural point of view, it is sufficient to consider only one representative from the equivalence classes induced by the equivalence relation where w_1 is equivalent to w_2 whenever w_2 is obtained by a composition of reversals and renamings from w_1. Considering an order on Σ, we choose the lexicographically smallest word from each class as representative.

3.1 Weakly c-balanced Words

In the current subsection which is mainly based on [26], we consider the combinatorial properties of k-spectra of weakly-c-balanced finite words over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. In particular, we are interested in the cardinalities of the k-spectra and in the question: which cardinalities are (not) possible? Since the k-spectra of $\mathtt{a}^n$ and $\mathtt{b}^n$ are just $\mathtt{a}^k$ and $\mathtt{b}^k$ respectively for all $n \in \mathbb{N}_0$ and $k \in [n]_0$, we assume $|w|_{\mathtt{a}}, |w|_{\mathtt{b}} > 0$ for given $w \in \Sigma^*$. It is a straightforward observation that not every subset of Σ^k is a k-spectrum of some word w. For example, for $k = 2$, $\mathtt{aa}$ and $\mathtt{bb}$ can only be scattered factors of a word containing both $\mathtt{a}$s and $\mathtt{b}$s, and therefore having either $\mathtt{ab}$ or $\mathtt{ba}$ as a scattered factor as well. Thus, there is no word w such that $\mathrm{ScatFact}_2(w) = \{\mathtt{aa}, \mathtt{bb}\}$.

In general, for any word containing only $\mathtt{a}$s or only $\mathtt{b}$s, there will be exactly one scattered factor of each length, while for words containing both $\mathtt{a}$s and $\mathtt{b}$s, the smallest k-spectra are realised for words of the form $w = \mathtt{a}^n\mathtt{b}$ (up to renaming and reversal), for which $\mathrm{ScatFact}_k(w) = \{\mathtt{a}^k, \mathtt{a}^{k-1}\mathtt{b}\}$ for each $k \in [|w| - 1]$. On the other hand, as Proposition 3.1.5 shows, the maximal k-spectra are those containing all words of length k – and, hence, have size 2^k, achieved by, e.g., $w = (\mathtt{ab})^n$ for $n \geqslant k$. These *k-universal words* are further investigated in Section 3.2. Note that when weakly-0-balanced words are considered, the same maximum applies, since $(\mathtt{ab})^n$ is weakly-0-balanced, while the minimum does not, since $\mathtt{a}^n\mathtt{b}$ is not weakly-0-balanced.

It is straightforward to enumerate all possible k-spectra, and describe the words realising them for $k \leqslant 2$, hence, we shall generally consider only k-spectra in the sequel for which $k \geqslant 3$. Our first result generalises the previous observation about minimal-size k-spectra.

3.1.1 Theorem. *For $k \in \mathbb{N}_{\geqslant 3}$, $c \in [k-1]_0$, $i \in [c]_0$, and a weakly-c-balanced word $w \in \Sigma^{2k-c}$, we have $|\mathrm{ScatFact}_{k-i}(w)| \geqslant k - c + 1$, where equality* *holds if and only if* $w \in \{\mathtt{a}^k\mathtt{b}^{k-c}, \mathtt{a}^{k-c}\mathtt{b}^k, \mathtt{b}^k\mathtt{a}^{k-c}, \mathtt{b}^{k-c}\mathtt{a}^k\}$*. Moreover, if* $w \in \Sigma^{2k}_{\mathrm{wzb}} \backslash \{\mathtt{a}^k\mathtt{b}^k\}$*, then* $|\operatorname{ScatFact}_k(w)| \geqslant k+3$*.*

Proof. First, consider only weakly-0-balanced words, i.e., $c = 0$ and w.l.o.g. only $w = \mathtt{a}^k\mathtt{b}^k$ (this is the lexicographically smallest word in the class of words obtained by renaming the letters or reversal). The cases $k = 1$ and $k = 2$ are the induction basis.

The word $\mathtt{a}^k\mathtt{b}^k$ has obviously all $\mathtt{a}^r\mathtt{b}^s$ for $r, s \in [k]_0$ as scattered factors, and there are $k+1$ of these. This proves the $\Leftarrow$-direction.

Consider now a word $w \in \Sigma^{2k}_{\mathrm{wzb}} \backslash \{\mathtt{a}^k\mathtt{b}^k, \mathtt{b}^k\mathtt{a}^k\}$.

Since w is not $\mathtt{a}^k\mathtt{b}^k$, w contains a factor $\mathtt{a}\mathtt{b}^\ell\mathtt{a}$ or $\mathtt{b}\mathtt{a}^\ell\mathtt{b}$ for an existing $\ell \in \mathbb{N}$. Assume w.l.o.g. that $w = x\mathtt{aba}y$ holds for some $x, y \in \Sigma^*$ with $|x| + |y| = 2k-3$. By $w \in \Sigma^{2k}_{\mathrm{wzb}}$, it follows that $|x|_\mathtt{b}$ or $|y|_\mathtt{b}$ is not zero. Choose w.l.o.g. $z_1, z_2 \in \Sigma^*$ with $y = z_1\mathtt{b}z_2$ which implies $w = x\mathtt{aba}z_1\mathtt{b}z_2$. Consequently, $|xz_1z_2|_a = |xz_1z_2|_b = k-2$ holds.

case 1: $xz_1z_2 = \mathtt{a}^{k-2}\mathtt{b}^{k-2}$

By induction hypothesis, $|\operatorname{ScatFact}_{k-2}(xz_1z_2)| = (k-2)+1 = k-1$. Let u be a scattered factor of xz_1z_2 of length $k-2$. Then there exist u_1, u_2, and u_3 such that u_1 is a scattered factor of x, u_2 of z_1, and u_3 of z_3 respectively. Consequently,

$$u_1\mathtt{aa}u_2u_3, \quad u_1\mathtt{ab}u_2u_3, \quad \text{and} \quad u_1\mathtt{ba}u_2u_3$$

are different elements of $\operatorname{ScatFact}_k(w)$. Each scattered factor of xz_1z_2 is of the form $\mathtt{a}^r\mathtt{b}^s$ for $r, s \in [k-2]_0$. We will now explore in which cases the aforementioned scattered factors are different. Consider $u = u_1u_2u_3 = \mathtt{a}^r\mathtt{b}^s$ and $u' = u'_1u'_2u'_3 = \mathtt{a}^{r'}\mathtt{b}^{s'}$ to be different scattered factors of this form, i.e., $r \neq r'$ and $s \neq s'$. Set

$$\begin{aligned} \alpha_1 &= u_1\mathtt{aa}u_2u_3, & \beta_1 &= u'_1\mathtt{aa}u'_2u'_3,\\ \alpha_2 &= u_1\mathtt{ba}u_2u_3, & \beta_2 &= u'_1\mathtt{ba}u'_2u'_3,\\ \alpha_3 &= u_1\mathtt{ab}u_2u_3, & \beta_3 &= u_1\mathtt{ab}u_2u_3. \end{aligned}$$

If $u_1 = \mathtt{a}^{r_1}$, $u_2u_3 = \mathtt{a}^{r_2}\mathtt{b}^s$ and $u'_1 = \mathtt{a}^{r'_1}$, $u'_2u'_3 = \mathtt{a}^{r'_2}\mathtt{b}^{s'}$ with $r_1 + r_2 = r$ and $r'_1 + r'_2 = r'$, we get because of $r \neq r'$, $r_1 \neq -1$,

$$\begin{aligned} \alpha_1 &= \mathtt{a}^{r+2}\mathtt{b}^s \neq \mathtt{a}^{r'+2}\mathtt{b}^s = \beta_1,\\ \alpha_1 &= \mathtt{a}^{r+2}\mathtt{b}^s \neq \mathtt{a}^{r'_1}\mathtt{ba}^{r'_2+1}\mathtt{b}^{s'} = \beta_2, \text{ and} \end{aligned}$$

$$\alpha_2 = \mathtt{a}^{r_1}\mathtt{ba}^{r_2+1}\mathtt{b}^{s} \neq \mathtt{a}^{r'_1}\mathtt{ba}^{r'_2+1}\mathtt{b}^{s'} = \beta_2.$$

If $u_1 = \mathtt{a}^{r_1}$, $u_2u_3 = \mathtt{a}^{r_2}\mathtt{b}^{s}$ and $u'_1 = \mathtt{a}^{r'}\mathtt{b}^{s'_1}$, $u'_2u'_3 = \mathtt{b}^{s'_2}$ with $r_1 + r_2 = r$, $s'_1 + s'_2 = s'$, and $s'_1 \neq 0$ ($s'_1 = 0$ was treated in the previous case) we get because of $s'_1 \neq 0$,

$$\begin{aligned}
\alpha_1 = \mathtt{a}^{r+2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{aab}^{s'_2} = \beta_1,\\
\alpha_1 = \mathtt{a}^{r+2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{bab}^{s'_2} = \beta_2,\\
\alpha_2 = \mathtt{a}^{r_1}\mathtt{ba}^{r_2+1}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{bab}^{s'_2} = \beta_2.
\end{aligned}$$

If $u_1 = \mathtt{a}^{r}\mathtt{b}^{s_1}$, $u_2u_3 = \mathtt{b}^{s_2}$ and $u'_1 = \mathtt{a}^{r'}\mathtt{b}^{s'_1}$, $u'_2u'_3 = \mathtt{b}^{s'_2}$ with $r_1 + r_2 = r$, $s'_1 + s'_2 = s'$, and $s_1, s'_1 \neq 0$ ($s_1, s'_1 = 0$ were treated in the previous case) we get because of $r' \neq r$ and $s_1, s'_1 \neq 0$,

$$\begin{aligned}
\alpha_1 = \mathtt{a}^{r}\mathtt{b}^{s_1}\mathtt{aab}^{s_2} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{aab}^{s'_2} = \beta_1,\\
\alpha_1 = \mathtt{a}^{r}\mathtt{b}^{s_1}\mathtt{aab}^{s_2} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{bab}^{s'_2} = \beta_2,\\
\alpha_2 = \mathtt{a}^{r}\mathtt{b}^{s_1}\mathtt{bab}^{s_2} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{bab}^{s'_2} = \beta_2.
\end{aligned}$$

Consequently, α_1 and α_2 are all different and we get $2(k-1)$ many different scattered factors. Additionally, assume now that $|r - r'| = 3$. If $u_1 = \mathtt{a}^{r_1}$, $u_2u_3 = \mathtt{a}^{r_2}\mathtt{b}^{s}$ and $u'_1 = \mathtt{a}^{r'_1}$, $u'_2u'_3 = \mathtt{a}^{r'_2}\mathtt{b}^{s'}$ with $r_1 + r_2 = r$ and $r'_1 + r'_2 = r'$, we get because of $s'_1 \neq 0$, $r' \neq r$, $r' \neq r + 1$

$$\begin{aligned}
\alpha_1 = \mathtt{a}^{r+2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'_1}\mathtt{aba}^{r'_2}\mathtt{b}^{s'} = \beta_3,\\
\alpha_2 = \mathtt{a}^{r_1}\mathtt{ba}^{r_2+1}\mathtt{b}^{s} &\neq \mathtt{a}^{r'_1}\mathtt{aba}^{r'_2}\mathtt{b}^{s'} = \beta_3,\\
\alpha_3 = \mathtt{a}^{r_1}\mathtt{aba}^{r_2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'_1}\mathtt{aba}^{r'_2}\mathtt{b}^{s'} = \beta_3.
\end{aligned}$$

If $u_1 = \mathtt{a}^{r_1}$, $u_2u_3 = \mathtt{a}^{r_2}\mathtt{b}^{s}$ and $u'_1 = \mathtt{a}^{r'}\mathtt{b}^{s'_1}$, $u'_2u'_3 = \mathtt{b}^{s'_2}$ with $r_1 + r_2 = r$, $s'_1 + s'_2 = s'$, and $s'_1 \neq 0$ ($s'_1 = 0$ was treated in the previous case) we get because of $s'_1 \neq 0$, $r' \neq r + 2$,

$$\begin{aligned}
\alpha_1 = \mathtt{a}^{r+2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{abb}^{s'_2} = \beta_3,\\
\alpha_2 = \mathtt{a}^{r_1}\mathtt{ba}^{r_2+1}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{abb}^{s'_2} = \beta_3,\\
\alpha_3 = \mathtt{a}^{r_1}\mathtt{aba}^{r_2}\mathtt{b}^{s} &\neq \mathtt{a}^{r'}\mathtt{b}^{s'_1}\mathtt{abb}^{s'_2} = \beta_3.
\end{aligned}$$

If $u_1 = \mathtt{a}^r\mathtt{b}^{s_1}$, $u_2u_3 = \mathtt{b}^{s_2}$ and $u_1' = \mathtt{a}^{r'}\mathtt{b}^{s_1'}$, $u_2'u_3' = \mathtt{b}^{s_2'}$ with $r_1 + r_2 = r$, $s_1' + s_2' = s'$, and $s_1, s_1' \neq 0$ ($s_1, s_1' = 0$ were treated in the previous case) we get because of $r' \neq r$ and $s_1, s_1' \neq 0$, $r' \neq r + 2$,

$$\alpha_1 = \mathtt{a}^r\mathtt{b}^{s_1}\mathtt{aab}^{s_2} \neq \mathtt{a}^{r'}\mathtt{b}^{s_1'}\mathtt{abb}^{s_2'} = \beta_3,$$

$$\alpha_2 = \mathtt{a}^r\mathtt{b}^{s_1}\mathtt{bab}^{s_2} \neq \mathtt{a}^{r'}\mathtt{b}^{s_1'}\mathtt{abb}^{s_2'} = \beta_3,$$

$$\alpha_3 = \mathtt{a}^r\mathtt{b}^{s_1}\mathtt{abb}^{s_2} \neq \mathtt{a}^{r'}\mathtt{b}^{s_1'}\mathtt{abb}^{s_2'} = \beta_3.$$

Consequently, we have another $\lfloor \frac{k-2}{3} \rfloor + 1$ different scattered factors. This sums up to $|\operatorname{ScatFact}_k(w)| \geqslant \frac{7k-8}{3} > k + 1$. An immediate result is that the k-spectrum has at least $k + 3$ elements for $k \geqslant 5$. For $k = 3$ and $k = 4$ the results can be easily verified by testing.

case 2: $xz_1z_2 \neq \mathtt{a}^{k-2}\mathtt{b}^{k-2}$

In this case all words of the form $\mathtt{a}^r\mathtt{abaa}^s$ for $r + s = k - 3$, $r \in [|x|_\mathtt{a}]_0$, and $s \in [|y|_\mathtt{a}]_0$ are $|x|_\mathtt{a} + 1$ different scattered factors of length k of w. Analogously all $\mathtt{b}^{r'}\mathtt{abab}^{s'}$ with $r' + s' = k - 3$, $r' \in [|x|_\mathtt{b}]_0$, $s' \in [|y|_\mathtt{b}]_0$ are $|x|_\mathtt{b} + 1$ different scattered factors of length k of w. All these factors are different and, additionally, w has $\mathtt{a}^k$ and $\mathtt{b}^k$ as scattered factors. Hence, $|\operatorname{ScatFact}_k(w)| \geqslant |x|_\mathtt{a} + |x|_\mathtt{b} + 4 = |x| + 4$ holds. Since the length of w is $2k$, the length of xy is $2k - 3$ and, consequently, x and y have different lengths. Assume w.l.o.g. $|x| > |y|$, i.e., $|x| \geqslant k - 1$. This implies $|\operatorname{ScatFact}_k(w)| \geqslant k + 3$ follows. This proves the claim for $c = 0$.

Assume now $c > 0$ and let $w = \mathtt{a}^k\mathtt{b}^{k-c}$. By the previous part, we know $|\operatorname{ScatFact}_{k-c}(u)| = k - c + 1$ if and only if $u = \mathtt{a}^{k-c}\mathtt{b}^{k-c}$. The claim about the $(k - c)$-spectrum follows immediately by $\operatorname{ScatFact}_{k-c}(w) = \operatorname{ScatFact}_{k-c}(\mathtt{a}^k\mathtt{b}^{k-c})$ since the prepended letters $\mathtt{a}$ do not change the $(k - c)$-spectrum. For $i \in [c - 1]_0$ notice that $x \in \operatorname{ScatFact}_{k-i}(\mathtt{a}^k\mathtt{b}^{k-c})$ implies that $\mathtt{a}x$ (resp. $x\mathtt{b}$, $x\mathtt{a}$, $\mathtt{b}x$) is a scattered factor of $\mathtt{a}^k\mathtt{b}^{k-c}$ of length $k - i + 1$. Thus, $|\operatorname{ScatFact}_{k-i+1}(w)| \geqslant k - c + 1$ follows. On the other hand a scattered factor of $\mathtt{a}^k\mathtt{b}^{k-c}$ of length $k - i + 1$ is exactly of this form, since it can neither start with $\mathtt{b}$ ($\mathtt{a}^k\mathtt{b}^{k-c}$ has only $(k - c)$ occurrences of $\mathtt{b}$) nor contain $\mathtt{ba}$ resp. $\mathtt{ab}$ (this would be the implication of a scattered factor being of the form $\mathtt{a}x'$ with $|x'| = k - i$, $x' \notin \operatorname{ScatFact}_{k-i}(\mathtt{a}^k\mathtt{b}^{k-c})$). □

3.1.2 Remark. Theorem 3.1.1 answers immediately the question, whether a given set $S \subseteq \Sigma^k$, with $|S| < k + 1$ or $|S| = k + 2$, is a k-spectrum of a weakly-0-balanced word w in the negative.

Theorem 3.1.1 shows that the smallest cardinality of the k-spectrum of a word w is reached when the letters in w are *nicely ordered*, both for weakly-0-balanced words as well as for weakly-c-balanced words with $c > 0$. The largest cardinality is, not surprisingly, reached for words where the alternation of $\mathtt{a}$ and $\mathtt{b}$ letters is, in a sense, maximal, e.g., for $w = (\mathtt{ab})^k$. To this end, one can show a general result.

3.1.3 Theorem. *For $w \in \Sigma^*$, the k-spectrum of w is Σ^k if and only if*

$$\{\mathtt{ab}, \mathtt{ba}\}^k \cap \mathrm{ScatFact}_{2k}(w) \neq \emptyset.$$

Proof. We will show this result by induction. For $k = 1$, the equivalence is:

$$\mathrm{ScatFact}_1(w) = \Sigma \text{ iff } \{\mathtt{ab}, \mathtt{ba}\} \cap \mathrm{ScatFact}_2(w) \neq \emptyset.$$

If both $\mathtt{a}$ and $\mathtt{b}$ are scattered factors of w, $\mathtt{ab}$ or $\mathtt{ba}$ has to be a factor and, thus, a scattered factor of w. On the other hand if w has $\mathtt{ab}$ or $\mathtt{ba}$ as a scattered factor, it has $\mathtt{a}$ and $\mathtt{b}$ as scattered factors.

Assume now that the equivalence holds for an arbitrary but fixed $k - 1 \in \mathbb{N}$. We will show that it holds for k.

For the $\Leftarrow$-direction consider $u \in \{\mathtt{ab}, \mathtt{ba}\}^k \cap \mathrm{ScatFact}_{2k}(u)$. Thus, $u \in \{\mathtt{ab}, \mathtt{ba}\}^{k-1}\{\mathtt{ab}, \mathtt{ba}\}$ and, hence, there exists $u' \in \{\mathtt{ab}, \mathtt{ba}\}^{k-1}$ with $u \in u'\{\mathtt{ab}, \mathtt{ba}\}$. By induction hypothesis, we have $\mathrm{ScatFact}_{k-1}(u') = \Sigma^{k-1}$. For any $x \in \Sigma^k$ there exists $x' \in \Sigma^{k-1}$ with $x \in x'\{\mathtt{a}, \mathtt{b}\}$. This implies that there exist $a_0, \ldots, a_{k-1} \in \Sigma^*$ with $u' = a_0 x'[1] a_1 \cdots x'[k-1] a_{k-1}$ since $x' \in \mathrm{ScatFact}_{k-1}(u')$. By

$$u \in a_0 x'[1] a_1 \cdots x'[k-1] a_{k-1} \{\mathtt{ab}, \mathtt{ba}\},$$

it follows in both cases, namely $x = x'\mathtt{a}$ or $x = x'\mathtt{b}$, that $x \in \mathrm{ScatFact}_k(w)$. This proves the inclusion $\Sigma^k \subseteq \mathrm{ScatFact}_k(w)$. By $\mathrm{ScatFact}_k(w) \subseteq \Sigma^k$, the first direction is proven.

For the $\Rightarrow$-direction assume $\mathrm{ScatFact}_k(w) = \Sigma^k$. Assume w.l.o.g. that $w[|w|] = \mathtt{a}$. Choose $x, y \in \Sigma^*$ with $w = xy$ and $x[|x|] = \mathtt{b}$, and $y \in \mathtt{a}^*$. As $\Sigma^{k-1}\mathtt{b} \subset \Sigma^k$, it follows that $\Sigma^{k-1}b \subseteq \mathrm{ScatFact}_k(x)$. Clearly, this means that $\Sigma^{k-1} \subseteq \mathrm{ScatFact}_{k-1}(x[1..|x|-1])$. By the induction hypothesis, we get that $\{\mathtt{ab}, \mathtt{ba}\}^{k-1} \cap \mathrm{ScatFact}_{2(k-1)}(x[1..|x|-1]) \neq \emptyset$. Thus, $\{\mathtt{ab}, \mathtt{ba}\}^{k-1} x[|x|]\mathtt{a} \cap$

$\mathrm{ScatFact}_{2k}(w[1..|x|+1]) \neq \emptyset$, because $w[1..|x|+1] = x[1..|x|]\mathtt{b}$. Hence, $\{\mathtt{ab}, \mathtt{ba}\}^{k-1}\mathtt{ba} \cap \mathrm{ScatFact}_{2k}(w) \neq \emptyset$. The conclusion follows. □

The previous theorem has an immediate consequence that characterises exactly the weakly-0-balanced words of length $2k$ for which the maximal cardinality of $\mathrm{ScatFact}_k(w)$ is reached.

3.1.5 Proposition. *For $k \in \mathbb{N}_{\geqslant 3}$ and $w \in \Sigma^{2k}_{\mathtt{wzb}}$ we have $w \in \{\mathtt{ab}, \mathtt{ba}\}^k$ if and only if* $\mathrm{ScatFact}_k(w) = \Sigma^k$*.*

Proof. If $w \in \{\mathtt{ab}, \mathtt{ba}\}^k$, then $\{\mathtt{ab}, \mathtt{ba}\}^k \cap \mathrm{ScatFact}_{2k}(w) \neq \emptyset$ and the claim follows by Theorem 3.1.3. On the other hand if $\mathrm{ScatFact}_k(w) = \Sigma^k$ then $\{\mathtt{ab}, \mathtt{ba}\}^k \cap \mathrm{ScatFact}_k(w) \neq \emptyset$ and since $|w| = 2k$ we get $w \in \{\mathtt{ab}, \mathtt{ba}\}^k$. □

To see why, from $w \in \{\mathtt{ab}, \mathtt{ba}\}^k$, it follows that $\mathrm{ScatFact}_k(w) = \Sigma^k$, note that, by definition, a word $w \in \{\mathtt{ab}, \mathtt{ba}\}^k$ is just a concatenation of k blocks from $\{\mathtt{ab}, \mathtt{ba}\}$. To construct the scattered factors of w, we can simply select from each block either the $\mathtt{a}$ or the $\mathtt{b}$. The resulting output is a word of length k, where in each position we could choose freely the letter. Consequently, we can produce all words in Σ^k in this way. The other implication follows by induction.

Generalising Proposition 3.1.5 for weakly-c-balanced words requires a more sophisticated approach. A generalisation would be to consider $w \in \{\mathtt{ab}, \mathtt{ba}\}^{k-c}\mathtt{a}^c$. By Theorem 3.1.3, we have $\mathrm{ScatFact}_{k-c}(w) = \Sigma^{k-c}$. But the size of $\mathrm{ScatFact}_{k-i}(w)$ for $i \in [c]_0$ depends on the specific choice of w. To see why, consider the words $w_1 = \mathtt{baabba}$ and $w_2 = (\mathtt{ba})^3$. Then by Proposition 3.1.5, $|\mathrm{ScatFact}_3(w_1)| = 8 = |\mathrm{ScatFact}_3(w_2)|$. However, when we append an $\mathtt{a}$ to the end of both w_1 and w_2, we see that in fact $|\mathrm{ScatFact}_4(w_1\mathtt{a})| = 11 \neq 12 = |\mathrm{ScatFact}_4(w_2\mathtt{a})|$. The main difference between weakly-0-balanced and weakly-c-balanced words for $c > 0$, regarding the maximum cardinality of the scattered factors-sets, comes from the role played by the factors $\mathtt{a}^2$ and $\mathtt{b}^2$ occurring in w.

In the remaining part of this section we present a series of results for weakly-c-balanced words. Intuitively, the words with many alternations between $\mathtt{a}$ and $\mathtt{b}$ have more distinct scattered factors. So, we will focus on such words mainly. Our first result is a direct consequence of Theorem 3.1.3.

The second result concerns words in which $\mathtt{a}^2$ and $\mathtt{b}^2$ do not occur. Here, we give a method to identify efficiently the ℓ-spectra of words which are prefixes of $(\mathtt{ab})^{\omega}$, for all ℓ. Finally, we are able to derive a way to efficiently enumerate (and count) the scattered factors of length k of $(\mathtt{ab})^{k-c}\mathtt{a}^c$.

3.1.6 Corollary. *For $k \in \mathbb{N}_{\geqslant 3}$, $c \in [k]_0$, and $w \in \Sigma^{2k-c}$ weakly-c-balanced, the cardinality of* $\text{ScatFact}_{k-c}(w)$ *is exactly* 2^{k-c} *if and only if* $\text{ScatFact}_{2(k-c)}(w) \cap \{\mathtt{ab}, \mathtt{ba}\}^{k-c} \neq \emptyset$.

Proof. The claim follows directly by Theorem 3.1.3. □

As announced, we further focus our investigation on the words $w = (\mathtt{ab})^{k-c}\mathtt{a}^c$. By Theorem 3.1.3, we have $|\,\text{ScatFact}_i(w)| = \Sigma^i$ for all $i \in [k-c]_0$. For all i with $k - c < i \leqslant k$, a more sophisticated counting argument is needed. Intuitively, a scattered factor of length i of $(\mathtt{ab})^{k-c}\mathtt{a}^c$ consists of a part that is a scattered factor (of arbitrary length) of $(\mathtt{ab})^{k-c}$ followed by a (possibly empty) suffix of as. Thus, a full description of the ℓ-spectra of words that occur as prefixes of $(\mathtt{ab})^{\omega}$, for all appropriate ℓ, is useful. To this end, we introduce the notion of a deleting sequence: for a word w and a scattered factor u of w the deleting sequence contains (in a strictly increasing order) w's positions that have to be deleted to obtain u.

3.1.7 Definition. For $w \in \Sigma^*$, $\sigma = (s_1, \ldots, s_\ell) \in [|w|]^\ell$, with $\ell \leqslant |w|$ and $s_i < s_{i+1}$ for all $i \in [\ell - 1]$, is a *deleting sequence*. The scattered factor u_σ associated to a deleting sequence σ is $u_\sigma = u_1 \cdots u_{\ell+1}$, where $u_1 = w[1..s_1 - 1]$, $u_{\ell+1} = w[s_\ell + 1..|w|]$, and $u_i = w[s_{i-1} + 1..s_i - 1]$ for $2 \leqslant i \leqslant \ell$. Two sequences σ, σ' with $u_\sigma = u_{\sigma'}$ are called *equivalent*.

For the word $w = \mathtt{abbaa}$ and $\sigma = (1, 3, 4)$ the associated scattered factor is $u_\sigma = \mathtt{ba}$. Since $\mathtt{ba}$ can also be generated by $(1, 3, 5)$, $(1, 2, 4)$ and $(1, 2, 5)$, these sequences are equivalent.

In order to determine the ℓ-spectrum of a word $w \in \Sigma^n$ for $\ell, n \in \mathbb{N}$, we can determine how many equivalence classes the equivalence defined above does have, for sequences of length $k = n - \ell$. The following three lemmata characterise the equivalence of deleting sequences.

3.1.8 Lemma. *Let $w \in \Sigma^n$ be a prefix of $(\mathtt{ab})^{\omega}$. Let $\sigma = (s_1, \ldots, s_k)$ be a deleting sequence for w such that there exists $j \geqslant 2$ with $s_{j-1} < s_j - 1$ and $s_j + 1 = s_{j+1}$.*

Then σ is equivalent to $\sigma' = (s_1, \ldots, s_{j-1}, s_j - 1, s_{j+1} - 1, s_{j+2}, \ldots, s_k)$, i.e., σ' is the sequence σ where both s_j and s_{j+1} were decreased by 1.

Proof. Since $s_{j-1} < s_j - 1$, the factor u_σ contains the letter $w[s_j - 1]$. If $w[s_j] = \mathtt{a}$ then $w[s_{j+1}] = w[s_j + 1] = \mathtt{b}$ and $w[s_j - 1] = \mathtt{b}$. Clearly, when deleting $w[s_j - 1]$ and $w[s_j]$ according to the sequence σ', the $\mathtt{b}$ that was corresponding to $w[s_j - 1]$ will be replaced by a letter $\mathtt{b}$ corresponding to $w[s_{j+1}]$, which is not deleted. So, in the end, $u_{\sigma'} = u_\sigma$. The case $w[s_j] = \mathtt{b}$ is analogous. □

3.1.9 Lemma. *Let $w \in \Sigma^n$ be a prefix of $(\mathtt{ab})^\omega$. Let $\sigma = (s_1, \ldots, s_k)$ be a deleting sequence for w. Then there exists an integer $j \geqslant 0$ such that σ is equivalent to the deleting sequence $(1, 2, \ldots, j, s'_{j+1}, \ldots, s'_k)$, where $s'_{j+1} > j + 1$ and $s'_i > s'_{i-1} + 1$, for all $j < i \leqslant k$. Moreover, $j \geqslant 1$ if and only if σ contains two consecutive positions or σ started with* 1.

Proof. Let $\sigma_0 = \sigma$. For $i \geqslant 0$, we iteratively transform σ_i into σ_{i+1} as follows: if σ_i contains on consecutive positions the numbers $g, t, t+1, h$, such that $g < t - 1$ and $h > t + 2$, we replace them by $g, t-1, t, h$ and obtain the sequence σ_{i+1}. By Lemma 3.1.8, σ_i is equivalent to σ_{i+1}. It is clear that in $O(n^2)$ steps we will reach a sequence σ_ℓ which cannot be transformed anymore. We take $\sigma' = \sigma_\ell$ and it is immediate that it will have the required form. □

3.1.10 Lemma. *Let $w \in \Sigma^n$ be a prefix of $(\mathtt{ab})^\omega$. Let $\sigma_1 = (1, 2, \ldots, j_1, s'_{j_1+1}, \ldots, s'_k)$, where $s'_{j_1+1} > j_1 + 1$ and $s'_i > s'_{i-1} + 1$, for all $j_1 < i \leqslant k$, and $\sigma_2 = (1, 2, \ldots, j_2, s''_{j_2+1}, \ldots, s''_k)$, where $s''_{j_2+1} > j_2 + 1$ and $s''_i > s''_{i-1} + 1$, for all $j_2 < i \leqslant k$. If $\sigma_1 \neq \sigma_2$ then σ_1 and σ_2 are not equivalent (i.e., $u_{\sigma_1} \neq u_{\sigma_2}$).*

Proof. First, we consider the case $j_1 = j_2$. Let ℓ be minimal such that $s'_\ell \neq s''_\ell$. We can assume without losing generality that $s'_\ell < s''_\ell$. Then u_{σ_1} and u_{σ_2} share the same prefix of length $t = (s'_\ell - 1) - (\ell - 1)$. This prefix ends with $w[s'_\ell - 1]$ and is followed by $w[s'_\ell + 1]$ in u_{σ_1} and, respectively, by $w[s'_\ell]$ in u_{σ_2}. But $w[s'_\ell + 1] \neq w[s'_\ell]$, so $u_{\sigma_1} \neq u_{\sigma_2}$.

Further, we consider the case when $j_1 < j_2$ (the case $j_2 < j_1$ is symmetric); assume, as a convention, that $s''_{k+1} = 0$ and let $d = j_2 - j_1$. Clearly, j_1 and j_2 must have the same parity, or u_{σ_1} and u_{σ_2} would start with different letters, so they would not be equal. Let ℓ be minimal integer such that $s'_\ell - j_1 \neq s''_{\ell+d} - j_2$; because $s''_{k+1} = 0$ by convention, we have $\ell \leqslant k$. If both ℓ and $\ell + d$ are at most k, then we get similarly to the case $j_1 = j_2$ that $u_{\sigma_1} \neq u_{\sigma_2}$. In the case when $\ell \leqslant k < \ell + d$, then, by length reasons, all positions $j > s_\ell$ (so, including $s_\ell + 1$) in w should belong to σ_1, a contradiction. This concludes our proof. □

The previous lemmata allow us to determine the cardinality of the ℓ-spectrum of a prefix of $(\mathtt{ab})^\omega$.

3.1.11 Theorem. *Let w be a word of length n which is a prefix of $(\mathtt{ab})^\omega$. Then $|\operatorname{ScatFact}_\ell(w)| = \sum_{j\in[n-\ell]_0} \binom{\ell}{n-\ell-j}$.*

Proof. Lemmata 3.1.8, 3.1.9, and 3.1.10 show that the representatives of the equivalence classes w.r.t. the equivalence relation between deleting sequences, introduced in Definition 3.1.7, are the sequences $(1, 2, \ldots, j, s'_{j+1}, \ldots, s'_k)$, where $s'_{j+1} > j + 1$ and $s'_i > s'_{i-1} + 1$, for all $j < i \leqslant k$. For a fixed $j \geqslant 1$, the number of such sequences is $\binom{(n-j-1)-(k-j)+1}{k-j} = \binom{n-k}{k-j}$. For $j = 0$, we have $\binom{(n-1)-k+1}{k} = \binom{n-k}{k}$ nonequivalent sequences (note that none starts with 1, as those were counted for $j = 1$ already). In total, we have, for a word w of length n, which is a prefix of $(\mathtt{ab})^\omega$, exactly $\sum_{j\in[k]_0} \binom{n-k}{k-j}$ nonequivalent deleting sequences of length k, so $\sum_{j\in[k]_0} \binom{n-k}{k-j}$ different scattered factors of length $n - k$. In the above formula, we assume that $\binom{\mathtt{a}}{\mathtt{b}} = 0$ when $\mathtt{a} < \mathtt{b}$.

Moreover, the distinct scattered factors of length $\ell = n - k$ of w can be obtained efficiently as follows. For j from 0 to ℓ, delete the first j letters of w. For all choices of $\ell - j$ positions in $w[j + 1..n]$, such that each two of these positions are not consecutive, delete the letters on the respective positions. The resulted word is a member of $\operatorname{ScatFact}_\ell(w)$, and we never obtain the same word twice by this procedure. □

A straightforward consequence of the above theorem is that, if $\ell \leqslant n - \ell$ then $|\operatorname{ScatFact}_\ell(w)| = 2^\ell$. With Theorem 3.1.11, we can now completely characterise the cardinality of the ℓ-spectra of the weakly-c-balanced word $(\mathtt{ab})^{k-c}\mathtt{a}^c$ for $\ell \leqslant k$.

3.1.12 Theorem. *Let* $w = (\mathtt{ab})^{k-c}\mathtt{a}^c$ *for* $k \in \mathbb{N}$, $c \in [k]_0$. *Then, for* $i \leqslant k - c$ *we have* $|\operatorname{ScatFact}_i(w)| = 2^i$. *For* $k \geqslant i > k - c$ *we have* $|\operatorname{ScatFact}_i(w)| = 1 + 2^{k-c} + \sum_{j \in [(i+c)-k-1]_0} |\operatorname{ScatFact}_{i-j-1}((\mathtt{ab})^{k-c-1}\mathtt{a})|$.

Proof. We will need to show the proof for $k \geqslant i > k - c$, as the other part follows immediately from Theorem 3.1.3.

We give a method to count the scattered factors of $w = (\mathtt{ab})^{k-c}\mathtt{a}^c$. To begin with, we have the scattered factor $\mathtt{a}^i$. All the other scattered factors must contain a letter $\mathtt{b}$. Thus, we count separately the scattered factors of the form $u\mathtt{ba}^j$, for each $j \in [i-1]_0$. This is equivalent to counting in how many ways we can choose u. For each such u we will just have to append $\mathtt{ba}^j$ at the end to get the desired scattered factors. Thus, $|u| = i - j - 1$. If $j \geqslant c$ then u should occur as a scattered factor of $(\mathtt{ab})^{k-j-1}\mathtt{a}$ (in order to be able to append $\mathtt{ba}^j$ at its end and still stay as a scattered factor of w), while if $j < c$ then u should occur as a scattered factor of $(\mathtt{ab})^{k-c-1}\mathtt{a}$. In the first case, the length of the scattered factor u we want to generate is less than half of the length of the word $(\mathtt{ab})^{\mathtt{a}}$ from which we generate it. So, there are 2^{i-j-1} choices for u. In the second case, if $j \geqslant (i + c) - k$, again, the length of the scattered factor u we want to generate is less than half of the length of the word $(\mathtt{ab})^{k-c-1}\mathtt{a}$ from which we generate it. So, there are 2^{i-j-1} choices for u again. Finally, if $j < (i + c) - k$, then there holds $i - j - 1 > k - c - 1$, and we need Theorem 3.1.3 to generate u. There are $|\operatorname{ScatFact}_{i-j-1}((\mathtt{ab})^{k-c-1}\mathtt{a})|$ ways to choose u in this case. Summing all these up, we get the result from the statement:

$$\begin{aligned} 1 + \sum_{j=i+c-k}^{i-1} 2^{i-j-1} + \sum_{j \in [i+c-k-1]_0} & \operatorname{ScatFact}_{i-j-1}((\mathtt{ab})^{k-c-1}\mathtt{a}) \\ = 1 + 2^{k-c} + \sum_{j \in [i+c-k-1]_0} & \operatorname{ScatFact}_{i-j-1}((\mathtt{ab})^{k-c-1}\mathtt{a}). \end{aligned}$$

This concludes our proof. □

As in the case of the scattered factors of prefixes of $(\mathtt{ab})^{\omega}$, we have a precise and efficient way to generate the scattered factors of $w = (\mathtt{ab})^{k-c}\mathtt{a}^c$. For scattered factors of length $i \leqslant k - c$ of w, we just generate all possible words of length i. For greater i, on top of $\mathtt{a}^i$, we generate separately the scattered factors of the form $u\mathtt{ba}^j$, for each $j \in [i-1]_0$. It is clear that, in such a word, $|u| = i - j - 1$, and if $j \geqslant c$ then u must be a scattered factor of $(\mathtt{ab})^{k-j-1}\mathtt{a}$, while if $j < c$ then u must be a scattered factor of $(\mathtt{ab})^{k-c-1}\mathtt{a}$. If $j \geqslant (i+c) - k$ then, by Theorem 3.1.11, u can take all 2^{i-j-1} possible values. For smaller values of j, we need to generate u of length $i - j - 1$ as a scattered factor of $(\mathtt{ab})^{k-c-1}\mathtt{a}$, by the method described after Proposition 3.1.5.

Nevertheless, Theorems 3.1.11 and 3.1.12 are useful to see that in order to determine the cardinality of the sets of scattered factors of words consisting of alternating $\mathtt{a}$s and $\mathtt{b}$s or, respectively, of $(\mathtt{ab})^{k-c}\mathtt{a}^c$, it is not needed to generate these sets effectively.

So far, a characterisation for the smallest and the largest k-spectra of words of a given length were presented (Theorem 3.1.1 and Proposition 3.1.5). Now the part in between will be investigated for weakly-0-balanced words (i.e., words of length $2k$ with k occurrences of each letter). As before, we shall assume that $k \in \mathbb{N}_{\geqslant 3}$. In the particular case that $k = 3$, we have already proven that the k-spectrum with minimal cardinality has 4 elements and that the maximal cardinality is 8. Moreover, as mentioned in Remark 3.1.2 a k-spectrum of cardinality 5 does not exist for weakly-0-balanced words of length $2k$. The question remains if k-spectra of cardinalities 6 and 7 exist, and if so, for which words.

Before showing that a k-spectrum of cardinality $2^k - 1$ for weakly-0-balanced words of length $2k$ also exists for all $k \in \mathbb{N}_{\geqslant 3}$, we prove that only scattered factors of the form $\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$ for $i \in [k-2]_0$ (up to renaming, reversal) can be "taken out" from the full set of possible scattered factors independently, without additionally requiring the removal of more scattered factors as well. In particular, if a word of length k of another form is absent from the set of scattered factors of w, then $|\operatorname{ScatFact}_k(w)| < 2^k - 1$ follows.

3.1.13 Lemma. *If for $w \in \Sigma^{2k}_{\mathrm{wzb}}$ there exists $u \notin \operatorname{ScatFact}_k(w)$ with $u \notin \{\mathtt{b}^i\mathtt{a}^{k-i} \mid i \in [k-1]\} \cup \{\mathtt{a}^i\mathtt{b}^{k-i} \mid i \in [k-1]\}$, then $|\operatorname{ScatFact}_k(w)| < 2^k - 1$.*

Proof. Let be $i \in [k-2]_0$. First, consider $u = \mathtt{b}^r\mathtt{a}^s$ for $r + s = k$ and $r \notin [i] \cup \{k-i, \dots, k\}$ and $\Sigma^k \backslash \{u\} \supset \operatorname{ScatFact}_k(w)$ for a word $w \in \Sigma^{2k}_{\mathrm{wzb}}$. If $\mathtt{b}^{r+1}\mathtt{a}^{s-1}$ is also not a scattered factor of w, the claim is proven (in this case two elements of Σ^k are missing in $\operatorname{ScatFact}_k(w)$). Assume $\mathtt{b}^{r+1}\mathtt{a}^{s-1} \in$

ScatFact(w). This implies that (possibly intertwined) $(s-1)$ occurrences of $\mathtt{a}$ follow $(r+1)$ occurrences of $\mathtt{b}$. Since u is not a scattered factor of w, after these $(s-1)$ as only bs may occur. If $\mathtt{b}^{r-1}\mathtt{a}^{s}\mathtt{b}$ is not a scattered factor, the claim is again proven and so suppose that it is one. This implies that the $(r-1)$ bs are preceded by as and not by bs. Hence, $\mathtt{b}^{r+1}\mathtt{a}^{s-1}$ is not a scattered factor which contradicts the assumption. Consider now $u = u_1\mathtt{b}^{r}\mathtt{a}^{s}\mathtt{b}^{t}u_2$ with $|u| = k$ not to be a scattered factor of w for $r, s, t \in \mathbb{N}$. Following the same arguments as before, the claim is proven if $u_1\mathtt{b}^{r-1}\mathtt{a}^{s}\mathtt{b}^{t+1}u_2$ is not a scattered factor and, hence, it is assumed to be one. This implies that exactly $|u_1|_{\mathtt{b}}$ bs occur before $\mathtt{b}^{r-1}$. This implies that $u_1\mathtt{b}^{r+1}\mathtt{a}^{s}\mathtt{b}^{t-1}u_2$ is not a scattered factor of w of length k. Analogously it can be proven that scattered factors containing the switch from $\mathtt{a}$ to $\mathtt{b}$ and back to $\mathtt{a}$ cannot lead to the cardinality $2^k - 1$. □

3.1.14 Proposition. *For $k \in \mathbb{N}_{\geqslant 3}$ and $w \in \Sigma^{2k}_{\mathrm{wzb}}$, the set* $\mathrm{ScatFact}_k(w)$ *has* $2^k - 1$ *elements if and only if* $w \in \{(\mathtt{ab})^i\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2} \mid i \in [k-2]_0\}$ *(up to renaming and reversal). In particular* $\mathrm{ScatFact}_k(w) = \Sigma^k \backslash \{\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}\}$ *holds for* $w = (\mathtt{ab})^i\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$ *with* $i \in [k-2]_0$.

Proof. Let be $i \in [k-2]_0$. First, the $\Leftarrow$-direction will be proven and for that consider $w = (\mathtt{ab})^i\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$. By Lemma 3.1.5, it follows

$$\mathrm{ScatFact}_i((\mathtt{ab})^i) = \Sigma^i \text{ and } \mathrm{ScatFact}_{k-i-2}((\mathtt{ab})^{k-i-2}) = \Sigma^{k-i-2}.$$

With $\mathrm{ScatFact}_2(\mathtt{a}^2\mathtt{b}^2) = \{\mathtt{aa}, \mathtt{ab}, \mathtt{bb}\}$ the k-spectrum of w has at least $3 \cdot 2^i \cdot 2^{k-i-2} = 3 \cdot 2^{k-2} = 2^k - 2^{k-2}$ elements. Notice that by this construction, scattered factors with a $\mathtt{ba}$ at the middle position cannot be reached. For this reason we have to have a look at w's remaining scattered factors not being gained by the above construction. This means that not only i letters are allowed to be taken of the first part and not only $k-i-2$ letters from the last part.

Having a deeper look into $(\mathtt{ab})^i$ one can notice that all *binary numbers* (encoded by $\mathtt{a}, \mathtt{b}$) of length i are scattered factors of $(\mathtt{ab})^{i-1}\mathtt{a}$. Appending to these scattered factors a $\mathtt{b}$ implies that nearly all *binary numbers* are in the $i+1$-spectrum of $\mathtt{ab}^i$. Appending now an $\mathtt{a}$ from the middle part and then each of the words from the last part leads to nearly all remaining scattered factors of the k-spectrum of w. The only missing word is $\mathtt{b}^{i+i}$, since the last $\mathtt{b}$ cannot be reached within the first part. This implies that the word $\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$ is not in the k-spectrum of w since with the (i+1)th $\mathtt{b}$ the middle part is reached and the last part contains only $k-i-2$ $\mathtt{a}$s. This concludes $|\operatorname{ScatFact}_k(w)| = 2^k - 1$.

On the other hand if $|\operatorname{ScatFact}_k(w)| = 2^k - 1$ an element of the form $\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$ for an $i \in [k-2]_0$ is missing in the k-spectrum of w. Moreover, this is exactly the only element missing. Fix an $i \in [k-2]_0$ and set $u = \mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$. The proof will be very technical and exclude step by step all other possibilities than w being $(\mathtt{ab})^i\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$. First, consider $i = k-2$. This implies $u = \mathtt{b}^{k-1}\mathtt{a}$. In this case w has to end in $\mathtt{b}^2$ but not in $\mathtt{b}^3$ since otherwise $\mathtt{b}^{k-2}\mathtt{a}^2$ would not be a scattered factor. If w were of the form $w_1\mathtt{bab}^2$, $|w_1|_a = k-1$ and $|w_1|_b = k-3$ would hold which would imply that $\mathtt{b}^{k-2}\mathtt{a}^2$ is not a scattered factor. If w ended in $\mathtt{a}^3\mathtt{b}^2$, $\mathtt{a}^{k-2}\mathtt{ba}$ would be excluded. Hence, w ends in $\mathtt{a}^2\mathtt{b}^2$. Suppose at last that $w = (\mathtt{ab})^\ell\mathtt{a}^2\mathtt{b}^2 w_2$ holds for $\ell < k-2$ and $w_2 \in \Sigma^*$. Then w_2 has each $(k-\ell-2)$ occurrences of $\mathtt{a}$ and $\mathtt{b}$. Thus, $\mathtt{b}^{\ell+1}\mathtt{a}^{k-\ell-1}$ is not a scattered factor of length k. This proves that for $i = k-2$, $w = (\mathtt{ab})^{k-2}\mathtt{a}^2\mathtt{b}^2$ is implied by $\mathtt{b}^{k-2}\mathtt{a}^1$ being the only excluded scattered factor from Σ^k. Hence, assume $i \in [k-3]_0$.

`Supposition:` w ends in $\mathtt{b}^\ell$ for $\ell \geqslant 2$

If $i < k-2$ holds, then $\mathtt{b}^{k-1}\mathtt{a} \notin \operatorname{ScatFact}_k(w)$ follows and since $i+1 < k-1$ holds, this element is different from u.

In the next step it will be shown that exactly $k-i-2$ repetitions of $\mathtt{ab}$ are a suffix of w.

`Supposition:` $w = w_1\mathtt{b}^2(\mathtt{ab})^\ell$

If $\ell > k-i-2$ held, $\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$ would not be a scattered factor of w. If $\ell < k-i-2$ held, $\mathtt{b}^{k-\ell-1}\mathtt{a}^{\ell+1}$ would not be a scattered factor since w_1 has $(k-1)$ occurrences of $\mathtt{a}$ and $(k-\ell-2)$ occurrences of $\mathtt{b}$.

`Supposition:` $w = w_1\mathtt{a}^2(\mathtt{ba})^\ell b$

In this case $|w_1|_\mathtt{a} = k-2-\ell$ and $|w_1|_\mathtt{b} = k-\ell-1$ holds. This implies that $\mathtt{a}^{k-2-\ell}\mathtt{b}^{\ell+1}\mathtt{a}$ is not in the k-spectrum of w.

Consequently, there exists a w_1 such that $w = w_1\mathtt{b}^2(\mathtt{ab})^{k-i-2}$ holds. In the next step it will be shown that $\mathtt{b}^2$ has to be preceded by $\mathtt{a}^2$.

`Supposition:` $w = w_1\mathtt{b}^3(\mathtt{ab})^{k-i-2}$

Here w_1 has $(i+2)$ $\mathtt{a}$ and $(i-1)$ $\mathtt{b}$ and, hence, $\mathtt{b}^i\mathtt{a}^{k-i-2}\mathtt{b}^2$ is not a scattered factor of length k of w.

`Supposition:` $w = w_1\mathtt{bab}^2(\mathtt{ab})^{k-i-2}$

This implies $\mathtt{a}^{i+2}\mathtt{bab}^{k-i} \notin \operatorname{ScatFact}_k(w)$ since w_1 has $i+1$ occurrences of $\mathtt{a}$

and $i-1$ occurrences of $\mathtt{b}$.
This proves that $\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$ is a suffix of w. The case that this is preceded by another $\mathtt{a}$ is excluded since then $\mathtt{a}^i\mathtt{ba}^{k-i-1}$ would not be in the k-spectrum of k. In the last step it will be shown that the first occurrence of $\mathtt{a}^2$ is at the point 2ℓ.
`Supposition:` $w = (\mathtt{ab})^\ell \mathtt{a}^2 w_2$ for $\ell \neq i$
If ℓ is smaller than i, $|w_2|_\mathtt{a} = k-\ell-2$ and $|w_2|_\mathtt{b} = k-\ell$ hold and $\mathtt{b}^{\ell+1}\mathtt{a}^{k-\ell-1} \notin \mathrm{ScatFact}_k(w)$ follows. If ℓ is greater than i, in contradiction to the main assumption $\mathtt{b}^{i+1}\mathtt{a}^{k-i-1}$ is a scattered factor, because $\mathtt{b}^{i+1}$ is a scattered factor of $(\mathtt{ab})^\ell$ and $k-\ell+\ell-(i+1) = k-i-1$ $\mathtt{a}$ are left in the rest of w.
Combining $w = (\mathtt{ab})^i\mathtt{a}^2 w_2$ and $w = w_1\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$ the claim that w is of the form $(\mathtt{ab})^i\mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-i-2}$ is proven. □

By Proposition 3.1.14, we get that 7 is a possible cardinality of the set of scattered factors of length 3 of weakly-0-balanced words of length 6 and, moreover, that exactly the words $\mathtt{a}^2\mathtt{b}^2\mathtt{ab}$ and $\mathtt{aba}^2\mathtt{b}^2$ (and symmetric words obtained by reversal and renaming) have seven different scattered factors. The following theorem demonstrates that there always exists a weakly-0-balanced word w of length $2k$ such that $|\mathrm{ScatFact}_k(w)| = 2k$. Thus, for the case $k = 3$ also the question if six is a possible cardinality of $\mathrm{ScatFact}_3(w)$ can be answered positively.

3.1.15 Theorem. *The k-spectrum of a word $w \in \Sigma^{2k}_{\mathrm{wzb}}$ has exactly $2k$ elements if and only if $w \in \{\mathtt{a}^{k-1}\mathtt{bab}^{k-1}, \mathtt{a}^{k-1}\mathtt{b}^k\mathtt{a}\}$ holds (up to renaming and reversal). Moreover, there does not exist a weakly-0-balanced word $w \in \Sigma^{2k}_{\mathrm{wzb}}$ with a k-spectrum of cardinality $2k-i$ for $i \in [k-2]$.*

Proof. First, consider $w = \mathtt{a}^{k-1}\mathtt{bab}^{k-1}$. Since the k-spectrum of $\mathtt{a}^k\mathtt{b}^k$ is a subset of the k-spectrum of w, the k-spectrum of w has at least $k+1$ elements. Additionally, w has the scattered factors of the form $\mathtt{a}^i\mathtt{bab}^{k-2-i}$, which sum up to $k-1$. Hence, $|\mathrm{ScatFact}_k(w)| = k+1+k-1 = 2k$ holds. Moreover, $\mathtt{a}^{k-1}\mathtt{b}^k\mathtt{a}$ has all elements of $\mathtt{a}^k\mathtt{b}^k$'s k-spectrum as scattered factors. Here the word has in addition all words of the form $\mathtt{a}^i\mathtt{b}^{k-1-i}\mathtt{a}$ as scattered factors which sum up to $k-1$ as well. This proves that both words have a scattered factor set of cardinality $2k$.

The other direction will be proven by contraposition following the two main cases

$$\mathtt{a}^{k-1}\mathtt{bab}^{k-1} \quad \text{and} \quad \mathtt{a}^{k-1}\mathtt{b}^{k}\mathtt{a}.$$

First, assume $w = \mathtt{a}^{\ell}\mathtt{b}x$ for $\ell \in [k-2]_{\geqslant 2}$. Notice that it does not have to be considered that the word starts with one a, since this is symmetric to the reversal of the case $\mathtt{a}^{k-1}\mathtt{b}^{k}\mathtt{a}$. This implies $|x|_a = k - \ell$ and $|x|_b = k - 1$. Notice here $k - \ell < k - 1$. Thus, there exists a scattered factor x' of x of length $2(k-\ell)$ with $|x'|_a = |x'|_b = k - \ell$. By Lemma 3.1.1, it follows

$$|\operatorname{ScatFact}_{k-\ell}(y)| = k - \ell + 1 \Leftrightarrow y \in \{\mathtt{a}^{k-\ell}\mathtt{b}^{k-\ell}, \mathtt{b}^{k-\ell}\mathtt{a}^{k-\ell}\}$$

and $|\operatorname{ScatFact}_{k-\ell}(y)| > k - \ell + 1$ otherwise. This implies that the $(k-\ell)$-spectrum of x' is minimal with respect to cardinality if x' is either $\mathtt{a}^{k-\ell}\mathtt{b}^{k-\ell}$ or $\mathtt{b}^{k-\ell}\mathtt{a}^{k-\ell}$. For giving a lower bound of the cardinality of w's scattered factor set of length k, it is sufficient to only take these both options into consideration. This implies that it is not necessary to examine the cases where x contains other scattered factors with both $k - \ell$ a and b.

case 1: $x' = \mathtt{a}^{k-\ell}\mathtt{b}^{k-\ell}$
Thus, x contains $\ell - 1$ b which are not in x'.
case a: $x = \mathtt{b}^{\ell-1}\mathtt{a}^{k-\ell}\mathtt{b}^{k-\ell}$
In this case $w = \mathtt{a}^{\ell}\mathtt{b}^{\ell}\mathtt{a}^{k-\ell}\mathtt{b}^{k-\ell}$ holds and that the k-spectrum of $\mathtt{a}^{k}\mathtt{b}^{k}$ is a subset of $\operatorname{ScatFact}_k(w)$ follows.
case i: $\ell < k - \ell$
For all $s \in [\ell]$ the words $\mathtt{a}^{\ell-s}\mathtt{b}^{s}\mathtt{a}^{k-\ell}, \ldots, \mathtt{a}^{\ell}\mathtt{b}^{s}\mathtt{a}^{k-\ell-s}$ are well-defined and sum up to $s+1$. Moreover, for every $s_2 \in [k-\ell]$ exists $r_1 \in \mathbb{N}_0$ and exist $r_2, s_2 \in \mathbb{N}$ such that the words $\mathtt{a}^{r_1}\mathtt{b}^{s_1}\mathtt{a}^{r_2}\mathtt{b}^{s_2}$ with $s_1 + r_1 + s_2 + r_2 = k$ are all distinct and distinct to the aforementioned. Thus, in this case

$$k + 1 + \sum_{s=1}^{\ell}(s+1) + k - \ell = 2k + 1 - \ell + \frac{\ell(\ell+1)}{2} + \ell \geqslant 2k + 4$$

is a lower bound for $\operatorname{ScatFact}_k(w)$.
case ii: $\ell > k - \ell$
Consider here for $r \in [k-\ell]$ the words $\mathtt{b}^{\ell-r}\mathtt{a}^{r}\mathtt{b}^{k-\ell}, \ldots, \mathtt{b}^{\ell}\mathtt{a}^{r}\mathtt{b}^{k-\ell-r}$. For fixed r these are $r+1$. Moreover, in this case for all $r_1 \in [\ell]$ exist $s_1, r_2 \in \mathbb{N}$ and $s_2 \in \mathbb{N}$ such that the words $\mathtt{a}^{r_1}\mathtt{b}^{s_1}\mathtt{a}^{r_2}\mathtt{b}^{s_2}$ with $s_1 + r_1 + s_2 + r_2 = l$ are

all distinct and distinct to the aforementioned. In total this sums up to

$$k+1+\sum_{r=1}^{k-\ell}(r+1)+\ell = k+1+\frac{(k-\ell)(k-\ell+1)}{2}+(k-\ell)+\ell \geqslant 2k+4$$

different scattered factors.
case b: $x = \mathtt{a}^{k-\ell}\mathtt{b}^{k-1}$
Thus, $w = \mathtt{a}^{\ell}\mathtt{b}\mathtt{a}^{k-\ell}\mathtt{b}^{k-1}$ holds. Here it holds as well that the k-spectrum of $\mathtt{a}^k\mathtt{b}^k$ is a subset of $\mathrm{ScatFact}_k(w)$. Moreover, all words of the form $\mathtt{b}\mathtt{a}^r\mathtt{b}^s$ for $r+s = k-1$ and $r \in [k-\ell]$ are different scattered factors, i.e., $k-\ell$ many. Additionally, the words $\mathtt{a}^r\mathtt{bab}^s$ for $r+s = k-2$ and $r, s > 0$ are different scattered factors and distinct to the aforementioned. This sums up to $k+1+k-1+k-2 = 3k-2$ for the cardinality of $\mathrm{ScatFact}_k(w)$. This proves the claim for $k \geqslant 3$.
case 2: $x' = \mathtt{b}^{k-\ell}\mathtt{a}^{k-\ell}$
Consequently, $x \in \{\mathtt{b}^{k-1}\mathtt{a}^{k-\ell}, \mathtt{b}^{k-\ell}\mathtt{a}^{k-\ell}\mathtt{b}^{\ell-1}\}$ holds.
case a: $x = \mathtt{b}^{k-1}\mathtt{a}^{k-\ell}$
Hence, $w = \mathtt{a}^{\ell}\mathtt{b}^k\mathtt{a}^{k-\ell}$. Here only $\ell+1$ different scattered factors of the form $\mathtt{a}^r\mathtt{b}^s$ exist and $k-\ell$ of the form $\mathtt{b}^s\mathtt{a}^r$ with $r+s = k$ (notice that the latter ones are only $k-\ell$ since among all of them one is in common with the first ones). Finally, consider the words of the form $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$ with $r_1+r_2+s = k$ and $r_1, r_2, s > 0$. This sums up to $\ell+1+k-\ell+k$. By $\mathtt{a}^k \in \mathrm{ScatFact}_k(w)$, $|\mathrm{ScatFact}_k(w)| \geqslant 2k+2$ follows.
case b: $x = \mathtt{b}^{k-\ell}\mathtt{a}^{k-\ell}\mathtt{b}^{\ell-1}$
In this case $w = \mathtt{a}^{\ell}\mathtt{b}^{k-\ell+1}\mathtt{a}^{k-\ell}\mathtt{b}^{\ell-1}$ holds. Here the cardinality of the k-spectrum of w is determined analogously to case 1a. □

By Proposition 3.1.14 and Theorem 3.1.15, the possible cardinalities of $\mathrm{ScatFact}_3(w)$ for weakly-0-balanced words w of length 6 are completely characterized. Theorem 3.1.15 determines the first gap in the set of cardinalities of $|\mathrm{ScatFact}_k(w)|$ for $w \in \Sigma^{2k}_{\mathrm{wzb}}$: there does not exist a word $w \in \Sigma^{2k}_{\mathrm{wzb}}$ with $|\mathrm{ScatFact}_k(w)| = k+i+1$ for $i \in [k-2]$ and $k \geqslant 3$, since all words that are not of the form $\mathtt{a}^k\mathtt{b}^k$, $\mathtt{b}^k\mathtt{a}^k$, $\mathtt{a}^{k-1}\mathtt{bab}^{k-1}$, or $\mathtt{a}^{k-1}\mathtt{b}^k\mathtt{a}$ have a scattered factor set of cardinality at least $2k+1$. As the size of this first gap is linear in k, it is clear that the larger k is, the more unlikely it is to find a k-spectrum of a small cardinality.

In the following we will prove that the cardinalities $2k+1$ up to $3k-4$

are not reachable, i.e., $3k-3$ is the third smallest cardinality after $k+1$ and $2k$ (witnessed by, e.g., $\mathtt{a}^{k-2}\mathtt{b}^k\mathtt{a}^2$).

3.1.16 Lemma. *For $i \in [\lfloor \frac{k}{2} \rfloor]$ and $j \in [k-1]$ for $k \geqslant 4$ we get*

$$|\operatorname{ScatFact}_k(\mathtt{a}^{k-i}\mathtt{b}^k\mathtt{a}^i)| = k(i+1) - i^2 + 1.$$

Proof. Let be $i \in [\lfloor \frac{k}{2} \rfloor]_{\geqslant 2}$. The k-spectrum of $\mathtt{a}^{k-i}\mathtt{b}^k\mathtt{a}^i$ contains exactly all words of the form $\mathtt{a}^r\mathtt{b}^s\mathtt{a}^t$ with $r+s+t=k$ for $t \in [i]_0$, $r \in [k-i]_0$, and $s \in [k]_0$. If t and r are fixed, s is uniquely determined. Since all these scattered factors are different, the k-spectrum has $(i+1)(k-i+1) = k(i+1) - i^2 - 1$ elements. □

3.1.17 Lemma. *For $i \in [\lfloor \frac{k}{2} \rfloor]$ and $j \in [k-1]$ we have*

$$|\operatorname{ScatFact}_k(\mathtt{a}^{k-1}\mathtt{b}^2\mathtt{a}\mathtt{b}^{k-2})| = 3k-2.$$

Proof. The scattered factors $\mathtt{a}^{k-1}\mathtt{b}^2\mathtt{a}\mathtt{b}^{k-2}$ are of four different forms: $\mathtt{b}^r\mathtt{a}\mathtt{b}^t$, $\mathtt{a}^r\mathtt{b}^s\mathtt{a}$, $\mathtt{a}^r\mathtt{b}^s$, and $\mathtt{a}^r\mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}$. Notice that all these scattered factors are different if in the second one s is chosen greater than or equal to 1 and in the last one $r, s_1, s_2 \geqslant 1$ holds. The first and second one lead to two scattered factors, since for every $s \in [2]$ there are enough as at the beginning for padding from the left. The third form leads to $k+1$ different scattered as shown in Theorem 3.1.1. The last one is a little bit more complicated. Notice that r is at most $k-3$ since $s_1, s_2 > 0$ holds. In this case there exists only one possibility for choosing s_1 and s_2, namely $s_1, s_2 = 1$. If r is $k-4$ there exist two possibilities, namely $s_1 = 1$ and $s_2 = 2$ or vice versa. For $r \in [k-5]$ there exist always 2 possibilities for the bs between the as. This leads to $2(k-5)$ possibilities. Thus, we get $2+2+k+1+1+2+2(k-5) = 8+3k-10 = 3k-2$. □

3.1.18 Lemma. *For $i \in [\lfloor \frac{k}{2} \rfloor]$ and $j \in [k-1]$ for $k \geqslant 5$ we get*

$$|\operatorname{ScatFact}_k(\mathtt{a}^{k-2}\mathtt{b}^j\mathtt{a}\mathtt{b}^{k-j}\mathtt{a})| = k(2j+2) - 6j + 2$$

and

$$|\operatorname{ScatFact}_k(\mathtt{a}^{k-2}\mathtt{b}^j\mathtt{a}^2\mathtt{b}^{k-j})| = k(2j+1) - 4j + 2.$$

Proof. As in the proof of the second part of Lemma 3.1.16 the scattered factors can be categorized in the form $\mathtt{a}^r\mathtt{b}^s$, $\mathtt{b}^{r_1}\mathtt{a}^s\mathtt{b}^{r_2}$, $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$, and

$\mathtt{a}^{r_1}\mathtt{b}^{s_1}\mathtt{a}^{r_2}\mathtt{b}^{s_2}$, where with appropriately chosen exponents no factor is counted twice. Also as before, i can be chosen in $\left[\lfloor\frac{k}{2}\rfloor\right]$, since otherwise the proof is analogous for $k-i$. The first form contributes $k+1$ elements. The second and third form contribute $2i$ each, since s and r_2 are in $[2]$. For the last form a distinction is necessary. If $r = k-3$ holds, $\mathtt{a}^{k-3}\mathtt{bab}$ is the only scattered factor. If r is smaller than $k-3$, $2i$ possibilities for each $r \in [k-3]$ lead to scattered factors. This sums up to $k+1+2i+2i+1+2i(k-4) = k(2i+1)-4i+2$. By this, the first claim is proven.

For the second claim again scattered factors of different forms will be distinguished. Since also here the minimal k-spectrum is a subset of the k-spectrum of w, these $k+1$ elements count for the cardinality. There exists i many scattered factors of the form $\mathtt{a}^r\mathtt{b}^s\mathtt{a}^2$ and $k-2$ of the form $\mathtt{a}^r\mathtt{b}^s\mathtt{a}$, since with the last $\mathtt{a}$ all occurrences of $\mathtt{b}$ are before it. Assuming w.l.o.g. again that i is at most $\frac{k}{2}$ only $\mathtt{b}^{k-1}\mathtt{a}$ is a scattered factor of the form $\mathtt{b}^s\mathtt{a}^r$. The scattered factors of the form $\mathtt{b}^{r_1}\mathtt{a}\mathtt{b}^{r_2}\mathtt{a}$ contribute i many. The remaining two forms need again a case analysis. There exists exactly one scattered factor of the form $\mathtt{a}^r\mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}$ for $r = k-3$ and exactly one scattered factor of the form $\mathtt{a}^{r_1}\mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}\mathtt{a}$ for $r_1 = k-4$. If r resp. r_1 are smaller there exists i different scattered factors for each choice of $r \in [k-4]$ resp. $r_1 \in [k-5]$. This sums up to $k+1+k-2+i+i+1+i+1+i(k-5)+1+k(i-4) = 2k+2+3i+ik-5i+ik-4i = k(2+2i)-6i+2$. □

Notice that for $i \in \left[\lfloor\frac{k}{2}\rfloor\right]$ the sequence $(k(2i+1)-4i+2)_i$ is increasing and its minimum is $3k-2$ while for $i \in \left[\lfloor\frac{k}{2}\rfloor\right]$ the sequence $(k(2i+2)-6i+2)_i$ is increasing and its minimum is $4k-4$. The following lemma only gives lower bounds for specific forms of words, since, on the one hand, it proves to be sufficient for the Theorem 3.1.23 which describes the second gap, and, on the other hand, the proofs show that the formulas describing the exact number of scattered factors of a specific form are getting more and more complicated. It has to be shown that also words starting with i letters $\mathtt{a}$, for $i \in [k-3]$, have a k-spectrum of greater (as lower is already excluded) cardinality. By Lemma 3.1.16, only words with another transition from $\mathtt{a}$s to $\mathtt{b}$s need to be considered ($w = \mathtt{a}^{r_1}\mathtt{b}^{s_1}w_1\mathtt{a}^{r_1}\mathtt{b}^{s_2}$). W.l.o.g. we can assume s_1 to be maximal, such that w_1 starts with an $\mathtt{a}$, and similarly, by maximality of r_2, ends with a $\mathtt{b}$, thus, only words of the form $\mathtt{a}^{r_1}\mathtt{b}^{s_1}\cdots\mathtt{a}^{r_n}\mathtt{b}^{s_n}$ have to be considered, and by Proposition 3.1.5, it is sufficient to investigate $n < k$.

3.1.19 Lemma. *We have* $|\mathrm{ScatFact}_k(\mathtt{a}^{k-2}\mathtt{b}^i\mathtt{a}\mathtt{b}^j\mathtt{a}\mathtt{b}^{k-i-j})| \geqslant 3k-3$ *for* $i, j \in [k-2]$, $i+j \leqslant k-1$.

Proof. Choose $i, j \in [k-2]$. Then all words of the form $\mathtt{a}^r\mathtt{b}^s$ for $r, s \in [k]_0$ are scattered factors of some w_{ij} and, by Lemma 3.1.1, it follows that w_{ij} has $k+1$ scattered factors of this form. Scattered factors of the form $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$ can occur in three variants. In the first variant only the second block of $\mathtt{a}$s is involved after the first block of $\mathtt{b}$s, namely the second single $\mathtt{a}$ is not involved. Since $i \in [k-2]$ holds, for each $s \in [i]$ exists r_1, r_2 ($r_2 = 1$) such that $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$ is a scattered factor of w_{ij}, i.e., w_{ij} has additionally i scattered factors. The second variant uses the $\mathtt{a}$ of the second and the third $\mathtt{a}$-block. Thus, only scattered factors of the form $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$ are of interest, the second $\mathtt{b}$-block is not involved. If $i+j = k-1$ holds only $i-1$ scattered factors of this form occur, otherwise again i new elements are in the k-spectrum. If only the $\mathtt{a}$ from the third block is involved then j (resp. $j-1$) new elements are in the spectrum. This sums up to at least $2i+j-2$ elements of the form $\mathtt{a}^{r_1}\mathtt{b}^s\mathtt{a}^{r_2}$. A similar distinction leads to the number of scattered factors of the form $\mathtt{a}^{r_1}\mathtt{b}^{s_1}\mathtt{a}^{r_2}\mathtt{b}^{s_2}$. Assume first $r_2 = 1$ and for this only the $\mathtt{a}$ from the second $\mathtt{a}$-block. This implies that either only $\mathtt{b}$ from the second block or from the second and third block can be taken for the last $\mathtt{b}$-block in the scattered factor. Moreover, r_1, s_1, s_2 are at most $k-3$. For each choice of r_1 in $[k-3]$ there are $\min\{j, k-2-i\}$ possibilities, which leads to

$$i\left((k-j-2)j + \sum_{\ell=1}^{k-j-2} k-2-\ell\right) = 6i + 1\frac{1}{2}k^2 i - kji - 3\frac{1}{2}ki + 1\frac{1}{2}j^2 i + 1\frac{1}{2}ji.$$

If $\mathtt{b}$ from the second and third block are allowed, all $\mathtt{a}$s of the second block have to occur for obtaining different scattered factors to the previous ones. Thus,

$$\begin{aligned} & i\left((k-j-i-2)j + \sum_{\ell=1}^{k-j-i-2} k-2-\ell\right) \\ &= kij + \frac{1}{2}k^2 - 1\frac{1}{2}k - ik - jk - ij^2 - i^2 j - ij + 1\frac{1}{2}i + 1\frac{1}{2}j + \frac{1}{2}i^2 + \frac{1}{2}j^2. \end{aligned}$$

If both, the second and the third $\mathtt{a}$-block, are involved, there are $ik-1\frac{1}{2}i^2-ij-\frac{1}{2}i$ additional scattered factors in the k-spectrum. This all sums up to

$$k+1+9i-2+1\frac{1}{2}k^2i-3\frac{1}{2}ki+\frac{1}{2}j^2i-\frac{1}{2}ji+\frac{1}{2}j+\frac{1}{2}i^2+\frac{1}{2}j^2.$$

Since either $i^2 \geqslant ij$ or $j^2 \geqslant ij$ and $i, j \in [k-3]$ hold, this is greater than or equal to

$$1\frac{1}{2}k^2-2\frac{1}{2}k+9\frac{1}{2} \geqslant 3k-3.$$

Notice that, additionally, there exist scattered factors of other forms, which enlarge the k-spectrum. □

3.1.21 Lemma. *We have* $|\mathrm{ScatFact}_k(\mathtt{a}^{k-2}\mathtt{b}^{s_1}\mathtt{a}^{r_1}\mathtt{b}^{s_2}\mathtt{a}^{r_2}\mathtt{b}^{s_3})| \geqslant 3k-4$ *for* $s_1+s_2+s_3=k$, $r_1+r_2=2$, $s_1>0$, $r_1, r_2, s_2, s_3 \geqslant 0$.

Proof. Consider first the case when $s_2=0$, $r_1=0$, or $r_2=0$. This leads to words of the form matching Lemma 3.1.16 and, consequently, the k-spectrum has $k(2i+1)-4i+2 \geqslant 3k-2 > 3k-4$ elements. Consider now the case that $s_3=0$ holds and all other exponents are at least 1. By Lemma 3.1.16, it follows again that each such word has at least $k(2i+2)-6i+2 \geqslant 4k-4 > 3k-4$ elements. Finally, by Lemma 3.1.19, it follows that the remaining words of the given form have at least $3k-3$ scattered factors. □

3.1.22 Lemma. *We have* $|\mathrm{ScatFact}_k(\mathtt{a}^{r_1}\mathtt{b}^{s_1}\cdots\mathtt{a}^{r_n}\mathtt{b}^{s_n})| \geqslant 3k-3$ *for* $r_1 \leqslant k-3$, $\sum_{i\in[n]} r_i = \sum_{i\in[n]} s_i = k$, *and* $r_i, s_i \geqslant 1$.

Proof. Obviously $\mathtt{a}^k$ and $\mathtt{a}^{k-i}\mathtt{b}^i$ are scattered factors of s_n. Notice here, that the proof leads to s_{n-1} scattered factors, if in the claim $s_n=0$ would be allowed. Consider now the scattered factors of the form $\mathtt{a}^i\mathtt{b}^j$ for $i, j \in [k]$. Let m be the number of the block in which the i$^{\text{th}}$ $\mathtt{a}$ occurs. If $s_m+\ldots+s_n \geqslant k-i$ holds, $\mathtt{a}^i\mathtt{b}^{k-i}$ is a scattered factor of w. Consider the opposite. This implies that from the m$^{\text{th}}$ until the n$^{\text{th}}$ block less then $k-i$ $\mathtt{b}$ occur. Thus, in the blocks 1 to i there occur more than i $\mathtt{b}$. Since the i$^{\text{th}}$ $\mathtt{a}$ is in the m$^{\text{th}}$ block, from this point until the end there are $k-i$ $\mathtt{a}$. Hence, $\mathtt{b}^i\mathtt{a}^{k-i}$ is a scattered factor of w. So in each case at least one scattered factor occurs, i.e., at least $k+1$ scattered factors of this form are in the k-spectrum. Notice here, that the argument still holds if $s_m=0$ is allowed. With a similar argumentation the number of occurrences of the form $\mathtt{a}^i\mathtt{b}^j\mathtt{a}^{k-i-j}$ will be shown. If for a specific i, j-combination $\mathtt{a}^i\mathtt{b}^j\mathtt{a}^{k-i-j}$ is not a scattered factor, then choose m_1, m_2 such that the i^th^ a is in block m_1 and the j^th^ b after that is in block m_2. Thus, in the blocks $m_2 + 1$ to n are less than $k - i - j$ occurrences of a. Let r'_{m_1} be the a in the ${m_1}^{\text{th}}$ block which does not belong to $\mathtt{a}^i$. Then $r'_{m_1} + \ldots + r_{m_2}$ contains more than $k - j$ letters a since $k - j - i$ a occur in the ${m_1}^{\text{th}}$ to the n^{th} block. Thus, $\mathtt{a}^{r'_{m_1}}\mathtt{b}^{s_{m_1}} \cdots \mathtt{a}^{r_{m_2}}\mathtt{b}^{s'_{m_2}}$ is a scattered factor of length at least $k + 1$ where s'_{m_2} describes the part of the ${m_2}^{\text{th}}$ block until the j^th^ b. If $1 < m_1, m_2 < n$ holds, $\mathtt{b}\mathtt{a}^{k-j-1}\mathtt{b}^{j-2}$ is a scattered factor of w. If $m_1 = m_2 = 1$ holds, $\mathtt{a}^{k-j-3}\mathtt{bab}$ is a scattered factor. If both are equal to n, $\mathtt{b}\mathtt{a}^{k-j-1}\mathtt{b}^{j-2}$ is a scattered factor. In both cases the last b exists even if $s_m = 0$ holds, since the scattered factor ends in the examined block m_2. If $m_1 < m_2$ holds, there exists a factor of length $> k$ which can be narrowed to a factor starting in a, ending in b, and having at least one *switch* from b back to a and back to b. This gives at least $(k-2)^2$ scattered factors of the form $\mathtt{a}^i\mathtt{b}^j\mathtt{a}^{k-i-j}$ (or a different one in exchange). By $k^2 - k + 3 \geqslant 3k - 3$ for $k \geqslant 5$ follows the claim. □

By Lemmata 3.1.16 and 3.1.19, we are able to prove the following theorem, which shows the second gap in the set of cardinalities of $\mathrm{ScatFact}_k$ for words in Σ^{2k}_{wzb}.

3.1.23 Theorem. *For $k \geqslant 5$ there does not exist a word $w \in \Sigma^{2k}_{\text{wzb}}$ with k-spectrum of cardinality $2k + i$ for $i \in [k-4]$. In other words between $2k + 1$ and $3k - 4$ is a cardinality gap.*

Proof. Theorems 3.1.1 and 3.1.15 show that exactly the words $\mathtt{a}^k\mathtt{b}^k$, $\mathtt{a}^{k-1}$, $\mathtt{bab}^{k-1}$, and $\mathtt{a}^{k-1}\mathtt{b}^k\mathtt{a}$ have k-spectra of cardinality less than or equal to $2k$. By Lemma 3.1.16 and Lemma 3.1.19, it follows that $\mathtt{a}^{k-2}\mathtt{b}^k\mathtt{a}^2$ has a k-spectrum of cardinality $3k - 3$. Assume a $w \in \Sigma^{2k}_{\text{wzb}} \backslash \{\mathtt{a}^k\mathtt{b}^k, \mathtt{a}^{k-1}\mathtt{ba}\ \mathtt{b}^{k-1}, \mathtt{a}^{k-1}\mathtt{b}^k\mathtt{a}, \mathtt{a}^{k-2}\mathtt{b}^k\mathtt{a}^2\}$. Since renaming and reversal do not influence the cardinality, it can be assumed that w starts with a. By assumption, w does not start with $\mathtt{a}^k$. If w starts with $\mathtt{a}^{k-1}$, $w = \mathtt{a}^{k-1}\mathtt{b}^i\mathtt{ab}^{k-i}$ follows with $i \in [k-1]_{\geqslant 2}$ and, by Lemma 3.1.16, the k-spectrum has $(i+1)k - 4i + 6 \geqslant 3k - 2 > 3k - 4$ elements. By Lemma 3.1.19, the claim follows for words starting with $(k-2)$ a and it is shown that words starting with at least two and at most $k - 3$ a lead to k-spectra of cardinality greater than $3k - 3$. □

Going further, we analyse the larger possible cardinalities of $\mathrm{ScatFact}_k$, trying to see what values are achievable (even if only asymptotically, in some cases).

3.1.24 Corollary. *All square numbers, greater or equal to four, occur as the cardinality of the k-spectrum of a word $w \in \Sigma_{\mathrm{wzb}}^{2k}$; in particular for even k we have $|\,\mathrm{ScatFact}_k(\mathtt{a}^{\frac{k}{2}}\mathtt{b}^k\mathtt{a}^{\frac{k}{2}})| = \left(\frac{k}{2}+1\right)^2$.*

Proof. Apply Lemma 3.1.16 to $i = \frac{k}{2}$. This implies that the cardinality of the k-spectrum of $\mathtt{a}^{\frac{k}{2}}\mathtt{b}^k\mathtt{a}^{\frac{k}{2}}$ is

$$k\left(\frac{k}{2}+1\right) - \frac{k^2}{4} - 1 = \frac{1}{4}k^2 + k - 1 = \left(\frac{k}{2}+1\right)^2.$$

□

Inspired by the previous Corollary, we can show the following result concerning the asymptotic behaviour of the cardinality of $\mathrm{ScatFact}_k$ for words of length $2k$.

3.1.25 Proposition*. *Let $i > 1$ be a fixed (constant) integer. Let $d = \lfloor \frac{k}{i} \rfloor$ and $r = k - di$, and $d' = \lfloor \frac{k}{i-1} \rfloor$ and $r' = k - d'(i-1)$. Then the following hold:*

▻ *the word $\mathtt{a}^r\mathtt{b}^r(\mathtt{a}^d\mathtt{b}^d)^i$ has $\Theta(k^{2i-1})$ scattered factors of length n;*

▻ *the word $\mathtt{a}^r\mathtt{b}^{r'}(\mathtt{a}^d\mathtt{b}^{d'})^{i-1}\mathtt{a}^d$ has $\Theta(k^{2i-2})$ scattered factors of length n.*

3.1.26 Remark.* Let i be an integer, and consider k another integer divisible by i. Consider the word $w_k = (a^{\frac{k}{i}}b^{\frac{k}{i}})^i$. The exact number of scattered factors of length k of w_k is equal to the number $C\left(k, 2i, \frac{k}{i}\right)$ of weak $2i$-compositions of k, whose terms are bounded by $\frac{k}{i}$, i.e., the number of ways in which k can be written as a sum $\sum_{j\in[2i]} r_j$ where $r_j \in \left[\frac{k}{i}\right]_0$. From Proposition 3.1.25, we also get that this number is $\Theta(n^{2k-1})$, but we also have:

$$C\left(k, 2i, \frac{k}{i}\right) = \sum_{0\leq j<M} (-1)^j \binom{2i}{j} \binom{k+2i-j(\frac{k}{i}+1)-1}{2i-1},$$

for $M = \frac{i(k+2i-1)}{k+i}$. It is known that there exists a constant $E > 0$ such that

$$C\left(k, 2i, \frac{k}{i}\right) \leqslant E \cdot \sum_{0 \leqslant j < M} (-1)^j \binom{2i}{j} \left(k + 2i - j\left(\frac{k}{i} + 1\right) - 1\right)^{2i-1}.$$

The coefficient of k^{2i-1} in the right hand side of this inequality has to be positive. Consequently, $\sum_{0 \leqslant j < M} (-1)^j \binom{2i}{j} (i-j)^{2i-1} > 0$. This seems to be an interesting combinatorial inequality in itself.

One can also show as in Proposition 3.1.25 that the number of scattered factors of length k of w_k, which have, at their turn, $(ab)^i$ as a scattered factor, is $\Theta(k^{2i-1})$. This number is also equal to the number $C'\left(k, 2i, \frac{k}{i}\right)$ of $2i$-compositions of k whose terms are strictly positive integers upper bounded by $\frac{k}{i}$, i.e., the number of ways in which k can be written as a sum $\sum_{j \in [2i]} r_j$ where $r_j \in \left[\frac{k}{i}\right]$. Just as above, from this we get $\sum_{0 \leqslant j < i} (-1)^j \binom{2i}{j} (i-j)^{2i-1} > 0$. Again, this inequality seems interesting to us.

We will end this analysis with the conjecture that, in contrast to the first gap, which always starts immediately after the first obtainable cardinality, the last gap ends earlier the larger k is. More precisely, if $w = \mathtt{a}^2\mathtt{b}^2(\mathtt{ab})^{k-3-i}\mathtt{ba}(\mathtt{ab})^i$ for $k \in \mathbb{N}_{\geqslant 4}$, $i \in [k-2]_0$ then $|\mathrm{ScatFact}_k(w)| = 2^k - 2 - i$.

At the end of this section, we will briefly introduce θ-palindromes in this specific setting. Let $\theta : \Sigma^* \to \Sigma^*$ be an antimorphic involution, i.e., $\theta(uv) = \theta(v)\theta(u)$ and θ^2 is the identity on Σ^*. If $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ then only the identity and renaming are such mappings. The fixed points of θ are called θ-palindromes ($\mathtt{ab}^3.\theta(\mathtt{b})^3\theta(\mathtt{a})$) and they exactly the words where $w^R = \overline{w}$ holds. They were well studied in different fields (see, e.g., [37], [68]). A word $w \in \Sigma^{2k}_{\mathrm{wzb}}$ is a θ-palindrome iff either $w \in \{\mathtt{a}w'\mathtt{b}, \mathtt{b}w'\mathtt{a}\}$ for some θ-palindrome $w' \in \Sigma^{2(k-1)}_{\mathrm{wzb}}$ or additionally $w = \mathtt{a}^{\frac{k}{2}}\mathtt{b}^k\mathtt{a}^{\frac{k}{2}}$ in the case that k is even. Two cardinality results for θ-palindromes are presented in Lemma 3.1.16 and Corollary 3.1.24. We believe that persuing the k-spectra of θ-palindromes may lead to a deeper insight into which cardinalities can be reached, but due to space restrictions we will only mention one conjecture here, which may already show that cardinalities are somehow propagating for θ-palindromes. Notice that this conjecture implies that indeed similar to the second gap here $4k-4$ is always reached but that in contrast to the second gap, the third gap is not of the form $4k-4-i$ for $i \in [k-4]$.

3.1.27 Conjecture. *The k-spectrum of $w = \mathtt{a}\mathtt{b}^{k-1}\mathtt{a}^{k-1}\mathtt{b}$ has $4(k-1)$ elements and, moreover, if $w' = w^R$ with a k-spectrum of cardinality $\ell \in \mathbb{N}_{\geqslant 12}$ then the scattered factor set of $\mathtt{a}w\mathtt{b}$ has cardinality $2\frac{1}{4}\ell - 5$.*

3.2 Scattered Factor Universality

This subsection is mainly based on [5]. Recall that a word $w \in \Sigma^*$ is called k-universal w.r.t Σ (for a given $k \in \mathbb{N}$) if $\mathrm{ScatFact}_k(w) = \Sigma^k$.

Our first result extends and improves the results of Fleischer and Kufleitner [45].

3.2.1 Theorem*. *Given a word w over an integer alphabet Σ, with $|w| = n$, and a number $k \leqslant n$, we can compute the shortlex normal form of w w.r.t. $\sim_k$ in time $O(n)$. Moreover, given two words w', w'' over an integer alphabet Σ, with $|w'| \leqslant |w''| = n$, and a number $k \leqslant n$, we can test if $w' \sim_k w''$ in time $O(n)$.*

Proof. The main idea of the algorithm is that checking $w' \sim_k w''$ is equivalent to checking whether the shortlex normal forms w.r.t. $\sim_k$ of w' and w'' are equal. To compute the shortlex normal form of a word $w \in \Sigma^n$ w.r.t. $\sim_k$ the following approach was used in [45]: first, for each position of w the x- and y-coordinates were defined. The x-coordinate of i, denoted x_i, is the length of the shortest sequence of indices $1 \leqslant i_1 < i_2 < \ldots < i_t = i$ such that i_1 is the position where the letter $w[i_1]$ occurs in w for the first time and, for $1 < j \leqslant t$, i_j is the first position where $w[i_j]$ occurs in $w[i_{j-1}+1..i]$. Obviously, if $\mathtt{a}$ occurs for the first time at position i in w, then $x_i = 1$ (see [45] for more details). A crucial property of the x-coordinates is that if $w[\ell] = w[i] = \mathtt{a}$ for some $i > \ell$ such that $w[j] \neq \mathtt{a}$ for all $\ell + 1 \leqslant j \leqslant i-1$, then $x_i = \min\{x_\ell, x_{\ell+1}, \ldots, x_{i-1}\} + 1$. The y-coordinate of a position i, denoted y_i, is defined symmetrically: y_i is the length of the shortest sequence of indices $n \geqslant i_1 > i_2 > \ldots > i_t = i$ such that i_1 is the position where the letter $w[i_1]$ occurs last time in w and, for $1 < j \leqslant t$, i_j is the last position where $w[i_j]$ occurs in $w[i..i_{j-1}-1]$. Clearly, if $w[\ell] = w[i] = \mathtt{a}$ for some $i < \ell$ such that $w[j] \neq \mathtt{a}$ for all $\ell - 1 \geqslant j \geqslant i + 1$, then $y_i = \min\{y_{i+1}, \dots, y_{\ell-1}, y_\ell\} + 1$.

Computing the coordinates is done in two phases: the x-coordinates are computed and stored (in an array x with elements $x_1, \dots, x_n$) from left to right in phase 1a, and the y-coordinates are stored in an array y with elements $y_1, \dots, y_n$ and computed from right to left in phase 1b (while dynamically deleting a position whenever the sum of its coordinates is greater then $k + 1$ (cf. [45, Prop. 2])). Then, to compute the shortlex normal form, in a third phase, labelled phase 2, if letters $\mathtt{b} > \mathtt{a}$ occur consecutively in this order, they are interchanged whenever they have the same x- and y-coordinates and the sum of these coordinates is $k + 1$ (until this situation does not occur anymore).

We now show how these steps can be implemented in $O(n)$ time for input words over integer alphabets. For simplicity, let $x[i..j]$ denote the sequence of coordinates $x_i, x_{i+1}, \dots, x_j$; $\min(x[i..j])$ denotes $\min\{x_i, \dots, x_j\}$. It is clear that in $O(n)$ time we can compute all values $\mathrm{last}[i] = \max(\{0\} \cup \{j < i | w[j] = w[i]\})$.

First, phase 1a. For simplicity, assume that $x_0 = 0$. Increasing i from 1 to n, we maintain a list L of positions $0 = i_0 < i_1 < i_2 < \dots < i_t = i$ such that the following property is invariant: $x_{i_{\ell-1}} < x_{i_\ell}$ for $1 \leqslant \ell \leqslant t$ and $x_p \geqslant x_{i_\ell}$ for all $i_{\ell-1} < p \leqslant i_\ell$. After each i is read, if $\mathrm{last}[i] = 0$ then set $x_i = 1$; otherwise, determine $x_i = \min(x[\mathrm{last}[i]..i-1]) + 1$ by L, then append i to L and update L accordingly so that its invariant property holds. This is done as follows: we go through the list L from right to left (i.e., inspect the elements $i_t, i_{t-1}, \dots$) until we reach a position $i_{j-1} < \mathrm{last}[i]$ or completely traverse the list (i.e., $i_{j-1} = 0$). Let us note now that all elements x_ℓ with $i - 1 \geqslant \ell \geqslant \mathrm{last}[i]$ fulfil $x_\ell \geqslant x_{i_j}$ and $i_j \geqslant \mathrm{last}[i]$. Consequently, $x_i = x_{i_j} + 1$. Moreover, $x_{i_{j+1}} \geqslant x_{i_j} + 1$. As such, we update the list L so that it becomes $i_1, \dots, i_j, i$ (and x_i is stored in the array x).

Note that each position of w is inserted once in L and once deleted (but never reinserted). Also, the time needed for the update of L caused by the insertion of i is proportional to the number of elements removed from the list in that step. Accordingly, the total time needed to process L, for all i, is $O(n)$. Clearly, this procedure computes the x-coordinates of all positions of w correctly.

Second, phase 1b. We cannot proceed exactly like in the previous case, because we need to dynamically delete a position whenever the sum of its coordinates is greater than $k+1$ (i.e., as soon as we finished computing its y-coordinate and see that it is strictly greater than $k+1$; this position does not influence the rest of the computation). If we would proceed just as above (right to left this time), it might be the case that after computing some y_i we need to delete position i, instead of storing it in our list and removing some of the elements of the list. As such, our argument showing that the time spent for inspecting and updating the list in the steps where the y-coordinates are computed amortises to $O(n)$ would not work.

So, we will use an enhanced approach. For simplicity, assume that $y_{n+1} = 0$ and that every time we should eliminate position i we actually set y_i to $+\infty$. Also, let $y[i..j]$ denote the sequence of coordinates $y_i, y_{i+1}, \dots, y_j$; note that some of these coordinates can be $+\infty$. Let $\min(y[i..j])$ denote the minimum in the sequence $y[i..j]$. Similarly to what we did in phase 1a, while increasing i from n to 1, we maintain a list L' of positions $n+1 = i_0 > i_1 > i_2 > \dots > i_t \geqslant i$ such that the following property is invariant: $y_{i_{\ell-1}} < y_{i_\ell}$ for $1 \leqslant \ell \leqslant t$ and $y_p \geqslant y_{i_\ell}$ for all $i_{\ell-1} > p \geqslant i_\ell$. In the current case, we also have that $y_p = +\infty$ for all $i_t > p \geqslant i$. The numbers $i_0, i_1, i_2, \dots, i_t \geqslant i$ contained in the list L' at some moment in our computation define a partition of the universe $[1, n]$ in intervals: $\{1\}, \{2\}, \dots, \{i-1\}, [i, i_{t-1} - 1], [i_{t-1}, i_{t-2} - 1], \dots, [i_1, i_0 - 1]$ for which we define an *interval union-find* data structure [52, 60]; here the singleton $\{a\}$ is seen as the interval $[a, a]$. According to [60], in our model of computation, such a structure can be initialized in $O(n)$ time such that we can perform a sequence of $O(n)$ `union` and `find` operations on it in $O(n)$ time, with the crucial restriction that one can only unite neighbouring intervals. We assume that `find(j)` returns the bounds of the interval stored in our data structure to which j belongs. From the definition of the list L', it is clear that, before processing position i (and after finishing processing position $i+1$), $y_{i_\ell} = \min(y[i+1..i_{\ell-1} - 1])$ holds. We maintain a new array $\text{next}[\cdot]$ with $|\Sigma|$ elements: before processing position i, $\text{next}[w[i]]$ is the smallest position $j > i$ where $w[i]$ occurs after position i, which was not eliminated (i.e., smallest $j > i$ with $y_j \neq +\infty$), or 0 if there is no such position. Position i is now processed as follows: let $[a, b]$ be the interval returned by `find`$(\text{next}[i])$. If $a = i+1$ then let $\min = y_{i_t}$; if $a > i+1$ then there exists j such that $[a, b] = [i_j, i_{j-1} - 1]$ and $t > j > 0$, so let $\min = y_j$.

Let now $y = \min +1$, and note that we should set $y_i = y$, but only if $x_i + i \leqslant k + 1$. So, we check whether $x_i + i \leqslant k + 1$ and, if yes, let $y_i = y$ and set $\text{next}[w[i]] = i$; otherwise, set $y_i = +\infty$ (note that position i becomes, as such, irrelevant when the y-coordinate is computed for other positions). If $y_i = +\infty$ then make the union of the intervals $\{i\}$ and $[i+1, i_{t-1} - 1]$ and start processing $i - 1$; L' remains unchanged. If $y_i \neq +\infty$ then make the union of the intervals $\{i\}, [i+1, i_{t-1} - 1], \ldots, [i_{j+1}, i_j - 1]$ and start processing $i - 1$; L' becomes $i, i_j, i_{j-1}, \ldots, i_0$.

As each position of w is inserted at most once in L', and then deleted once (never reinserted), the number of list operations is $O(n)$. The time needed for the update of L', caused by the insertion of i in L', is proportional to the number of elements removed from L' in that step, so the total time needed (exclusively) to process L is $O(n)$. On top of that, for each position i, we run one `find` operation and a number of `union` operations proportional to the number of elements removed from L' in that step. Overall we do $O(n)$ `union` and `find` operations on the *union-find* data structure. This takes in total, for all i, $O(n)$ time (including the initialisation). Thus, the time complexity of phase 1b is linear.

Third, phase 2. Assume that w_0 is the input word of this phase. Clearly, $|w_0| = m \leqslant n$, and we have computed the coordinates for all its positions (and maybe eliminated some positions of the initial input word w). We partition in linear time $O(n)$ the interval $[1, m]$ into $2t + 1$ (possibly empty) lists of positions $L_1, \ldots, L_{2t+1}$ such that the following conditions hold. First, all elements of L_i are smaller than those of L_{i+1} for $1 \leqslant i \leqslant 2t$. Second, for i odd, the elements j in L_i have $x_j + y_j < k + 1$; for each i even, there exist a_i, b_i such that $a_i + b_i = k + 1$ and for all j in L_i we have $x_j = a_i, y_j = b_i$. Third, we want t to be minimal with these properties. We now produce, also in linear time, a new list U: for each $i \leqslant t$ and $j \in L_{2i}$ we add the triplet $(i, w[j], j)$ in U. We sort the list of triples U (cf. [45, Prop. 10]) with radix sort in linear time [21]. After sorting it, U can be decomposed in t consecutive blocks $U_1, U_2, \ldots, U_t$, where U_i contains the positions of L_{2i} sorted w.r.t. the order on Σ (i.e., determined by the second component of the pair). As such, U_i induces a new order on the positions of w_0 stored in L_{2i}. We can now construct a word w_1 by just writing in order the letters of w_0 corresponding to the positions stored in L_i, for i from 1 to $2t + 1$, such that the letters of L_i are written in the original order, for i odd, and in the order induced by U_i, for i even. Clearly, this is a correct implementation of phase 2 which runs in linear time. The word w_1 is the shortlex normal form of w.

Summing up, we have shown how to compute the shortlex normal form of a word in linear time (for integer alphabets). Both our claims follow. □

The above theorem improves the complexity of the algorithm reported in [45], where the problem was solved in $O(n|\Sigma|)$ time. As such, over integer alphabets, testing Simon congruence for a given k can be done in optimal time, that does not depend on the input alphabet or on k. When no restriction is made on the input alphabet, we can first sort it, replace the letters by their ranks, and, as such, reduce the problem to the case of integer alphabets. In that case, testing Simon congruence takes $O(|\Sigma| \log |\Sigma| + n)$ time which is again optimal: for $k = 1$, testing if $w_1 \sim_1 w_2$ is equivalent (after a linear time processing) to testing whether two subsets of Σ are equal, and this requires $\Theta(|\Sigma| \log |\Sigma|)$ time [29].

3.2.1 Combinatorial Results

In this section we present several algorithmic and combinatorial results.

3.2.2 Remark. Theorem 3.2.1 allows us to decide in linear time $O(n)$ whether a word w over $\Sigma = \{1 < 2 < \ldots < \sigma\}$ is k-universal, for a given $k \leqslant n$. We compute the shortlex normal form of w w.r.t. $\sim_k$ and check whether it is $(1 \cdot 2 \cdots \sigma)^k$.

Actually, we can compute the universality index $\iota(w)$ efficiently by computing its arch factorisation in linear time in $|w|$. Moreover this allows us to check whether w is k-universal for some given k by just checking if $\iota(w) \geqslant k$ or not.

3.2.3 Proposition*. *Given a word $w \in \Sigma^n$, $n \in \mathbb{N}$, we can compute $\iota(w)$ in linear time $O(n)$.*

Proof. We actually compute the number ℓ of arches in the arch factorisation. For a lighter notation, we use $u_i = \mathrm{ar}_w(i)$ for $i \in [\ell]_0$. The factors u_i can be computed in linear time as follows. We maintain an array C of $|\Sigma|$ elements, whose elements are initially all 0, and a counter h, which is initially $|\Sigma|$. For simplicity, let $m_0 = 0$. We go through the letters $w[j]$ of $w[m_{i-1}+1..n]$, from left to right, and if $C[w[j]]$ equals 0, we decrement h by 1 and set $C[w[j]] = 1$. Intuitively, we keep track of which letters of Σ we meet while traversing $w[m_{i-1}+1..n]$ using the array C, and in h we store how many letters we still need to see. As soon as $h = 0$ or $j = n$, we stop: set $m_i = j$ (the position of the last letter of w we read), $u_i = w[m_{i-1}+1..m_i]$ (the i^{th} arch), and $h = |\Sigma|$ again. If $j < n$ then reinitialize all elements of C to 0 and restart the procedure for $i+1$. Note that if $j = n$ then u_i is $r(w)$ as introduced in the definition of the arch factorization. The time complexity of computing u_j is $O(|u_j|)$, because we process each symbol of $u_i = w[m_{i-1}+1..m_i]$ in $O(1)$ time, and, at the end of the procedure, we reinitialize C in $O(|\Sigma|)$ time iff u_i contained all letters of Σ, so $|u_i| \geqslant |\Sigma|$. The conclusion follows. □

The following combinatorial result characterises universality by repetitions.

3.2.4 Remark. For all $w \in \Sigma^*$ and all $i, k \in \mathbb{N}$ we have $\mathrm{ScatFact}_k(w^i) \subseteq \mathrm{ScatFact}_k(w^{i+1})$ since w^i is a factor of w^{i+1}.

3.2.5 Theorem. *A word $w \in \Sigma^{\geqslant k}$ with $alph(w) = \Sigma$ is k-universal for $k \in \mathbb{N}_0$ iff $\mathrm{ScatFact}_k(w^n) = \mathrm{ScatFact}_k(w^{n+1})$ for all $n \in \mathbb{N}$. Moreover, we have $\iota(w^n) \geqslant kn$ if $\iota(w) = k$.*

Proof. For the second claim, we get immediately that w^n is at least kn-universal if $\iota(w) = k$, since the arch factorisation of w occurs in each w of the repetition. For the first claim, assume first w to be k-universal, i.e., we have $\mathrm{ScatFact}_k(w) = \Sigma^k$. This implies $\Sigma^k \subseteq \mathrm{ScatFact}_k(w^n)$ for all $n \in \mathbb{N}$. On the other hand we have $\mathrm{ScatFact}_k(w^n) \subseteq \Sigma^k$ and, thus, $\mathrm{ScatFact}_k(w^n) = \Sigma^k = \mathrm{ScatFact}_k(w^{n+1})$ for all $n \in \mathbb{N}$. For the second direction assume that there exists an $n \in \mathbb{N}$ with $\mathrm{ScatFact}_k(w^n) \neq \mathrm{ScatFact}_k(w^{n+1})$. Choose n minimal. By Remark 3.2.4, we get $\mathrm{ScatFact}_k(w^n) \subset \mathrm{ScatFact}_k(w^{n+1})$. Let $v \in \mathrm{ScatFact}_k(w^{n+1}) \setminus \mathrm{ScatFact}_k(w^n)$. Again by Remark 3.2.4, if v were a scattered factor of w, v would be a scattered factor of w^n - a contradiction. Thus, w is not k-universal. □

As witnessed by $w = \mathtt{aabb} \in \{\mathtt{a}, \mathtt{b}\}^*$, the universality index $\iota(w^n)$ can be greater than $n \cdot \iota(w)$: w is universal but $w^2 = \mathtt{aab.ba.ab.b}$ is 3-universal.

We study this phenomenon at the end of this section. Theorem 3.2.5 can also be used to compute an discommon scattered factor of w and ww over arbitrary alphabets; note that the shortest such a factor has to have length $k+1$ if $\iota(w) = k$.

3.2.6 Proposition*. *Given a word $w \in \Sigma^*$ we can compute in linear time $O(|w|)$ one of the discommon scattered factors of w und ww of minimal length.*

Proof. Just like in the proof of Theorem 3.2.3, we compute the decomposition (arch factorisation) $w = u_1 \cdots u_k$ such that, for $i \in [k-1]$, the factor $w[1..m_i] = u_1 \cdots u_i$ is the shortest prefix of w such that $\Sigma^i \subseteq \mathrm{ScatFact}_i(w[1..m_i])$, and u_k (called in the arch factorisation *the rest*) either does not contain all letters of Σ or it does, but if we remove its last letter then it does not contain all letters of Σ anymore, i.e., $\Sigma^k \subseteq \mathrm{ScatFact}_k(w)$ but $\Sigma^k \nsubseteq \mathrm{ScatFact}_k(w[1..n-1])$.

If u_k does not contain all letters of Σ, then $k > 1$ (as w contains all letters of Σ). The procedure described in the proof of Theorem 3.2.3 identifies a letter $\mathtt{a}$ that does not occur in u_k. We construct the word $x = w[m_1]w[m_2] \cdots w[m_{k-1}]\mathtt{a} = m(w)\mathtt{a}$ (where $m(w)$ is defined w.r.t. the arch factorisation). Then x is not a scattered factor of w (and all shorter words are scattered factors of w), but x is scattered factor of ww (as $\mathtt{a}$ occurs in w, because $k > 1$). Indeed, if x were a scattered factor of w, then its i^{th} letter should correspond to the letter occurring at position $j_i \geqslant m_i$ of w. This is clear for m_1: if $w[m_1]$ occurred also to the left of m_1 in w, then u_1 would not be the shortest prefix of $w[1..n]$ that contains all letters of Σ. Then, for $i \geqslant 1$, assume the property holds for the first i letters of x. We show it for $i+1$. So, $x[i+1]$ should correspond to a letter of w occurring to the right of $x[i]$, i.e., at a position strictly greater than m_i. But $x[i+1] = w[m_{i+1}]$ occurs for the first time to the left of m_i at position m_{i+1}. So, our statement is correct. Now, if the $(k-1)^{\text{th}}$ letter of x occurs on a position greater or equal to m_{k-1}, then the last letter of x, namely $\mathtt{a}$, should occur in $u_k = w[m_{k-1}+1..m_k]$, a contradiction.

If u_k contains all letters of Σ, then let $x = w[m_1]w[m_2] \cdots w[m_k]\mathtt{a} = m(w)\mathtt{a}$, for some $\mathtt{a} \in \Sigma$. Just like before, we can show that x is not a scattered factor of w, but all shorter words are scattered factors of w; also x is clearly a scattered factor of ww.

Running the procedure described in Theorem 3.2.3 takes linear time, and constructing x also takes linear time. The conclusion follows. □

3.2.7 Remark. By Proposition 3.2.6, computing the shortest discommon scattered factor of w and ww takes optimal $O(n)$ time, which is more efficient than running an algorithm computing the shortest discommon scattered factor of two arbitrary words (see, e.g., [34, 45]. Note that we are not aware of any linear time algorithm performing this task for integer alphabets). In particular, we can use Theorem 3.2.1 to find by binary search the smallest k for which two words have distinct k-spectra in $O(n \log n)$ time. In [57] a linear time algorithm solving this problem is given for binary alphabets; an extension seems non-trivial.

Continuing the idea of Theorem 3.2.5, we investigate even-length palindromes, i.e., appending w^R to w. The first result is similar to Theorem 3.2.5 for $n = 1$. Notice that $\iota(w) = \iota(w^R)$ follows immediately with the arch factorisation.

3.2.8 Corollary. *A word w is k-universal iff* $\mathrm{ScatFact}_k(w) = \mathrm{ScatFact}_k(ww^R)$.

Proof. The proof is analogous to the one of Theorem 3.2.5. □

In contrast to $\iota(w^2)$, $\iota(ww^R)$ is never greater than $2\iota(w)$.

3.2.9 Proposition. *Let $w \in \Sigma^*$ be a palindrome and $u = \mathrm{Pref}_{\lfloor \frac{|w|}{2} \rfloor}(w)$ with $\iota(u) = k \in \mathbb{N}$. Then $\iota(w) = 2k$ if $|w|$ even and $\iota(w) = 2k+1$ iff $w[\frac{n+1}{2}] \cup alph(r(u)) = \Sigma$ if $|w|$ is odd.*

Proof. First, consider $|w| \equiv_2 0$, i.e., $w = uu^R$. By $\iota(u) = k$, u has an arch factorisation with k factors which also occur in u^R. This implies $\iota(ww^R) \geqslant 2k$. Suppose $\iota(uu^R) = 2k+1$. Let $uu^R = \mathrm{ar}_{uu^R}(1) \cdots \mathrm{ar}_{uu^R}(2k+1) r(uu^R)$ be the arch factorisation. Since k is maximal, $\mathrm{ar}_{uu^R}(1) \cdots \mathrm{ar}_{uu^R}(k+1)$ is not a prefix of u, i.e., $\mathrm{ar}_{uu^R}(k+2)$ is a factor of u^R and, thus, $\mathrm{ar}_{uu^R}(k+2) \ldots \mathrm{ar}_{uu^R}(2k+1) r(uu^R)$ is a suffix of u^R. Hence, we get $\mathrm{ar}_{uu^R}(k+1) = r(u)y$ for a prefix y of u^R. If $|r(u)| = |y|$ we have $r(u) = y^R$ and, thus, $\Sigma = \mathrm{alph}(\mathrm{ar}_{uu^R}(k+1)) = \mathrm{alph}(r(u)) \subset \Sigma$. If $|r(u)| < |y|$, then $r(u)^R$ is a prefix of y. This implies $\Sigma = \mathrm{alph}(\mathrm{ar}_{uu^R}(k+1)) = \mathrm{alph}(y)$ and, consequently, we found an arch factorisation of w (the second one) with $k+1$ factors. Consider now $|r(u)| > |y|$. Then y^R is a suffix of $r(u)$ but, by the definition of the arch factorisation, $y[|y|]$ does not occur in $r(u)[1..|r(u)|-1]$. Since we get a contradiction in all three cases, the claim is proven for even-length palindromes.

By a similar argument, odd-length palindromes have to have exactly the letter in the middle which is missing in $r(u)$ to be 1-universal. □

3.2.10 Remark. If we consider the universality of a word $w = w_1 \cdots w_m$ for $m \in \mathbb{N}$ with $w_i \in \{u, u^R\}$ for a given word $u \in \Sigma^*$, then a combination of the previous results can be applied. Each time either u^2 or $(u^R)^2$ occurs Theorem 3.2.5 can be applied (and the results about circular universality that finish this section). Whenever uu^R or $u^R u$ occur in w, the results of Proposition 3.2.9 are applicable.

Another generalisation of Theorem 3.2.5 is to investigate concatenations under permutations: for a morphic permutation π of Σ can we compute $\iota(w\pi(w))$?

3.2.11 Lemma. *Let $\pi : \Sigma^* \to \Sigma^*$ be a morphic permutation. Then $\iota(w) = \iota(\pi(w))$ for all $w \in \Sigma^*$ and especially the image of the arch factorisation is the arch factorisation of the image.*

Proof. Let $w \in \Sigma^*$ and $w = \mathrm{ar}_w(1) \cdots \mathrm{ar}_w(k) r(w)$ be the arch factorisation of w for an appropriate $k \in \mathbb{N}_0$. By the definition of the arch factorisation, $\mathrm{ar}_w(i)[|\mathrm{ar}_w(i)|]$ does not occur in $\mathrm{ar}_w(i)[1..|\mathrm{ar}_w(i)| - 1]$ for all $i \in [k]$. Set $k_i = \sum_{j=1}^{i} |\mathrm{ar}_w(j)|$ for $i \in [k]$. Thus, $\pi(\mathrm{ar}_w(i)[|\mathrm{ar}_w(i)|])$ occurs only once in $\pi(w)[k_i + 1..k_{i+1}]$ and exactly as the last letter. Hence, $\pi(\mathrm{ar}_w(1)) \cdots \pi(\mathrm{ar}_w(k))\pi(r(w))$ is the arch factorisation of $\pi(w)$. The other direction follows by applying π^{-1} as a permutation to $\pi(w)$. □

By Lemma 3.2.11, we have $2\iota(w) \leqslant \iota(w\pi(w)) \leqslant 2\iota(w) + 1$. Consider the 1-universal word $w = \mathtt{abcba}$. For $\pi(\mathtt{a}) = \mathtt{c}$, $\pi(\mathtt{b}) = \mathtt{b}$, and $\pi(\mathtt{c}) = \mathtt{a}$ we obtain $w\pi(w) = \mathtt{abc.bac.babc.}$ which is 3-universal. However, for the identity id on Σ we get that $w\,\mathrm{id}(w)$ is 2-universal. We can show exactly the case when $\iota(w\pi(w)) = 2\iota(w) + 1$.

3.2.12 Proposition. *Let $\pi : \Sigma^* \to \Sigma^*$ be a morphic permutation and $w \in \Sigma^*$ with the arch factorisation $w = \mathrm{ar}_w(1) \cdots \mathrm{ar}_w(k) r(w)$ and $\pi(w)^R = \mathrm{ar}_{\pi(w)^R}(1) \cdots \mathrm{ar}_{\pi(w)^R}(k) r(\pi(w)^R)$ for an appropriate $k \in \mathbb{N}_0$. Then $\iota(w\pi(w))$ is $2\iota(w) + 1$ iff alph$(r(w)\pi(r(w)))) = \Sigma$, i.e., the both rests together are 1-universal.*

Proof. First, consider that $r(w)r(\pi(w)^R)$ is 1-universal. Then we get
$$w\pi(w) = \mathrm{ar}_w(1)\cdots\mathrm{ar}_w(k)r(w).r(\pi(w)^R)^R(\mathrm{ar}_{\pi(w)^R}(k))^R\cdots(\mathrm{ar}_{\pi(w)^R}(1))^R.$$
Since all arches are 1-universal by definition, the assumption implies that $w\pi(w)$ is $(2\iota(w)+1)$-universal and, thus, $\iota(w\pi(w)) \geqslant 2\iota(w)+1$. The equality follows by the definition of ι. For the other direction assume $\iota(w\pi(w)) = 2\iota(w)+1$. Here, we get the arch factorisation
$$w\pi(w) = \mathrm{ar}_{w\pi(w)}(1)\cdots\mathrm{ar}_{w\pi(w)}(2\iota(w)+1)r(w\pi(w)).$$
This implies
$$\begin{aligned}&\mathrm{ar}_{w\pi(w)}(1)\cdots\mathrm{ar}_{w\pi(w)}(2\iota(w)+1)r(w\pi(w))\\&\quad= \mathrm{ar}_w(1)\cdots\mathrm{ar}_w(k)r(w)r(\pi(w)^R)^R(\mathrm{ar}_{\pi(w)^R}(k))^R\cdots(\mathrm{ar}_{\pi(w)^R}(1))^R.\end{aligned}$$
By $\iota(w) = k$, only the first k arches can be contained in w. This implies that $r(w)$ is a prefix of $\mathrm{ar}_{w\pi(w)}(k+1)$. Choose $y \in \Sigma^+$ with $\mathrm{ar}_{w\pi(w)}(k+1) = r(w)y$. By Lemma 3.2.11, we have $\mathrm{ar}_w(i) = \pi(\mathrm{ar}_{\pi(w)}(i))$ and, thus, $y = r(\pi(w)^R))^R$. By $\mathrm{ar}_{w\pi(w)}(k+1) = \Sigma$, the claim is proven. □

Proposition 3.2.12 ensures that, for a given word with a non-empty rest, we can raise the universality-index of $w\pi(w)$ by one if π is chosen accordingly.

3.2.13 Remark. Appending permutations of the word instead of its images under permutations of the alphabet, i.e., appending to w abelian equivalent words, does not lead to immediate results as the universality depends heavily on the permutation. If w is k-universal, a permutation π may arrange the letters in lexicographical order, so $\pi(w)$ would only be 1-universal. On the other hand, the universality can be increased by sorting the letters in 1-universal factors: $\mathtt{a}_1^m\mathtt{a}_2^m\cdots\mathtt{a}_{|\Sigma|}^m$ for $\Sigma = \{\mathtt{a}_1,\ldots,\mathtt{a}_{|\Sigma|}\}$ is 1-universal but $(\mathtt{a}_1\cdots\mathtt{a}_{|\Sigma|})^m$ is m-universal, for $m \in \mathbb{N}$.

In the rest of this section we present results regarding circular universality. Recall that a word w is k-circular universal if a conjugate of w is k-universal. Consider $\Sigma = \{\mathtt{a},\mathtt{b},\mathtt{c},\mathtt{d}\}$ and $w = \mathtt{abbccdabacdbdc}$. Note that w is not 3-universal ($\mathtt{dda} \notin \mathrm{ScatFact}_3(w)$) but 2-universal. Moreover, the conjugate $\mathtt{bbccdabacdbdca}$ of w is 3-universal; accordingly, w is 3-circular universal.

3.2.14 Lemma. *Let $w \in \Sigma^*$. If $\iota(w) = k \in \mathbb{N}$ then $k \leqslant \zeta(w) \leqslant k+1$. Moreover, if $\zeta(w) = k+1$ then $\iota(w) \geqslant k$.*

Proof. Since w is a conjugate of itself, w is at least k-circular universal. Suppose $\zeta(w) = k+2$. Choose $x, y \in \Sigma^*$ with $w = xy$ and $yx = \mathrm{ar}_{yx}(1) \cdots \mathrm{ar}_{yx}(k+2) r(yx)$. Since $\iota(w) = k$ there is no i such that $y = w_1 \cdots w_i$ (otherwise $w = xy$ would be $(k+1)$-universal). Thus, there exists a $j \in [k+2]$ and a proper prefix y_1 of w_j such that $y = w_1 \cdots w_{j-1} y_1$; let x_1 be such that $w_j = y_1 x_1$. This implies $w = xy = x_1 w_{j+1} \cdots w_{k+2} w_1 \cdots w_j y_1$ and we get that $k+1$ arches are contained in w. This contradicts the maximality of k.

For the second claim let $w = xy$ and $yx = \mathrm{ar}_{yx}(1) \cdots \mathrm{ar}_{yx}(k+1) r(yx)$. If y contains all arches then $\iota(w) = k+1$. If y does not contain all arches, there exists an $i \in [k+1]$ such that a prefix of $\mathrm{ar}_{yx}(i)$ is a suffix of y and the corresponding suffix of $\mathrm{ar}_{yx}(i)$ is a prefix of x. Thus, $\mathrm{ar}_{yx}(1) \cdots \mathrm{ar}_{yx}(i-1)\, \mathrm{ar}_{yx}(i+1)\, \mathrm{ar}_{yx}(k+1)$ is a scattered factor of w. □

3.2.15 Lemma. *Let $w \in \Sigma^+$. If $\iota(w) = k$ and $\zeta(w) = k+1$ then there exist $v, z, u \in \Sigma^*$ such that $w = vzu$, with $u, v \neq \varepsilon$ and $\iota(z) = k$.*

Proof. By $\zeta(w) = k+1$, there exist $x, y \in \Sigma^*$ with $w = xy$ and $yx = \mathrm{ar}_{yx}(1) \cdots\ \mathrm{ar}_{yx}(k+1) r(yx)$. Since $\iota(w) = k$ there is no i such that $y = w_1 \cdots w_i$ (otherwise $w = xy$ would be $(k+1)$-universal). Thus, there exists $i \in [k+1]_0$ with $w_{i+1} = uv$ and u is a proper and non-empty suffix of y and v is a proper and non-empty prefix of x with $\mathrm{alph}(u), \mathrm{alph}(v) \subset \Sigma$. This implies

$$yw = xy = v\, \mathrm{ar}_w(i+2) \cdots \mathrm{ar}_w(k+1)\, \mathrm{ar}_w(1) \cdots \mathrm{ar}_w(i) u.$$

Let $z = \mathrm{ar}_w(i+2) \cdots \mathrm{ar}_w(k+1)\, \mathrm{ar}_w(1) \cdots \mathrm{ar}_w(i)$. Clearly, z contains 1-universal words, so $\iota(z) \geqslant k$. By $\iota(w) = k$, it follows immediately $\iota(z) \leqslant k$. □

The following theorem connects the circular universality index of a word with the universality index of the repetitions of that word.

3.2.16 Theorem. *Let $w \in \Sigma^*$. If $\iota(w) = k$ and $\zeta(w) = k+1$ then $\iota(w^s) = sk + s - 1$, for all $s \in \mathbb{N}$.*

Proof. By Lemma 3.2.15, there exist $v, z, u \in \Sigma^*$ with $w = vzu$, $\iota(z) = k$, and $\mathrm{alph}(v)$, $\mathrm{alph}(u) \subset \Sigma$. Consequently, we have that

$$w^s = (vzu)^s = v(zuv)^{s-1}zu$$

is $((s-1)(k+1)+k)$-universal, thus, $\iota(w^s) \geqslant (sk+s-1)$. Since $\iota(w) = k$ and w^s only contains $s-1$ transitions from one w to another, w^s cannot have a higher universality. □

The other direction of Theorem 3.2.16 does not hold for arbitrary alphabets: Consider the 2-universal word $w = \mathtt{babccaabc}$. We have that w^2 is 5-universal but w is not 3-circular universal. Nevertheless, Lemma 3.2.15 helps us to show that the converse of Theorem 3.2.16 holds for binary alphabets:

3.2.17 Theorem. *Let $w \in \{\mathtt{a},\mathtt{b}\}^*$ with $\iota(w) = k$ and $s \in \mathbb{N}$. Then $\iota(w^s) = sk+s-1$ if $\zeta(w) = k+1$ and sk otherwise.*

Proof. By Theorem 3.2.16, it suffices to prove $\zeta(w) = k+1$ if w^s is $(sk+1)$-universal. Assume $\iota(w) \geqslant sk+1$. If for all conjugates v of w we have $v[1] \neq v[|w|]$ then w is of even length and we have $w = (\mathtt{ab})^k$ or $w = (\mathtt{ba})^k$; this implies immeditaly $\zeta(w) = k$ by the arch factorisation. Thus, we know that there exists a conjugate v of w with $v[1] = v[|w|]$. Since w^s is a conjugate of v^s and w^s is $(sk+1)$-universal, v^s is $(sk+1)$-circular universal. By Lemma 3.2.14, we get that v^s is (sk)-universal and, by Theorem 3.2.16, it follows that v^{2s} is $(2sk+1)$-universal. By [26, Theorem4], v^{2s} contains $2sk+1$ disjoint occurrences of $\mathtt{ab}$ or $\mathtt{ba}$. By $v[1] = v[n]$, none of these occurrences can start in one v and end in the following. This implies that one v contains $k+1$ of these occurrences and therefore $\iota(v) \geqslant k+1$. Hence, we get $\zeta(w) = k+1$. □

3.2.2 Algorithmic Results

In this section we present algorithms answering the questions that are most natural to us. The questions regard: is a specific factor v of $w \in \Sigma^*$ universal?, what is the minimal $\ell \in \mathbb{N}$ such that w^ℓ is k-universal for a given $k \in \mathbb{N}$?, how many (and which) words from a given set do we have to concatenate such that the resulting word is k-universal for a given $k \in \mathbb{N}$?, what is the longest (shortest) prefix (suffix) of a word being k-universal for a given $k \in \mathbb{N}$? We introduce in the following lemma some preliminary data structures.

3.2.18 Lemma*. *Given a word $x \in \Sigma^n$ with $alph(x) = \Sigma$, we can compute for all $j \in [n]$ in $O(n)$ and*

- *the shortest* 1*-universal prefix of $x[j..n]$: $u_x[j] = \min\{i \mid x[j..i] \text{ is universal}\}$,*
- *the value $\iota(x[j..n])$: $t_x[j] = \max\{t \mid \mathrm{ScatFact}_t(x[j..n]) = \Sigma^t\}$, and*
- *minimal $\ell \in [n]$ with $\iota(x[j..\ell]) = \iota(x[j..|x|])$: $m_x[j] = \min\{i \mid \mathrm{ScatFact}_{t_x[j]}(x[j..i]) = \Sigma^{t_x[j]}\}$.*

Proof. For each $j \in [n]$ and letter $\mathtt{a} \in \Sigma$, denote $g_{\mathtt{a}}[j] = \min\{i \mid i \geqslant j, w[i] = \mathtt{a}\}$ (by convention, $g_{\mathtt{a}}[j] = +\infty$ if $\mathtt{a}$ does not occur in $x[j..n]$). Clearly, $u_x[j] = \max\{g_{\mathtt{a}}[j] \mid \mathtt{a} \in \Sigma\}$ holds, i.e., $u_x[j]$ is the end position of the shortest word starting at position j in x which contains all letters of Σ (the value $g_{\mathtt{a}}[j]$ is strongly related to the value $X_{\mathtt{a}}(w[j..n])$ - read "next $\mathtt{a}$ in $w[j..n]$"- used in [45] to denote the first position of $\mathtt{a}$ in $w[j..n]$). It is essential to note that we will not compute all the values $g_{\mathtt{a}}[j]$, but only the values $u_x[j]$, for all j. As such, $x[j..u_x[j]]$ is the shortest universal prefix of $x[j..n]$.

Computing the elements of $u_x[\cdot]$ is done as follows: let C be an array with $|\Sigma|$ elements, all initialised to 0. As Σ is considered to be the set of numbers $\{1, \ldots, |\Sigma|\}$, we will consider that C is indexed by the letters of Σ. Also, initialise the variable h with $|\Sigma|$.

While $h > 0$, we consider the positions j of x from the right to the left, i.e., from n downwards. When reading $x[j]$, we set $C[x[j]] = j$, and if $C[x[j]]$ was 0 before setting it to j, then we decrement h by 1. As soon as we have $h = 0$ we stop. At this point we have $C[\mathtt{a}] = g_{\mathtt{a}}[j]$ for all $\mathtt{a} \in \Sigma$, so $C[\mathtt{a}]$ is the leftmost occurrence of $\mathtt{a}$ to the right of j, and $x[j..n]$ is the shortest suffix of x that contains all letters of Σ. We can set $u_x[j'] = +\infty$, for all $j' > j$, and $u_x[j] = \max\{C[\mathtt{a}] \mid \mathtt{a} \in \Sigma\}$.

Now let $m = u_x[j]$, and $d = j + 1$ ($x[d..n]$ is the longest suffix of x which is not universal).

For i from $j - 1$ downto 1, we do the following. If $m \neq C[x[i]]$ (i.e., $x[i]$ is not the same as the letter whose leftmost occurrence in $x[i + 1..n]$ was the rightmost among all letters of Σ), we just set $C[x[i]] = i$. If $m = C[x[i]]$ (i.e., $x[i]$ is the same as the letter whose leftmost occurrence in $x[i+1..n]$ was the rightmost among all letters of Σ), we first set $C[x[i]] = i$ and then we need to recompute m, the maximum of C (the position of the letter whose first occurrence in $x[i..n]$ is the rightmost among all letters). To do this, we decrement m by 1 repeatedly, until it reaches a value p such that $C[x[p]] = p$. At that point, $m = p$ is the leftmost position on which the letter $x[m]$ occurs in $x[i..n]$, and all letters of Σ occur in $x[i..m]$. In this way, we ensure that $C[\mathtt{a}] = g_{\mathtt{a}}[i]$ for all $\mathtt{a} \in \Sigma$ and m points to the maximum element of C. In both cases, we set $u_x[i] = m$, and repeat the process for $i - 1$.

At the end of the computation described above, we computed $u_x[j]$ for every position j of x, i.e., we know for each position j of x the shortest universal prefix of $x[j..n]$. The computation described above runs in time $O(n)$. For each value j we set $C[x[j]]$ in constant time and then, if needed, recompute the value of m; this last part is not carried in constant time for each j, but in total m traverses only once the entire word x from right to left, so, summing the time spent to update m for all values of j, we still get $O(n)$ time in total.

We now move on to the main phase of our algorithm. For $j \in [n]$, we want to compute $t_x[j] = \max\{t \mid \mathrm{ScatFact}_t(x[j..n]) = \Sigma^t\}$ and $m_x[j] = \min\{i \mid \mathrm{ScatFact}_{t_x[j]}(x[j..i]) = \Sigma^{t_x[j]}\}$.

We show how to compute $m_x[j]$ and $t_x[j]$ for all positions j of x, in $O(n)$ total time, by a simple dynamic programming algorithm. For $j \geqslant d$, we have $t_x[j] = 0$ and $m_x[j] = u_x[j]$. For smaller values of j, we have $m_x[j] = u_x[j] + m_x[u_x[j] + 1]$ and $t_x[j] = 1 + t_x[u_x[j] + 1]$. Indeed, the maximum exponent $t_x[j]$ such that $\Sigma^{t_x[j]} = \mathrm{ScatFact}_{t_x[j]}(x[j..n])$ is obtained by taking the shortest prefix $x[j..u_x[j]]$ of $x[j..n]$ that contains all letters of Σ, and returning 1 plus the maximum exponent $t_x[u_x[j] + 1]$ such that $\Sigma^{t_x[u_x[j]+1]}$ is included in the set of scattered factors of the suffix $x[u_x[j] + 1..n]$ that follows $x[j..u_x[j]]$. The value $m_x[j]$ is computed according to a similar idea. Clearly, computing $m_x[j]$ and $t_x[j]$ takes constant time for each j, so linear time overall. □

The data structures constructed in Lemma 3.2.18 allow us to test in $O(1)$ time the universality of factors $w[i..j]$ of a given word w, w.r.t. $\mathrm{alph}(w) = \Sigma$:

$w[i..j]$ is Σ-universal iff $j \geqslant u_w[i]$. The combinatorial results of Section 3.2.1 give us an initial idea on how the universality of repetitions of a word relates to the universality of that word: Theorem 3.2.16 shows that in order to compute the minimum s such that w^s is ℓ-universal, for a given *binary* word w and a number ℓ, can be reduced to computing the circular universality of w. Unfortunately, this is not the case for all alphabets, as also shown in Section 3.2.1. However, this number s can be computed efficiently, for input words over alphabets of all sizes. While the main idea for binary alphabets was to analyse the universality index of the conjugates of w (i.e., factors of length $|w|$ of ww), in the general case we can analyse the universality index of the suffixes of ww, by constructing the data structures of Lemma 3.2.18 for $x = ww$. The problem is then reduced to solving an equation over integers in order to identify the smallest ℓ such that w^ℓ is k-universal.

3.2.19 Proposition*. *Given a word $w \in \Sigma^n$ with $alph(w) = \Sigma$ and $k \in \mathbb{N}$, we can compute the minimal ℓ such that w^ℓ is k-universal in $O(n + \frac{\log k}{\log n})$ time.*

Proof. Consider the word $x = ww$. In a preprocessing phase, using Lemma 3.2.18, we compute in $O(|x|) = O(n)$ time the values $t_x[j]$ and $m_x[j]$ for $j \in [2n]$.

We want to compute the minimum ℓ such that w^ℓ is k-universal. The general idea is the following: for $p \geqslant 1$, we compute the largest value i_p such that $\Sigma^{i_p} = \text{ScatFact}_{i_p}(w^p)$ as well as the shortest prefix $w^{p-1}w[1..s_p]$ of w^p which is i_p-universal (as each w contains all letters of Σ, it is clear that the shortest prefix of w^p which is i_p-universal must extend inside the p^{th} w). These values can be computed for a certain p using the corresponding values for $p-1$ and the arrays we constructed in the preprocessing phase: $i_p = i_{p-1} + t_x[s_{p-1}]$ and $s_p = s_{p-1} + m_x[s_{p-1}] - n$. Essentially, for each p, we just extend to the right in w^p, as much as we can, the shortest prefix with the desired property constructed for w^{p-1}. In a simple version of our algorithm we could do that until $i_p \geqslant k$ (which happens after at most k iterations), and return p as the value we are searching for. However, this would lead to an algorithm with running time $O(n + \ell \log k / \log n)$ (where the $\log k / \log n$ factor comes from the fact that the operands in each addition $i_p = i_{p-1} + t_x[s_{p-1}]$ may have up to $\log k$ digits). As $\ell \leqslant k$

and it is natural to assume that k is given in its binary representation, this algorithm could be exponential in the worst case.

We can optimise the idea above to work faster by exploiting the periodicity that occurs in the sequence $(s_p)_{p\in\mathbb{N}}$, defined for the repetitions of a word w. By the pigeonhole principle, there always exist $p_1, p_2 \leqslant n+1$ such that $s_{p_1} = s_{p_2}$. So, while $p \leqslant n+1$ we compute i_p and s_p, as above, but keep track of the values taken by s_p and stop this loop as soon as the current s_p has the same value as some previously computed $s_{p'}$ or $i_p \geqslant k$ (in the latter case, we proceed as above and return p as the value ℓ we look for). More precisely, we use an array S with n elements, all set initially to 0. After computing s_p, if $S[s_p] = 0$ then we set $S[s_p] = p$; if $S[s_p] \neq 0$ we proceed as follows. We stop the loop and compute two values $p_1 = S[s_p]$ and $p_2 = p$. It is immediate that p_2 is the smallest p such $s_{p_1} = s_p$ and there are no other $p, p' < p_2$ such that $s_p = s_{p'}$. Computing p_1 and p_2 takes $O(n)$ time. Note that all arithmetic operations we did so far are done on numbers that fit in constant memory.

Assume now that we have computed $p_2 = p_1 + \delta$ and $i_{p_2} = i_{p_1} + d$. It is clear that, for all $j \geqslant 0$, we have $s_{p_1+j\delta} = s_{p_1}$ and $i_{p_1+j\delta} = i_{p_1} + jd$. Now, let $m = k - i_{p_1}$ and $g = \lfloor \frac{m}{d} \rfloor$. Computing these numbers takes $O(\log k/\log n)$ time.

Let $p_3 = p_1 + g\delta$ (again, we need $O(\log k/\log n)$ time to compute p_3). We have $s_{p_3} = s_{p_1}$ and $i_{p_3} = i_{p_1} + gd \leqslant k$ (these operations take $O(\log k/\log n)$ time). Also, $i_{p_3+d} > k$. Let $z = k - i_{p_3}$ (and we have $z \leqslant d$). So, for p from p_3 to $p_3 + \delta$, we proceed as follows. If $i_p - i_{p_3} \geqslant z$ (i.e., $i_p \geqslant k$), return p as the value ℓ we search for. Otherwise, compute $i_{p+1} - i_{p_3} = (i_p - i_{p_3}) + t_{s_p}$ (in time $O(1)$ as it can be done with only adding numbers which are smaller than d) and $s_{p+1} = s_p + m_{s_p} - n$, and iterate. Because we certainly reach, in this loop, a p such that $i_p \geqslant k$, and $\delta \leqslant n$, the execution of the loop takes $O(n)$ time.

Hence, we get the smallest ℓ such that w^ℓ is k-universal (i.e., $i_\ell \geqslant k$), in $O(n + \log k/\log n)$ time. □

We can extend the previous result to the more general (but less motivated) case of arbitrary concatenations of words from a given set, not just repetitions of the same word. The following preliminary results are obtained. In all cases we give the number of steps of the algorithms, including arithmetic operations on $\log k$-bit numbers; the time complexities of these algorithms are obtained by multiplying these numbers by $O(\frac{\log k}{\log n})$.

1. Given $k \in \mathbb{N}$ and the words $w_1, \ldots, w_p \in \Sigma^*$ with $|w_1 \cdots w_p| = n$ and $\mathrm{alph}(w_1 \cdots w_p) = \Sigma$, we can compute the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ such that $w_{i_1} \cdots w_{i_\ell}$ is k-universal in $O(2^{3|\Sigma|} p^2 \log \ell + n)$ steps.

2. Given $k \in \mathbb{N}$ and $w_1, \ldots, w_p \in \{\mathtt{a}, \mathtt{b}\}^*$ with $\mathrm{alph}(w_1 \cdots w_p) = \{\mathtt{a}, \mathtt{b}\}$ and $|w_1 \cdots w_p| = n$, we can compute the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ such that $w_{i_1} \cdots w_{i_\ell}$ is k-universal in $O(n + \log \ell)$ steps.

3. Given $k \in \mathbb{N}$ and $w_1, \ldots, w_p \in \Sigma^*$ with $\mathrm{alph}(w_i) = \Sigma$ for all $i \in [p]$ and $|w_1 \cdots w_p| = n$, we can compute in $O(n + p^3 |\Sigma| \log \ell)$ steps the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ with $w_{i_1} \cdots w_{i_\ell}$ is k-universal.

Recall, for $\ell, n \in \mathbb{N}$ and $w_1, \ldots, w_n \in \Sigma^*$, the definitions $\langle w_1, \ldots, w_n \rangle_\ell$ as the set of all words $w = x_1 \cdots x_\ell$ with $x_i \in \{w_1, \ldots, w_n\}$ and $\langle w_1, \ldots, w_n \rangle = \bigcup_{\ell \in \mathbb{N}} \langle w_1, \ldots, w_n \rangle_\ell$.

3.2.20 Definition. Let $n \in \mathbb{N}$. The set $S = \{w_1, \ldots, w_n | \, w_i \in \Sigma^*, i \in [n]\}$ is *k-universal* if there exists $u \in \langle w_1, \ldots, w_n \rangle$ such that u is k-universal.

First, we need to introduce some notation for convenience and to prove an auxiliary lemma. To each $S \subseteq \Sigma$ we associate a word u_S with $|u_S| = |S|$ and $\mathrm{alph}(u_S) = S$ (i.e., u_S is a linear ordering of the letters from S). Following the notations from Lemma 3.2.18, for a word x, let $t_x = \max\{t \in \mathbb{N}_0 \mid \mathrm{ScatFact}_t(x) = \Sigma^t\}$ and $m_x = \min\{i \in \mathbb{N}_0 \mid \mathrm{ScatFact}_{t_x}(x[1..i]) = \Sigma^{t_x}\}$; clearly, if $t_x = 0$, then $m_x = 0$, too. Note now that, for a word u with $\mathrm{alph}(u) = S$ and $|u| = |S|$, we have $t_{u_S w} = t_{uw}$ and $m_{u_S w} = m_{uw}$, for all $w \in \Sigma^*$. Consider $w_1, \ldots, w_p \in \Sigma^*$, and take $j \in [p]$. For $\ell \in \mathbb{N}$ and $S' \subset \Sigma$, we define $\max_\ell(S, j, S') = \max\{t_w \mid w = u_S w' w_j, w' \in \langle w_1, \ldots, w_p \rangle_{\ell - 1}$ and $\mathrm{alph}(w[m_w + 1..|w|]) = S'\}$. By the remarks regarding the choice of the word u_S, $\max_\ell(S, j, S')$ is clearly well defined.

3.2.21 Lemma*. *For $w_1, \ldots, w_p \in \Sigma^*$, $S \subseteq \Sigma$, $\ell \in \mathbb{N}_{\geqslant 2}$, and all $\ell' \in [\ell - 1]$, we have* $\max_\ell(S, j, S') = \max\{\max_{\ell'}(S, k, S'') + \max_{\ell - \ell'}(S'', j, S')) \mid k \in [p], S'' \subseteq \Sigma\}$.

Proof. Let ℓ' be a natural number such that $1 \leqslant \ell' < \ell$. Let $i_1, \ldots, i_\ell \in [p]$ such that $i_\ell = j$, $\max_\ell(S, j, S') = t_w$, and $\mathrm{alph}(w[m_w + 1..|w|]) = S'$, for $w = u_S w_{i_1} \cdots w_{i_\ell}$. Take $x' = u_S w_{i_1} \cdots w_{i_{\ell'}}$, $S'' = \mathrm{alph}(x'[m_{x'}..|x'|])$, and $x'' = u_{S''} w_{i_{\ell'+1}} \cdots w_{i_\ell}$. It is not hard to see that $\max_\ell(S, j, S') = t_{x'} + t_{x''}$.

Assume that $\max_{\ell'}(S, i_{\ell'}, S'') > t_{x'}$. Let $h_1, \ldots, h_{\ell'} \in [p]$ and $w_{h_1}, \ldots, w_{h_{\ell'}} \in \Sigma^*$ with $h_{\ell'} = i_{\ell'}$, $\max_{\ell'}(S, h_{\ell'}, S'') = t_{v'}$, and for $v' = u_S w_{h_1} \cdots w_{h_{\ell'}}$ there holds $\mathrm{alph}(v'[m_{v'} + 1..|v'|]) = S'$. Then, for

$$v'' = u_S w_{h_1} \cdots w_{h_{\ell'}} w_{i_{\ell'+1}} \cdots w_{i_\ell}$$

we have $t_{v''} > t_w = \max_\ell(S, j, S')$ - a contradiction. Consequently, we have $\max_{\ell'}(S, i_{\ell'}, S'') = t_{x'}$. We can similarly show that $\max_{\ell - \ell'}(S'', j, S') = t_{x''}$.

Assume now that there exists $r \in [p]$ and $T \subseteq \Sigma$ with $\max_{\ell'}(S, r, T) + \max_{\ell - \ell'}(T, j, S') > t_{x'} + t_{x''} = t_w$. Let $j_1, \ldots, j_\ell \in [p]$ and $w_{j_1}, \ldots, w_{j_\ell} \in \Sigma^*$ such that $j_{\ell'} = r$, $j_\ell = j$, $\max_{\ell'}(S, j_{\ell'}, T) = t_x$ and $\mathrm{alph}(x[m_x + 1..|x|]) = T$, for $x = u_S w_{j_1} \cdots w_{j_{\ell'}}$, and $\max_\ell(T, j_{\ell'}, S') = t_y$ and $\mathrm{alph}(y[m_y + 1..|y|]) = S'$, for $y = u_T w_{j_{\ell'+1}} \cdots x_{j_\ell}$. Then, clearly, for $v = u_S x_{j_1} \cdots x_{j_\ell}$ we have $t_v > t_w = \max_\ell(S, j, S')$, a contradiction to the form of v. □

3.2.22 Theorem*. *Given $k \in \mathbb{N}$ and $w_1, \ldots, w_p \in \Sigma^*$ with $|w_1 \cdots w_p| = n$ and alph$(w_1 \cdots w_p) = \Sigma$, we can compute the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ such that $w_{i_1} \cdots w_{i_\ell}$ is k-universal in $O(2^{3|\Sigma|} p^2 \log \ell + n)$ steps, some being arithmetic operations on numbers with* $\log k$ *bits. The overall time complexity of our algorithm is $O(\frac{\log k}{\log n}(2^{3|\Sigma|} p^2 \log \ell + n))$.*

Proof. Note first that, because $\Sigma = \mathrm{alph}(w_1 \cdots w_p)$, we have $\ell \leqslant pk$. We shall first sketch the algorithm computing ℓ. The general idea is first to construct, by dynamic programming, concatenations of 2^e factors of the set $\{w_1, \ldots, w_p\}$, for larger and larger e, until we find one such concatenation with 2^f elements that is k'-universal, for some $k' \geqslant k$. That is, we compute the values $N_e[S, S', j] = \max_{2^e}(S, i, S')$, for e from 0 until we reach an array N_f which contains a value $N_f[\varnothing, S', j] \geqslant k$. Note that 2^f is the smallest power of 2 such that the concatenation of 2^f numbers is k-universal, so, consequently, $f \leqslant 2\ell$. The values in each of the arrays N_e are computed by dynamic programming based on the values in the array N_{e-1}, using Lemma 3.2.21 for $\ell = 2^e$ and $\ell' = 2^{e-1}$. Once this computation is completed, we use binary search to obtain the exact value of ℓ, as required.

However, we now have the benefit that we can perform this binary search in an interval upper bounded by $2^f \in O(\ell)$.

In the following we describe the algorithm in detail. We will evaluate its complexity first as the number of steps (including arithmetic operations on numbers with up to $\log k$ bits) it performs. Then we compute its actual time complexity.

We start with a preprocessing phase. We construct the $p \times |\Sigma|$ matrix $F[\cdot,\cdot]$, indexed by the numbers between 1 and p and the letters of Σ (which are numbers between 1 and $|\Sigma|$). We have that $F[i,x]$ is the position of the first (i.e., leftmost) occurrence of each letter $x \in \Sigma$ in w_i. This matrix can be computed as follows. Initialise all elements of F with 0. For each i, we traverse w_i, letter by letter, from left to right. When the letter $x \in \Sigma$ is read at position j of w_i, if $F[i,x] = 0$ then we set $F[i,x] = j$. The total number of steps needed to do this is $O(|\Sigma|p + n)$ (as it includes the initialisation of F). Similarly, we construct the $p \times |\Sigma|$ matrix $L[\cdot,\cdot]$, indexed by the numbers between 1 and p and the letters of Σ, where $L[i,x]$ is the position of the rightmost occurrence of each letter $x \in \Sigma$ in w_i.

Also in the preprocessing phase, we compute the data structures from Lemma 3.2.18, for each word w_i, with $i \in [p]$. So, we have for each word w_i the arrays $t_{w_i}[j] = \max\{t \mid \text{ScatFact}_t(w_i[j..n]) = \Sigma^t\}$ and $m_{w_i}[j] = \min\{g \mid \text{ScatFact}_{t_{w_i}[j]}(w_i[j..g]) = \Sigma^{t_{w_i}[j]}\}$. This is done in $O(n)$ steps.

Then, for each set $S \subseteq \Sigma$ and $i \in [p]$, we compute in $O(|\Sigma|)$, the value $j = \max\{F[x,i] \mid x \in \Sigma \backslash S\}$. Basically, $w_i[1..j]$ is the shortest prefix of w_i such that $u_S w_i$ contains all letters of Σ. Let $g = m_{w_i}[j+1]$, and let $S' \subseteq \Sigma$ be the set of letters contained by $w_i[g+1..|w_i|]$. The set S' can be computed in $O(|\Sigma|)$ time, by selecting in S' the letters $x \in \Sigma$ with $L[i,x] > g$. We set $M_1[S,i] = (1 + t_{w_i}[j], S')$, where M_1 is an additional matrix we use. The computation of $M_1[S,i]$, performed for a set S and a number $i \in [p]$, takes $O(|\Sigma|)$ time. So, in total we compute the matrix M_1 in $O(2^{|\Sigma|}|\Sigma|p)$ time. It is worth noting that if $M_1[S,i] = (h, S')$, then $\max_1(S,i,S') = h$.

The main phase of the algorithm follows. If there is an element $M_1[\emptyset,i] = (h, S')$ such that $h \geqslant k$, then we return $\ell = 1$. If not we proceed as described next.

For e natural number such that $\log(pk) + 1 \geqslant e \geqslant 1$, we define the 3-dimensional array $N_e[\cdot,\cdot,\cdot]$, whose first two indices are subsets of Σ, and the third is a number from $[p]$, and $N_e[S,S',i] = \max_{2^e}(S,i,S')$. That is, $N_e[S,S',i]$ stores the maximum k such that there exists a k-universal word w which is the concatenation of u_S followed by 2^e words from $\{w_1,\ldots,w_p\}$, ending with w_i, and, moreover, if w' is the suffix of w that follows the shortest k-universal prefix of w, then $\mathrm{alph}(w') = S'$. The elements $N_e[S,S',i]$ will be computed by dynamic programming, using Lemma 3.2.21 for $\ell = 2^e$ and $\ell' = \frac{\ell}{2}$.

For $e = 1$, the elements of the array N_e are computed as follows. By Lemma 3.2.21, $N_1[S,S',i] = \max\{g \mid g = g_1 + g_2$ where $M_1[S,j] = (g_1,S'')$ and $M_1[S'',i] = (g_2,S')$, with $j \in [p], S'' \subseteq \Sigma\}$. For $e > 1$, we have $N_e[S,S',i] = \max\{g \mid g = g_1 + g_2$ where $N_{e-1}[S,S'',j] = g_1$ and $N_{e-1}[S'',S',i] = g_2$, with $j \in [p], S'' \subseteq \Sigma\}$. Clearly, computing each of the arrays N_e takes $O(2^{3|\Sigma|}p^2)$. It is not hard to see that the maximum element of N_e is strictly greater than the maximum element of N_{e-1}.

We stop the computation of the arrays N_e as soon as we reach such array N_f such that there exists i and S' with $N[\emptyset,S',i] \geqslant k$. We get that $2^{f-1} < \ell \leqslant 2^f$ (where ℓ is the value we want to compute), so the total time needed to compute all these arrays is $O(2^{3|\Sigma|}p^2 \log \ell)$.

Now we need to search for ℓ between $b = 2^{f-1}$ and $s = 2^f$. We will do this by an adapted binary search. Denote $N' = N_{f-1}$ and $N'' = N_f$. Let h be maximal such that $b + 2^h < s$. We compute the 3-dimensional array $N_{mid}[\cdot,\cdot,\cdot]$, indexed just as the arrays N_e. We have $N_{mid}[S,S',i] = \max\{g \mid g = g_1 + g_2$ where $N'[S,S'',j] = g_1$ and $N_h[S'',S',i] = g_2$, with $j \in [p], S'' \subseteq \Sigma\}$. If N_{mid} contains an element greater or equal to ℓ, we repeat this search for the same b and N', and $s = b + 2^h$ and $N'' = N_{mid}$. Otherwise, we repeat the search for the same s and N'', and using $b + 2^h$ instead of b and N_{mid} instead of N'. We stop the process if $b = s - 1$, and return s. This procedure is iterated $O(f) = O(\log \ell)$ times. Thus, computing ℓ is done in $O(2^{3|\Sigma|}p^2 \log \ell)$ steps, some of which are arithmetic operations on numbers with up to $\log k$ bits.

The overall complexity of the algorithm is, thus, $O(\frac{\log k}{log n}(2^{3|\Sigma|}p^2 \log \ell) + n)$. □

Note that, in the case stated in the previous theorem, computing the minimal number of words (from a given set) that should be concatenated in order to obtain a k-universal word is fixed parameter tractable w.r.t.

the parameter $|\Sigma|$, the size of the alphabet of the input words. If both p, the number of input words, and $|\Sigma|$ are constant, the algorithm runs in $O(n + \log \ell)$ steps, which is linear w.r.t. the size of the input because $\log \ell \leqslant \log(pk) = \log p + \log k$ (but the overall time is still affected by the operations on large numbers). In fact, we can give a solution with a linear number of steps for this problem in the case of words over binary alphabets (and p is not necessarily constant). The main idea is in this case, that we can show that from an input set of words, only a constant number are useful when trying to construct a k-universal word by a minimal number of concatenations. The following result is based on the arch factorisation and Proposition 3.2.3.

3.2.23 Theorem*. *Given $k \in \mathbb{N}$ and $w_1, \ldots, w_p \in \{\mathtt{a}, \mathtt{b}\}^*$ with alph$(w_1 \cdots w_p)$ $= \{\mathtt{a}, \mathtt{b}\}$ and $|w_1 \cdots w_p| = n$, we can compute in $O(n + \log \ell)$ steps the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ such that $w_{i_1} \cdots w_{i_\ell}$ is k-universal. The overall complexity of the algorithm is, thus, $O(\frac{\log k}{log n} \log \ell + n)$.*

Proof. Let $u_0 \in \{w_1, \ldots, w_p\}$ be such that $t_{u_0} \geqslant t_{w_i}$, for all $i \in [p]$. For each $x \in \{\mathtt{a}, \mathtt{b}\}$, let $u_x \in \{w_1, \ldots, w_p\}$ be such that u_x starts with x and $t_{u_x[2..|u_x|]} \geqslant t_{w_i[2..|w_i|]}$, for all $i \in [p]$. For each $x \in \{\mathtt{a}, \mathtt{b}\}$, let $v_x \in \{w_1, \ldots, w_p\}$ be such that v_x ends with x and $t_{v_x[1..|v_x|-1]} \geqslant t_{w_i[1..|w_i|-1]}$, for all $i \in [p]$. For each pair $x, y \in \{\mathtt{a}, \mathtt{b}\}$, let $u_{x,y} \in \{w_1, \ldots, w_p\}$ be such that $u_{x,y}$ starts with x and ends with y and $t_{v_x[2..|v_x|-1]} \geqslant t_{w_i[2..|w_i|-1]}$, for all $i \in [p]$. In case of equalities, we just take any word that fulfils the desired property.

Let $S = \{u_0\} \cup \{u_x \mid x \in \{\mathtt{a}, \mathtt{b}\}\} \cup \{v_x \mid x \in \{\mathtt{a}, \mathtt{b}\}\} \cup \{u_{x,y} \mid x, y \in \{\mathtt{a}, \mathtt{b}\}\}$. Clearly, S contains at most 9 words. Note that all words of S can be computed in $O(n)$ time, using the same strategy as in Proposition 3.2.3.

One can show that for every concatenation of m words from the set $\{w_1, \ldots, w_p\}$ which is k-universal, there exist a concatenation of m words from S which is k'-universal, for some $k' \geqslant k$. Thus, it is enough to solve the problem for the input set S, of constant size, instead of the whole $\{w_1, \ldots, w_p\}$. Hence, by Theorem 3.2.22, the conclusion follows.

Indeed, let $w = w_{i_1} \cdots w_{i_{\ell-1}} w_{i_\ell} w_{i_{\ell+1}} \cdots w_{i_m}$, such that $w_{i_\ell} \notin S$. To compute $t_w = t$ we can proceed as in Proposition 3.2.3 and identify t factors $d_1, \ldots, d_t \in \{\mathtt{ab}, \mathtt{ba}\}$ of w such that $w = s_0 d_1 s_1 \cdots d_t s_t$, where $s_i \in \{\mathtt{a}, \mathtt{b}\}^*$

for $i \in [t]_0$. Clearly, $|\mathrm{alph}(s_i)| \leqslant 1$, for all $i \in [t]_0$. Now, we do a case analysis.

Let $x = w_{i_\ell}[1]$ and $y = w_{i_\ell}[|w_{i_\ell}|]$. If the first letter of w_{i_ℓ} is the last letter of a factor d_i and the last letter of w_{i_ℓ} is the first letter of a factor d_j (with $i < j$), let $w' = w_{i_1} \cdots w_{i_{\ell-1}} u_{x,y} w_{i_{\ell+1}} \cdots w_{i_m}$; it is immediate that $t_{w'} \geqslant t_w$. If the first letter of w_{i_ℓ} is the last letter of some d_i but the last letter of w_{i_ℓ} is not the first letter of any factor d_j (where $j > i$), let $w' = w_{i_1} \cdots w_{i_{\ell-1}} u_x w_{i_{\ell+1}} \cdots w_{i_m}$; it is immediate that $t_{w'} \geqslant t_w$. If the first letter of w_{i_ℓ} is not the last letter of any factor d_i but the last letter of w_{i_ℓ} is the first letter of a factor d_j, let $w' = w_{i_1} \cdots w_{i_{\ell-1}} u_y w_{i_{\ell+1}} \cdots w_{i_m}$; it is immediate that $t_{w'} \geqslant t_w$. Finally, if the first letter of w_{i_ℓ} is not the last letter of any factor d_i and the last letter of w_{i_ℓ} is not the first letter of any factor d_j, let $w' = w_{i_1} \cdots w_{i_{\ell-1}} u_0 w_{i_{\ell+1}} \cdots w_{i_m}$; it is immediate that $t_{w'} \geqslant t_w$.

So, if a concatenation of m words $w_{i_1} \cdots w_{i_m}$ is t-universal, we could iteratively replace all the words which are not part of S by words of S and obtain a concatenation with m input words, which is t'-universal, with $t' \geqslant t$. Therefore, to solve the problem from the statement of the theorem, it is enough to produce the set S and then solve the problem for the input set S instead of $\{w_1, \ldots, w_p\}$. For that we can use the algorithm from Theorem 3.2.22, which will run in $O(n + \frac{\log k \log \ell}{\log n})$ steps, because both S and Σ are constant. □ □

In a particular case of Theorem 3.2.22 each of the input words contains all letters of Σ. Once again, we obtain a polynomial algorithm.

3.2.24 Theorem*. *Given $w_1, \ldots, w_p \in \Sigma^*$, with $\mathrm{alph}(w_i) = \Sigma$ for all $i \in [p]$ and $|w_1 \cdots w_p| = n$, and $k \in \mathbb{N}$, we can compute in polynomial time $O(n + p^3 |\Sigma| \log \ell \frac{\log k}{logn})$ the minimal ℓ for which there exist $\{i_1, \ldots, i_\ell\} \subseteq [k]$ with $w_{i_1} \cdots\ w_{i_\ell}$ is k-universal.*

The proofs of Theorems 3.2.22 and 3.2.24 are based on a common dynamic programming algorithm: for all subsets $S \subset \Sigma$ and increasing values of an integer $\ell > 1$, we compute the maximal universality index m that we can obtain by concatenating 2^t words from the input set such that the respective concatenation consists of a prefix which is m-universal, followed by a suffix over S. We stop as soon as we reach an $m \geqslant k$ as universality index. We then optimise the number of concatenated words needed to obtain the universality index k by binary search. Now, for Theorem 3.2.22 we have to consider all the sets S, in each step, while in the case of Theorem 3.2.24 it is enough to consider only the sets that occur as alphabets of the suffixes of the input words. This is why this strategy can be implemented more efficiently in the case when all input words are universal to begin with.

Proof of Theorem 3.2.24. We follow the idea of the algorithm of Theorem 3.2.22: construct, by dynamic programming, longer and longer concatenations of factors of the set $\{w_1, \dots, w_p\}$, until one such concatenation which is k-universal is obtained. The main difference is that in each concatenation $w = w_{i_1} \cdots w_{i_m}$, the shortest prefix of w which is k-universal ends inside w_{i_m}, because $\text{alph}(w_i) = \Sigma$ for all $i \in [p]$. As such, the ℓ we search for is at most k, but also this allows us to get rid of the exponential dependency on Σ from Theorem 3.2.22, as we can now work with certain suffixes of the words w_i, instead of subsets of Σ, when defining our dynamic programming structures. Informally, our algorithm works as follows: we find the highest universality index of a concatenation of 2^e words of $\{w_1, \dots, w_p\}$, which starts inside w_i and ends inside w_j, for all i and j, and suitable starting and, respectively, ending positions. This can be efficiently computed for several reasons. First, such a concatenation is obtained by putting together two concatenations of roughly 2^{e-1} words of $\{w_1, \dots, w_p\}$ which have the highest universality index, the first starting in the same place within w_i, followed by $2^{e-1} - 2$ words of the input set, and ending with a prefix of length c of some w_q, and the second one starting with $w_q[c+1..|w_q|]$ followed by 2^{e-1} words from the input set, ending in the same place within w_j. Second, a concatenation of 2^e words of $\{w_1, \dots, w_p\}$ with the highest universality index, ending inside w_j, can actually only end on some very specific positions of w_j: the positions where each letter of Σ occurs for the first time in the shortest prefix of w_j that contains all letters of Σ. Consequently, the starting positions of such concatenations (useful in our algorithm either directly as solutions, or as building blocks for larger concatenations) are also restricted. Putting these two ideas together, and using an adapted binary search to find ℓ, we obtain an algorithm with the stated complexity.

Once again, we start with some preliminaries and a preprocessing phase. Let $\sigma = |\Sigma|$.

To begin with, let us consider a concatenation $w = w_{i_1} \cdots w_{i_m}$, and let t be the maximum number such that w is t-universal. By Lemma 3.2.3, we can greedily decompose $w = d_1 \cdots d_t d'$, such that $\mathrm{alph}(d_j) = \Sigma$, $\mathrm{alph}(d')$ is a strict subset of Σ, and $d_1 \cdots d_j$ is the shortest prefix of w which is j-universal, for all $j \in [t]$. Because $\mathrm{alph}(w_i) = \Sigma$ for all i, we have that each factor d_j is either fully contained in one of the words w_{i_g}, for $j \in [t]$ and $g \in [m]$, or it starts inside w_{i_g} and ends inside $w_{i_{g+1}}$, for some $g \in [m]$. In the following, we call a factor d_j crossing if it starts inside w_{i_g} and ends inside $w_{i_{g+1}}$, for some $g \in [m]$. If d_j is such a factor, then d_j can only start on some very specific positions inside w_{i_g}. First, the suffix of w_{i_g} that comes after d_{j-1} cannot contain all letters of Σ. So d_{j-1} must end inside the shortest suffix of w_{i_g} that contains all letters of Σ. Assume this suffix starts at position r and note that it starts with the last occurrence of some letter of Σ in w_{i_g}. So, d_{j-1} ends at a position $r' \geqslant r$. Due to the greedy construction of d_{j-1}, it follows that the last letter of d_{j-1} occurs in $w[r..r']$ exactly once. So, d_{j-1} ends on the first occurrence of a letter of Σ to the right of r. There are at most σ such positions. Consequently, d_j starts exactly at the next position after d_{j-1} ends, and we also have at most σ positions where d_j may start.

In conclusion, in each word w_i, part of a concatenation $w = w_{i_1} \cdots w_{i_m}$, there are at most Σ positions where a crossing factor can start. Each crossing factor d_j is constructed by appending to d_j (in a left to right traversal, from the starting position of the factor) the letters of the considered concatenation until $\Sigma = \mathrm{alph}(d_j)$. Therefore, d_j is uniquely determined by the two factors it crosses (w_{i_g} and $w_{i_{g+1}}$) and its starting position inside w_{i_g}. Hence, there can be at most $O(p^2|\Sigma|)$ crossing factors overall, and we will determine all of them in our preprocessing.

In the preprocessing phase, we construct the $p \times |\Sigma|$ matrices $F[\cdot,\cdot]$ and $L[\cdot,\cdot]$ as in the proof of Theorem 3.2.22. Using $L[i,\cdot]$ we also determine the position r_i of w_i such that $w_i[r_i..|w_i|]$ is the shortest suffix of w_i that contains all letters of Σ. Also, in another traversal of w_i we can compute the increasingly sorted list L_i of positions where each letter of Σ occurs for the first time in $w_i[r_i..|w_i|]$. More precisely, we construct the lists $L_i = (i_1, x_1), \ldots, (i_\sigma, x_\sigma)$, where $i_g < i_{g+1}$ for $g \in \Sigma$, and $\{x_1, \ldots, x_\sigma\} = \Sigma$. The time needed to compute all these structures is $O(n)$.

Now, we compute the factors crossing from w_i to w_j. They should start on one of the positions i_1+1, $i_2+1, \dots, i_\sigma+1$, obtained using L_i. Let $c_{i,j}[i_g+1]$ be the crossing factor that starts at position i_g+1 for some $g \in [\sigma]$. The prefix of $c_{i,j}[i_g+1]$ contained in w_i contains only the letters $x_{g+1}, \dots, x_\sigma$ and none of the letters $x_1, \dots, x_g$. Thus, $c_{i,j}[i_g+1]$ extends in w_j until it contains all the missing letters, i.e., until the maximum position among $F[j,x_1]$, $F[j,x_2], \dots$, $F[j,x_g]$. This observation allows us to compute the respective crossing factors efficiently. Let $C[i,j,g]$ be the last position (inside w_j) of $c_{i,j}[i_g+1]$. Then $C[i,j,1] = F[j,x_1]$. For $g > 1$, $C[i,j,g] = \max\{F[j,x_g], C[i,j,g-1]\}$.

The time needed to compute the values $C[i,j,\cdot]$ is $O(\sigma)$. We do this computation for all i and j, and, as such, we identify the starting and ending positions for all possible crossing factors in $O(p^2\sigma)$.

Still in the preprocessing phase, we compute the data structures from Lemma 3.2.18, for each word w_i, with $i \in [p]$. So, we have for each word w_i the arrays $t_{w_i}[j] = \max\{t \mid \mathrm{ScatFact}_t(w_i[j..n]) = \Sigma^t\}$ and $m_{w_i}[j] = \min\{g \mid \mathrm{ScatFact}_{t_{w_i}[j]}(w_i[j..g]) = \Sigma^{t_{w_i}[j]}\}$. Let $t_{w_i} = t_{w_i}[1]$ and $m_{w_i} = m_{w_i}[1]$. This takes $O(n)$ time.

Further, we present the main phase of our algorithm, that computes the value ℓ for which there exist $\{i_1, \dots, i_\ell\} \subseteq [k]$ such that $w_{i_1} \cdots w_{i_\ell}$ is k-universal.

First, if there exists i such that $t_{w_i} \geqslant k$, we have $\ell = 1$. Otherwise, we continue as follows.

For $e \in [k]$, $e \geqslant 1$, we define the 3-dimensional arrays $R_e[\cdot,\cdot,\cdot]$, whose first and third indices are numbers $i,j \in [p]$, and second index is a number from $\{0\} \cup L_i$ (so each R_e has size $O(p^2\sigma)$). We define $R_e[i,j,c] = (t,d)$ where t is the maximum number for which there exist $i_2, \dots, i_{2^e-1} \in [p]$ such that $t_w = t$, where $w = w_i[c+1..|w_i|]w_{i_2} \cdots w_{i_{2^e-1}} w_j$, and d is the minimum number for which there exist $i_2, \dots, i_{2^e-1}$ such that $t_w = t$, where $w = w_i[c+1..|w_i|]w_{i_2} \cdots w_{i_{e-1}} w_j[1..d]$. That is, $R_e[i,j,c]$ stores, on its first component, the maximum t such that there exists a t-universal word w which is the concatenation of the suffix of w_i that starts at position $c+1$, followed by 2^e-2 words from the set $\{w_1, \dots, w_p\}$, and then followed by w_j. Moreover, $R_e[i,j,c]$ also stores, on its second component, the minimum value m_w obtained for a concatenation $w = w_i[c+1..|w_i|]w_{i_2} \cdots w_{i_{2^e-1}} w_j$

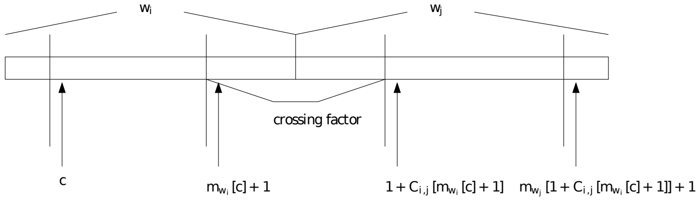
w_i
w_j
crossing factor
c
m_{w_i}[c] + 1
1 + C_{i,j}[m_{w_i}[c] + 1]
m_{w_j}[1 + C_{i,j}[m_{w_i}[c] + 1]] + 1

Figure 3.1. The computation of $R_1[i, j, c-1]$

for which $t_w = t$ (i.e., t_w is as large as possible). We define also the 3-dimensional arrays $P_e[\cdot, \cdot, \cdot]$, exactly as the above with the single difference that in the definition of the elements of P_e we consider the concatenation of $2^e + 1$ elements, not just 2^e as it was the case for R_e.

The elements $R_e[i, j, c]$ and $P_e[i, j, c]$ can be computed by dynamic programming, somehow similarly to the approach of Theorem 3.2.22. To simplify the exposure we also define the array $R_0[\cdot, \cdot, \cdot]$, in which only the elements $R_0[i, i, c-1] = (t_{w_i}[c], m_{w_i}[c])$ are defined (the others are set to $-\infty$). Clearly, R_0 can be computed in $O(p^2\sigma)$.

To describe the general computation, we need to compare pairs of numbers. We say that (a, b) is *more useful* than (c, d) if $a > b$ or $a = b$ and $c \leqslant d$. Also, if $p = (a, b)$ is a pair of natural numbers, then its first projection is $\pi_1(p) = a$ and its second projection is $\pi_2(p) = b$.

To compute R_1 we can use the formula:

$$R_1[i, j, c-1] = \\ (t_{w_i}[c] + 1 + \pi_1(R_0[j, j, C_{i,j}[m_{w_i}[c] + 1]]), m_{w_j}[1 + C_{i,j}[m_{w_i}[c] + 1]]),$$

for $i, j \in [p]$ and $c \in \{0\} \cup L_i$.

Indeed, when computing $R[i, j, c-1]$ we start at position c of w_i and essentially try to identify as many consecutive strings whose alphabet is Σ in the concatenation of w_i and w_j as possible. Using $t_{w_i}[c]$ and $m_{w_i}[c]$ we find the shortest factor $w_i[c..m_{w_i}[c]]$ which has the highest universality index among all factors of w_i starting at position c. Then we use the crossing factor that corresponds to $m_{w_i}[c]$ to move into w_j, at position $c' = C_{i,j}[m_{w_i}[c] + 1]$, and then find the shortest factor $w_j[c'..m_{w_j}[c']]$ which

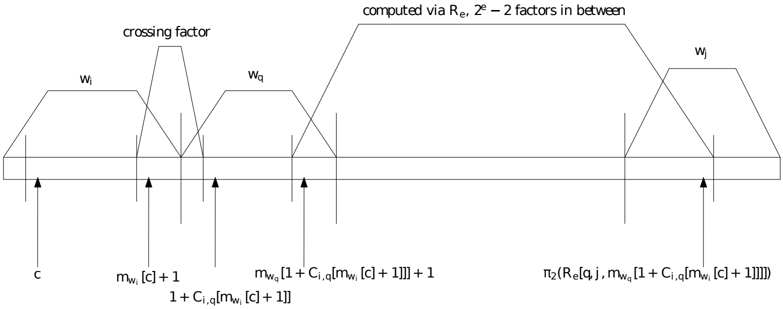
computed via R_e, $2^e - 2$ factors in between
crossing factor
w_j
w_i
w_q
c
$m_{w_i}[c] + 1$
$1 + C_{i,q}[m_{w_i}[c] + 1]]$
$m_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]]] + 1$
$\pi_2(R_e[q, j, m_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]]]])$

Figure 3.2. The computation of $P_e[i, j, c - 1]$

has the highest universality index among all factors of w_j starting at position c'. Following similar arguments to those in the proof of Lemma 3.2.3 we get that R_0 is correctly computed in this way: our strategy here corresponds exactly to the greedy strategy employed in the respective lemma.

After we compute R_e, for some $e \geqslant 1$, we first compute P_e. The formula for the elements of P_e is given in the following. Let $q \in [p]$ be such that $R_e[q, j, 1 + m_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]]]$ is more useful than any other pair $R_e[q', j, 1 + m_{w_{q'}}[1 + C_{i,q'}[m_{w_i}[c] + 1]]]$ for $q' \in [p]$. We then can compute

$$\begin{aligned}P_e[i, j, c - 1] = \\ &(t_{w_i}[c] + 1 + t_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]] \\ &+ \pi_1(R_e[q, j, m_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]]]), \\ &\qquad \pi_2(R_e[q, j, m_{w_q}[1 + C_{i,q}[m_{w_i}[c] + 1]]]),\end{aligned}$$

for $i, j \in [p]$ and $c \in \{0\} \cup L_i$. Similarly to the computation of R_1, when computing $P_e[i, j, c - 1]$ we start at position c of w_i and try to add to $w_i[c..|w_i|]$ a concatenation of 2^e words of $\{w_1, \ldots, w_p\}$ (ending with w_j), which contains as many consecutive strings over the alphabet Σ, as possible. This is done using the greedy approach of Lemma 3.2.3. As such, using $t_{w_i}[c]$ and $m_{w_i}[c]$ we find the shortest factor $w_i[c..m_{w_i}[c]]$ which has the highest universality index among all factors of w_i starting at position c.

Then we identify the word w_q, such that after using the crossing factor that corresponds to $m_{w_i}[c]$ to move from w_i into w_q we can reach w_j by concatenating another $2^e - 2$ factors in between, to obtain a word with the highest universality index among all such possible concatenations. Once again, it is not hard to see that this formula is correct (see also the figure below). First, the choice of the factor $w_i[c..m_{w_i}[c]]$ as the first group of consecutive strings, each with the alphabet Σ, is correct due to the greedy approach in Lemma 3.2.3. Then, we need to cross into the rest of the factors in the concatenation of words from $\{w_1, \ldots, w_p\}$. For each choice $w_{q'}$ of the factor following w_i in this concatenation, we cross into this word from w_i in an optimal way: we use the crossing string ending on $C_{i,q'}[m_{w_i}[c] + 1]$. Any shorter word would not work, any longer word does not make sense due to the greedy strategy of Lemma 3.2.3. Then, using the already computed m_{w_q} and R_e we start from $1 + C_{i,q'}[m_{w_i}[c] + 1]$ and follow the optimal selection of the concatenated strings given by these arrays. We then select from all these possibilities (computed for each q') the one that produces a string with higher universality index. So, the computation of $P_e[i, j, c-1]$ is correct.

After computing P_e for some $e \geqslant 1$, we compute R_{e+1}. For some $i, j \in [p]$ and c with $c \in \{0\} \cup L_i$, let $q \in [p]$ be such that $\pi_1(R_e[i, q, c]) + \pi_1(P_e[q, j, \pi_2(R_e[i, q, c])]) \geqslant \pi_1(R_e[i, q', c]) + \pi_1(P_e[q', j, \pi_2(R_e[i, q', c])])$ for all $q' \in [p]$. To break equalities, we select q with $\pi_2(P_e[q, j, \pi_2(R_e[i, q, c])])$ is minimal. Then, we can compute $R_{e+1}[i, j, c]$ by the following formula

$$\begin{aligned} R_{e+1}[i, j, c-1] = (&\pi_1(R_e[i, q, c-1]) + \pi_1(P_e[q, j, \pi_2(R_e[i, q, c-1])]), \\ &\pi_2(P_e[q, j, \pi_2(R_e[i, q, c-1])])), \end{aligned}$$

for $i, j \in [p]$ and $c \in \{0\} \cup L_i$. The idea is pretty similar to how we computed the other arrays. We start at position $c+1$ of w_i and try to add to $w_i[c+1..|w_i|]$ a concatenation of $2^{e+1} - 1$ words of $\{w_1, \ldots, w_p\}$ (ending with w_j), which contains as many consecutive strings over the alphabet Σ, as possible. We iterate over all possible choices for the 2^e-th word in this concatenation, namely w_q. We use the value computed found in $R_e(i, q, c)$ to find the concatenation of 2^e words with highest universality index that starts with $w_i[c..|w_i|]$ and ends with $w_{q'}$. Then we continue this concatenation again in the best way (i.e., by the concatenation of 2^{e+1} words

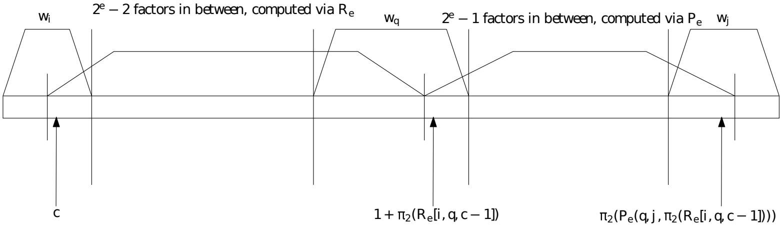
w_i
$2^e - 2$ factors in between, computed via R_e
w_q
$2^e - 1$ factors in between, computed via P_e
w_j
c
$1 + \pi_2(R_e[i, q, c - 1])$
$\pi_2(P_e(q, j, \pi_2(R_e[i, q, c - 1])))$

Figure 3.3. The computation of $R_{e+1}[i, j, c-1]$

with the highest universality index), as given by $P_e[q', j, \pi_2(R_e[i, q', c])]$. Then we just take the value q for which we obtained the most useful pair $(\pi_1(R_e[i, q, c]) + \pi_1(P_e[q, j, \pi_2(R_e[i, q, c])]), \pi_2(P_e[q, j, \pi_2(R_e[i, q, c])]))$. Once more, the greedy approach shown to be correct in Lemma 3.2.3 proves that the formula used for the elements of R_{e+1} is also correct.

Clearly, the complexity of computing each element of P_e and R_e is $O(p)$. So, computing each of these matrices takes $O(p^3\sigma)$.

As in the proof of Theorem 3.2.22, we stop as soon as we computed an array R_f that contains an element $R_f[i, j, 0]$ with $\pi_1(R_f[i, j, 0]) \geqslant k$. We have $f \in O(\log \ell)$.

Now we need to search for ℓ between $b = 2^{f-1}$ and $s = 2^f$. And we can proceed exactly as in the proof of the aforementioned theorem, by an adapted binary search. Set $R' = R_{f-1}$ and $R'' = R_f$ and let h be maximum with $b + 2^h < s$. We compute the 3-dimensional array $R_{mid}[\cdot, \cdot, \cdot]$, indexed just as the arrays R_e. We have

$$\begin{aligned} R_{mid}[i, j, c-1] = \\ &(\pi_1(R'[i, q, c-1]) + \pi_1(P_h[q, j, \pi_2(R'[i, q, c-1])]), \\ &\qquad \pi_2(P_h[q, j, \pi_2(R'[i, q, c-1])])), \end{aligned}$$

for $i, j \in [p]$ and $c \in \{0\} \cup L_i$.

If R_{mid} contains an element whose first component is greater or equal to ℓ, we repeat this search for the same b and R', and $s = b + 2^h$ and $R'' = R_{mid}$. Otherwise, we repeat the search for the same s and R'', and using $b + 2^h$ instead of b and R_{mid} instead of R'. We stop the process if

$b = s - 1$, and return s. This procedure is iterated $O(f) = O(\log \ell)$ times.

The overall number of steps of the algorithm we described is, thus, $O(p^3\sigma \log \ell + n)$. Of course, in the part where we compute concatenations with large universality index we need to manage arithmetic operations with $\log k$-bit numbers. So, our algorithm requires $O(p^3\sigma \log \ell \frac{\log k}{\log n} + n)$ time. □

Finally, we consider the case of decreasing the universality of a word by an operation opposed to concatenation, namely the deletion of a prefix or a suffix.

3.2.25 Theorem*. *Given $w \in \Sigma^n$ with $\iota(w) = m$ and a number $\ell < m$, we can compute in linear time the shortest prefix (resp., suffix) $w[1..i]$ (resp., $w[i..n]$) such that $w[i+1..n]$ (resp., $w[1..i-1]$) has universality index ℓ.*

Proof. To compute the longest prefix $w[1..i-1]$ of w which has universality index ℓ, we use data structures from Lemma 3.2.18. We start with $j = 1$ and $k = 0$. While $k \neq \ell + 1$ do $t = u_w[j]$, increase k, set $j = t + 1$. If $k = \ell + 1$ then $w[1..t]$ is the shortest prefix of w which is $\ell + 1$ universal. Therefore, the longest prefix $w[1..i-1]$ of w which has universality index ℓ is $w[1..t-1]$. A similar approach can be used for suffixes. □

Theorem 3.2.25 allows us to compute which is the shortest prefix (suffix) we should delete so that we get a string of universality index ℓ. Its proof is based on the data structures of Lemma 3.2.18. For instance, to compute the longest prefix $w[1..i-1]$ of w which has universality index ℓ, we identify the first $\ell + 1$ factors of the decomposition of Theorem 3.2.3, assume that their concatenation is $w[1..i]$, and remove the last symbol of this string. A similar approach works for suffixes.

3.2.26 Theorem. *Given a word $w \in \Sigma^n$ with $alph(w) = \{\mathtt{a}, \mathtt{b}\}$ and $k \in \mathbb{N}$, we can compute in $O(n)$ time the minimal number of insertions i needed to be applied to w to reach a k-universal word. Particularly, if ℓ is the universality index of w, we have $i = 0$ if $k \leqslant \ell$, $i = k - \ell$ if $\ell < k \leqslant n - 2\ell$, and $i = 2k - n$ if $n - 2\ell < k$.*

Proof. We show that, if ℓ is the universality index of w, we have that the number i of insertions, that we want to compute, is $i = 0$ if $k \leqslant \ell$, $i = k - \ell$ if $\ell < k \leqslant n - 2\ell$, and $i = 2k - n$ if $n - 2\ell < k$.

Define the mapping $\bar{\cdot} : \Sigma \to \Sigma$ by $\bar{\mathtt{a}} = \mathtt{b}$ and $\bar{\mathtt{b}} = \mathtt{a}$. The claim holds immediately for $k \leqslant \ell$ by Theorem 3.1.3. Assume $k > \ell$. Since $\{\mathtt{ab}, \mathtt{ba}\}^{\ell} \subseteq \mathrm{ScatFact}_{2\ell}(w)$ there exist $w_1, \ldots, w_\ell \in \{\mathtt{ab}, \mathtt{ba}\}$ and $r_1, \ldots, r_\ell \in [n]$ with $w[r_j, r_j + 1] = w_j$ for all $j \in [\ell - 1]$. Notice that by the choice of ℓ we have $r_j + 1 < r_{j+1}$. Set $u_1 = w[1..r_1 - 1]$, $u_{\ell+1} = w[r_\ell + 2..n]$, and $u_j = w[r_{j-1} + 2..r_j - 1]$ with $u_s = \varepsilon$ if the first index is strictly greater than the second, for $2 \leqslant j \leqslant \ell$ and $s \in [\ell]$. This implies $w = u_1 w_1 u_2 \cdots u_\ell w_\ell u_{\ell+1}$ and $u = u_1 \cdots u_{\ell+1}$ is of length $n - 2\ell$. If $n - 2\ell \geqslant k > \ell$ set

$$u' = u[1]\overline{u[1]}u[2]\overline{u[2]} \cdots u_{k-\ell}\overline{u[k-\ell]}u[k-\ell+1] \cdots u[n-2\ell]$$

and u'_s accordingly for $s \in [\ell + 1]$. Then $w' = u'_1 w_1 u'_2 \cdots u'_\ell w_\ell u'_{\ell+1}$ is obtained from w by $k - \ell$ insertions and by the definition of u', we have $\{\mathtt{ab}, \mathtt{ba}\}^{k} \subseteq \mathrm{ScatFact}_{2k}(w')$. This implies that w' is k-universal. If $k > n - 2\ell$ set

$$u' = u[1]\overline{u[1]}u[2]\overline{u[2]} \cdots u_{k-\ell}\overline{u[k-\ell]}u[k-\ell+1]\overline{u[k-\ell+1]} \cdots u[n-2\ell]\overline{u[n-2\ell]}$$

and u'_s accordingly for $s \in [\ell + 1]$. Then $w' = u'_1 w_1 u'_2 \cdots u'_\ell w_\ell u'_\ell (\mathtt{ab})^{k-n+\ell}$ is obtained from w by $n - 2\ell + 2(k - n + \ell) = 2k - n$ insertions. By definition of u' and the appended $(\mathtt{ab})^{k-n+\ell}$, we get $\{\mathtt{ab}, \mathtt{ba}\}^{\ell+n-2\ell+k-n+\ell} = \{\mathtt{ab}, \mathtt{ba}\}^{k}$ is a subset of $\mathrm{ScatFact}_k(w')$ and by Theorem 3.1.3, w' is k-universal. This proves that with i insertions a k-universal word can be obtained from w.

We prove now that i is minimal. Suppose that i is not minimal and let $i' < i$ be the minimal number of insertions such that the obtained word w' is k-universal. By Theorem 3.1.3, we have $\{\mathtt{ab}, \mathtt{ba}\}^{k} \subseteq \mathrm{ScatFact}_{2k}(w')$ and there exist $w_1, \ldots, w_k \in \{\mathtt{ab}, \mathtt{ba}\}$ such that $w_1 \cdots w_k$ is a scattered factor of w'. Let j'' be the number of w_s which were inserted completely and j' be the number of w_s in which one letter was already in w and one is inserted, i.e., $i' = j' + 2j''$. This implies $k \leqslant \ell + j' + j''$. By the first part of the proof, we have $k = i + \ell$. If $\ell < k \leqslant n - 2\ell$ we get $i' + \ell < i + \ell = k \leqslant \ell + j' + j''$ and, thus, $i' < j' + j''$ which contradicts $i' = j' + 2j''$. If $k > n - 2\ell$ we get with $n + i' \geqslant 2\ell + 2j'' + 2j'$ (for w''s length)

$$2(\ell + j' + j'') \leqslant n + i' < n + i = 2k = 2(\ell + j' + j'').$$

Hence, i is minimal.

By Theorem 3.2.3, the decomposition into $w_1, \dots, w_\ell$ can be found in time $\mathcal{O}(n)$. The word u'_s for $s \in [\ell+1]$ can be constructed by going once from left to right while inserting after each letter x the *opposite* letter $\overline{x}$ until $k-\ell$ insertions are reached. If $k > n-2\ell$ for each letter a new one is inserted and the remaining $(\mathtt{ab})^{k-n+\ell}$ letters are simply appended. □

We complete this section with an estimation of the shortest length of a word $w \in \Sigma^*$ such that $\iota(w) = k$ for a given $k \in \mathbb{N}$, i.e., how many equally distributed letters have to be chosen such that the generated word is k-universal with probability at least $\frac{1}{2}$. This problem is related to the *Coupon Collector Problem* [94] (for an overview see [38]): how many packages of coupons have to be bought before each coupon was seen at least once. In our scenario each package contains exactly one letter and for k-universality we have to collect each letter (coupon) not only once but k times (including the deletion of all letters we saw more than once when we restart to collect all coupons again).

Formally, we fix the alphabet Σ with $|\Sigma| = \sigma$ and consider the probability space Σ^ω. We define a probability measure μ by the finite prefixes $u \in \Sigma^*$ of a word $w \in \Sigma^\omega$, i.e., $\mu\{w \in \Sigma^\omega \,|\, u \in \mathrm{Pref}(w)\} = \frac{1}{\sigma^{|u|}}$. Let $w \in \Sigma^\omega$. We define three random variables for modelling that the word contains specific letters: For a given set $A \subset \Sigma^*$ with $|A| = j$, define X_A by

$$X_A = \min\{k \in \mathbb{N} \,|\, w[k] \in A\},$$

i.e., $X_A = \ell$ implies that the first $\ell - 1$ letters of w consist of letters from $\Sigma \backslash A$ and the ℓ^{th} letter is from A. This implies immediately $\mathrm{Prob}(X_A = \ell) = \left(\frac{\sigma-j}{\sigma}\right)^{\ell-1} \frac{j}{\sigma}$. By the geometric series $\sum_{i \in \mathbb{N}_0} q^i = \frac{1}{1-q}$, or especially its first derivative $\sum_{k \in \mathbb{N}} kq^{k-1} = \frac{1}{(1-q)^2}$, we obtain for the expected value

$$\begin{aligned} E[X_A] &= \sum_{\ell \in \mathbb{N}} \ell \,\mathrm{Prob}(X_A = \ell) = \sum_{\ell \in \mathbb{N}} \ell \left(\frac{\sigma - j}{\sigma}\right)^{\ell-1} \frac{j}{\sigma} \\ &= \frac{j}{\sigma} \sum_{\ell \in \mathbb{N}} \ell \left(\frac{\sigma - j}{\sigma}\right)^{\ell-1} = \frac{j}{\sigma} \frac{1}{(1 - \frac{\sigma-j}{\sigma})^2} = \frac{\sigma}{j}. \end{aligned}$$

In a second random variable we capture that we have seen all elements

Table 3.1. Expected number of letters and their variance.

σ	2	3	4	5	6	7	8	9	10
no. of letters	2	5	8	12	16	20	24	29	34
variance	3.6	9.3	18.0	29.7	44.5	62.5	83.5	107.8	135.2

from A. Define X by

$$X = \min\{k \in \mathbb{N} \mid \mathrm{alph}(w[k]) = A\}.$$

By the expected value of X_A, we get

$$E[X] = \sum_{j \in [\sigma]} \frac{\sigma}{j} = \sigma H_\sigma$$

for the σ^{th} partial sum H_σ of the harmonic series. Notice that we have $E[X'_A] \in \Theta(\sigma \log(\sigma))$ (see Table 3.1). Finally, we use a third random variable X_k for describing that a word is k-universal. We get immediately $E[X_k] = k\sigma H_\sigma$. Using Markov's inequality leads to

$$\mathrm{Prob}(\{w \in \Sigma^{\geqslant 2k\sigma H_\sigma} \mid \iota(w) \geqslant k\}) \geqslant \frac{1}{2}, \tag{3.2.27}$$

i.e., if we have chosen $\Theta(k\sigma\log(\sigma))$ letters we have a k-universal word with probability at least $\frac{1}{2}$. To obtain a better understanding of the estimation in (3.2.27), we calculate as well the standard deviation. First, by the second derivative of the geometric series, we get

$$\begin{aligned}
V[X_A] = E[X^2] - E^2[X] &= \left(\sum_{\ell \in \mathbb{N}} \ell^2 \,\mathrm{Prob}(X_A = \ell)\right) - \frac{\sigma^2}{j^2} \\
&= \left(\sum_{\ell \in \mathbb{N}} \ell^2 \left(\frac{\sigma - j}{\sigma}\right)^{\ell - 1} \frac{j}{\sigma}\right) + E[X_A] - \frac{\sigma^2}{j^2} \\
&= \left(\sum_{\ell \in \mathbb{N}_0} \left(\frac{\sigma - j}{\sigma}\right)^{\ell}\right)'' \left(\frac{\sigma - j}{\sigma}\right) \frac{j}{\sigma} + \frac{\sigma}{j} - \frac{\sigma^2}{j^2}
\end{aligned}$$

$$\begin{aligned}
&= \frac{2}{\left(1-\left(\frac{\sigma-j}{\sigma}\right)\right)^3}\left(\frac{\sigma-j}{\sigma}\right)\frac{j}{\sigma}+\frac{\sigma}{j}-\frac{\sigma^2}{j^2}\\
&= \frac{2}{\frac{j^3}{\sigma^3}}\frac{j(\sigma-j)}{\sigma^2}+\frac{\sigma}{j}-\frac{\sigma^2}{j^2}=\frac{2\sigma^3 j(\sigma-j)}{j^3\sigma^2}+\frac{\sigma j-\sigma^2}{j^2}\\
&= \frac{2\sigma(\sigma-j)+\sigma j-\sigma^2}{j^2}=\frac{\sigma^2-\sigma j}{j^2}
\end{aligned}$$

and, thus,

$$V[X]=\sum_{j\in[\sigma]}\frac{\sigma^2-\sigma j}{j^2}=\sum_{j\in[\sigma]}\frac{\sigma^2}{j^2}-\sum_{j\in[\sigma]}\frac{\sigma}{j}<\frac{\pi^2\sigma^2}{6}-\sigma H_\sigma.$$

The first values are presented in Table 3.1. Moreover, we get

$$V[X_k]<k\frac{\pi^2\sigma^2}{6}-k\sigma H_\sigma.$$

Thus, the standard deviation is always smaller than $\frac{\sqrt{k}\pi\sigma}{\sqrt{6}}\approx 1.3\sqrt{k}\sigma$. By Chebyshev's inequality, we finally get for $\lambda>0$

$$\mathrm{Prob}(|X_k-E[X_k]|\geqslant 1.3\lambda\sqrt{k}\sigma)\leqslant\frac{1}{\lambda^2}$$

which is indeed a better estimation than provided by the Markov inequality.

3.3 Reconstruction from Right-Bounded-Block Words

This section is mainly based on [47]. Recall that a word $w\in\Sigma^*$, with a total order $<$ on Σ, is called *right-bounded-block word* if there exist $\mathtt{x},\mathtt{y}\in\Sigma$ with $\mathtt{x}<\mathtt{y}$ and $\ell\in\mathbb{N}_0$ with $w=\mathtt{x}^\ell\mathtt{y}$.

The following definition addresses the reconstruction problem for scattered factors.

3.3.1 Definition. A word $w\in\Sigma^n$ is called *uniquely reconstructible/determined* by the set $S\subseteq\Sigma^*$ if for all words $v\in\Sigma^n\backslash\{w\}$ there exists a word $u\in S$

with $\binom{w}{u} \neq \binom{v}{u}$.

Consider $S = \{\mathtt{ab}, \mathtt{ba}\}$. Then $w = \mathtt{abba}$ is not uniquely reconstructible by S since $\left[\binom{w}{\mathtt{ab}}, \binom{w}{\mathtt{ba}}\right] = [2, 2]$ is also the 2-vector of binomial coefficients of $\mathtt{baab}$. On the other hand $S = \{\mathtt{a}, \mathtt{ab}, \mathtt{ab}^2\}$ reconstructs w uniquely. The following remark gives immediate results for binary alphabets.

3.3.2 Remark. Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ and $w \in \Sigma^n$. If $|w|_\mathtt{a} \in \{0, n\}$ then w contains either only $\mathtt{b}$ or $\mathtt{a}$ and by the given length n of w, w is uniquely determined by $S = \{\mathtt{a}\}$. This fact is in particular an equivalence: $w \in \Sigma^n$ can be uniquely determined by $\{\mathtt{a}\}$ iff $|w|_\mathtt{a} \in \{0, n\}$. If $|w|_\mathtt{a} \in \{1, n-1\}$, w is not uniquely determined by $\{\mathtt{a}\}$ as witnessed by $\mathtt{ab}$ and $\mathtt{ba}$ for $n = 2$. It is immediately clear that the additional information $\binom{w}{\mathtt{ab}}$ leads to unique determinism of w.

Lyndon words play an important role regarding the reconstruction problem. As shown in [99] only scattered factors which are Lyndon words are necessary to determine a word uniquely, i.e., S can always be assumed to be a set of Lyndon words.

3.3.3 Definition. Let $<$ be a total ordering on Σ. A word $w \in \Sigma^*$ is a *Lyndon word* iff for all $u, v \in \Sigma^+$ with $w = uv$, we have $w <_{lex} vu$ where $<_{lex}$ is the lexicographical ordering on words induced by $<$.

3.3.4 Proposition ([99])**.** *Let w and u be two words. The binomial coefficient $\binom{w}{u}$ can be computed using only binomial coefficients of the type $\binom{w}{v}$ where v is a Lyndon word of length up to $|u|$ such that $v \leqslant_{lex} u$.*

To obtain a formula to compute the binomial coefficient $\binom{w}{u}$ for $w, u \in \Sigma^*$ by binomial coefficients $\binom{w}{v_i}$ for Lyndon words $v_1, \ldots, v_k$ with $v_i \in \Sigma^{\leqslant |u|}$, $i \in [k]$, and $k \in \mathbb{N}$ the definitions of shuffle and infiltration are necessary (see, e.g., [81]).

3.3.5 Definition. Let $n_1, n_2 \in \mathbb{N}$, $u_1 \in \Sigma^{n_1}$, and $u_2 \in \Sigma^{n_2}$. Set $n = n_1 + n_2$. The *shuffle* of u_1 and u_2 is the polynomial $u_1 ⧢ u_2 = \sum_{I_1, I_2} w(I_1, I_2)$ where the sum is taken over all pairs (I_1, I_2) of sets that form partitions of $[n]$ such that $|I_1| = n_1$ and $|I_2| = n_2$. If $I_1 = \{i_{1,1} < \ldots < i_{1,n_1}\}$ and $I_2 = \{i_{2,1} < \ldots < i_{2,n_2}\}$, then the word $w(I_1, I_2)$ is defined such that $w[i_{1,1}]w[i_{1,2}] \cdots w[i_{1,n_1}] = u_1$ and $w[i_{2,1}]w[i_{2,2}] \cdots w[i_{2,n_2}] = u_2$ hold.

The infiltration is a variant of the shuffle in which equal letters can be merged.

3.3.6 Definition. Let $n_1, n_2 \in \mathbb{N}$, $u_1 \in \Sigma^{n_1}$, and $u_2 \in \Sigma^{n_2}$. Set $n = n_1 + n_2$. The *infiltration* of u_1 and u_2 is the polynomial $u_1 \downarrow u_2 = \sum_{I_1,I_2} w(I_1, I_2)$, where the sum is taken over all pairs (I_1, I_2) of sets of cardinality n_1 and n_2 respectively, for which the union is equal to the set $[n']$ for some $n' \leqslant n$. Words $w(I_1, I_2)$ are defined as in the previous definition. Note that some $w(I_1, I_2)$ are not well defined if $i_{1,j} = i_{2,k}$ but $u_1[j] \neq u_2[k]$. In that case they do not appear in the previous sum.

Considering for instance $u_1 = \mathtt{aba}$ and $u_2 = \mathtt{ab}$ gives the polynomials

$$\begin{aligned} u_1 \sqcup\!\sqcup\, u_2 &= 2\mathtt{ababa} + 4\mathtt{aabba} + 2\mathtt{aabab} + 2\mathtt{abaab}, \\ u_1 \downarrow u_2 &= \mathtt{aba} \sqcup\!\sqcup\, \mathtt{ab} + \mathtt{aba} + 2\mathtt{abba} + 2\mathtt{aaba} + 2\mathtt{abab}. \end{aligned}$$

Based on Definitions 3.3.5 and 3.3.6, we are able to give a formula to compute a binomial coefficient from the ones making use of Lyndon words. This formula is given implicitely in [99, Theorem 6.4]: Let $u \in \Sigma^*$ be a non-Lyndon word. By [99, Corollary 6.2], there exist non empty words $x, y \in \Sigma^*$ such that every word appearing in the polynomial $x \sqcup\!\sqcup\, y$ is lexicographically less than or equal to $u = xy$. Then, for all words $w \in \Sigma^*$, we have

$$\binom{w}{u} = \frac{1}{(x \sqcup\!\sqcup\, y, u)} \left[\binom{w}{x}\binom{w}{y} - \sum_{v \in \Sigma^*, v \neq u} (x \downarrow y, v) \binom{w}{v} \right],$$

where (P, v) is a notation giving the coefficient of the word v in the polynomial P. One may apply recursively this formula until only Lyndon factors are considered.

3.3.7 Example. Considering $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ the binomial coefficient $\binom{w}{\mathtt{ba}}$ can be computed using the Lyndon words $\mathtt{a}$, and $\mathtt{b}$ by

$$\begin{aligned} \binom{w}{\mathtt{ba}} &= \frac{1}{(\mathtt{b} \sqcup\!\sqcup\, \mathtt{a}, \mathtt{ba})} \left[\binom{w}{\mathtt{b}}\binom{w}{\mathtt{a}} - (\mathtt{b} \downarrow \mathtt{a}, \mathtt{ab}) \binom{w}{\mathtt{ab}} \right] \\ &= \binom{w}{\mathtt{b}}\binom{w}{\mathtt{a}} - \binom{w}{\mathtt{ab}}. \end{aligned}$$

Regarding word length three, the Lyndon words are $\mathtt{aab}$ and $\mathtt{abb}$. Let us give formulas to compute $\binom{w}{\mathtt{aba}}$, $\binom{w}{\mathtt{baa}}$, $\binom{w}{\mathtt{bab}}$ and $\binom{w}{\mathtt{bba}}$. Having $x = \mathtt{ab}$ and $y = \mathtt{a}$, we obtain

$$\binom{w}{\mathtt{aba}} = \binom{w}{\mathtt{ab}}\left[\binom{w}{\mathtt{a}} - 1\right] - 2\binom{w}{\mathtt{aab}}.$$

For $u = \mathtt{baa}$, we can either choose $x = \mathtt{b}$ and $y = \mathtt{aa}$ or $x = \mathtt{ba}$ and $y = \mathtt{a}$.

In the first case, we get

$$\binom{w}{\mathtt{baa}} = \binom{w}{\mathtt{b}}\binom{w}{\mathtt{aa}} - \binom{w}{\mathtt{aba}} - \binom{w}{\mathtt{aab}}$$

and by reinjecting formulas for $\binom{w}{\mathtt{aa}}$ and $\binom{w}{\mathtt{aba}}$, obtained recursively,

$$\binom{w}{\mathtt{baa}} = \left[\binom{w}{\mathtt{a}} - 1\right]\left[\frac{1}{2}\binom{w}{\mathtt{a}}\binom{w}{\mathtt{b}} - \binom{w}{\mathtt{ab}}\right] + \binom{w}{\mathtt{aab}}.$$

Finally, the last two formulas are quite similar to what we already had:

$$\binom{w}{\mathtt{bab}} = \binom{w}{\mathtt{ab}}\left[\binom{w}{\mathtt{b}} - 1\right] - 2\binom{w}{\mathtt{abb}}$$

and
$$\binom{w}{\mathtt{bba}} = \left[\binom{w}{\mathtt{b}} - 1\right]\left[\frac{1}{2}\binom{w}{\mathtt{a}}\binom{w}{\mathtt{b}} - \binom{w}{\mathtt{ab}}\right] + \binom{w}{\mathtt{abb}}.$$

3.3.1 The Reconstruction for Binary Alphabets

In this section we present a method to reconstruct a binary word uniquely from binomial coefficients of right-bounded-block words. Let $n \in \mathbb{N}$ be a natural number and $w \in \{\mathtt{a}, \mathtt{b}\}^n$ a word. Since the word length n is assumed to be known, $|w|_{\mathtt{a}}$ is known if $|w|_{\mathtt{b}}$ is given and vice versa. Set for abbreviation $k_u = \binom{w}{u}$ for $u \in \Sigma^*$. Moreover, we assume w.l.o.g. $k_{\mathtt{a}} \leqslant k_{\mathtt{b}}$ and that $k_{\mathtt{a}}$ is known (otherwise substitute each $\mathtt{a}$ by $\mathtt{b}$ and each $\mathtt{b}$ by $\mathtt{a}$, apply the following reconstruction method and revert the substitution). This implies that w is of the form

$$\mathtt{b}^{s_1}\mathtt{ab}^{s_2}\ldots\mathtt{b}^{s_{k_\mathtt{a}}}\mathtt{ab}^{s_{k_\mathtt{a}+1}} \qquad (3.3.8)$$

for $s_i \in \mathbb{N}_0$ and $i \in [|w|_{\mathtt{a}} + 1]$ with $\sum_{i \in [k_{\mathtt{a}}+1]} s_i = n - k_{\mathtt{a}} = k_{\mathtt{b}}$ and, thus, we get for $\ell \in [k_{\mathtt{a}}]_0$

$$k_{\mathtt{a}^\ell \mathtt{b}} = \binom{w}{\mathtt{a}^\ell \mathtt{b}} = \sum_{i=\ell+1}^{k_{\mathtt{a}}+1} \binom{i-1}{\ell} s_i. \tag{3.3.9}$$

3.3.10 Remark. Notice that for fixed $\ell \in [k_{\mathtt{a}}]_0$ and $c_i = \binom{i-1}{\ell}$ for $i \in [k_{\mathtt{a}} + 1] \backslash [\ell]$, we have $c_i < c_{i+1}$ and especially $c_{\ell+1} = 1$ and $c_{\ell+2} = \ell + 1$.

Equation (3.3.9) shows that reconstructing a word uniquely from binomial coefficients of right-bounded-block words equates to solve a system of Diophantine equations. The knowledge of $k_{\mathtt{b}}, \ldots, k_{\mathtt{a}^\ell \mathtt{b}}$ provides $\ell + 1$ equations. If the equation of $k_{\mathtt{a}^\ell \mathtt{b}}$ has a unique solution for $\{s_{\ell+1}, \ldots, s_{k_{\mathtt{a}}+1}\}$ (in this case we say, by abuse of language, that $k_{\mathtt{a}^\ell \mathtt{b}}$ is *unique*), then the system in row echelon form has a unique solution and, thus, the binary word is uniquely reconstructible. Notice that $k_{\mathtt{a}^{k_{\mathtt{a}}} \mathtt{b}}$ is always unique since $k_{\mathtt{a}^{k_{\mathtt{a}}} \mathtt{b}} = s_{k_{\mathtt{a}}+1}$.

Consider $n = 10$ and $k_{\mathtt{a}} = 4$. This leads to $w = \mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}\mathtt{a}\mathtt{b}^{s_3}\mathtt{a}\mathtt{b}^{s_4}\mathtt{a}\mathtt{b}^{s_5}$ with $\sum_{i \in [5]} s_i = 6$. Given $k_{\mathtt{ab}} = 4$ we get $4 = s_2 + 2s_3 + 3s_4 + 4s_5$. The s_i are not uniquely determined. If $k_{\mathtt{a}^2\mathtt{b}} = 2$ is also given, we obtain the equation $2 = s_3 + 3s_4 + 6s_5$ and, thus, $s_3 = 2$ and $s_4 = s_5 = 0$ is the only solution. Substituting these results in the previous equation leads to $s_2 = 0$ and since we only have six $\mathtt{b}$, we get $s_1 = 4$. Hence, $w = \mathtt{b}^4\mathtt{a}^2\mathtt{b}^2\mathtt{a}^2$ is uniquely reconstructed by $S = \{\mathtt{a}, \mathtt{ab}, \mathtt{a}^2\mathtt{b}\}$.

The following definition captures all solutions for the equation defined by $k_{\mathtt{a}^\ell \mathtt{b}}$ for $\ell \in [k_{\mathtt{a}}]_0$.

3.3.11 Definition. Set $M(k_{\mathtt{a}^\ell \mathtt{b}}) = \{(r_{\ell+1}, \ldots, r_{k_{\mathtt{a}}+1}) | \, k_{\mathtt{a}^\ell \mathtt{b}} = \sum_{i=\ell+1}^{k_{\mathtt{a}}+1} \binom{i-1}{\ell} r_i\}$ for fixed $\ell \in [k_{\mathtt{a}}]_0$. We call $k_{\mathtt{a}^\ell \mathtt{b}}$ *unique* if $|M(k_{\mathtt{a}^\ell \mathtt{b}})| = 1$.

By Remark 3.3.10, the coefficients of each equation of the form (3.3.9) are strictly increasing. The next lemma provides the range each $k_{\mathtt{a}^\ell \mathtt{b}}$ may take under the constraint $\sum_{i=1}^{k_{\mathtt{a}}+1} s_i = n - k_{\mathtt{a}}$.

3.3.12 Lemma. *Let $n \in \mathbb{N}$, $k \in [n]_0$, $j \in [k+1]$ and $c_1, \ldots, c_{k+1}, s_1, \ldots, s_{k+1} \in \mathbb{N}_0$ with $c_i < c_{i+1}$, for $i \in [k]$, and $\sum_{i=1}^{k+1} s_i = n - k$. The sum $\sum_{i=j}^{k+1} c_i s_i$ is maximal iff $s_{k+1} = n - k$ (and, consequently, $s_i = 0$ for all $i \in [k]$).*

Proof. The case $k = 0$ is trivial. Consider the case $n = k$, i.e., $\sum_{i=1}^{k+1} s_i = 0$. This implies immediately $s_i = 0$ for all $i \in [k+1]$ and the equivalence holds. Assume for the rest of the proof $k < n$. If $s_{k+1} = n-k$, then $s_i = 0$ for all $i \leqslant k$ and $\sum_{i=j}^{k+1} c_i s_i = c_{k+1}(n-k)$. Let us assume that the maximal value for $\sum_{i=j}^{k+1} c_i s_i$ can be obtained in another way and that there exist $s'_1, \ldots, s'_{k+1} \in \mathbb{N}_0$, $\ell \in [n-k]$ such that $\sum_{i=1}^{k+1} s'_i = n-k$ and $s'_{k+1} = n-k-\ell$. Thus,

$$c_{k+1}(n-k) \leqslant \sum_{i=j}^{k+1} c_i s'_i = \left(\sum_{i=j}^{k} c_i s'_i\right) + c_{k+1}(n-k-\ell).$$

This implies $\sum_{i=j}^{k} c_i s'_i \geqslant c_{k+1}\ell$. Since the coefficients are strictly increasing we get $\sum_{i=j}^{k} c_i s'_i \leqslant c_k \sum_{i=j}^{k} s'_i < c_{k+1}\ell$, hence, the contradiction. □

3.3.13 Corollary. *Let* $k_{\mathtt{a}} \in [n]_0$, $\ell \in [k_{\mathtt{a}}]_0$, *and* $s_1, \ldots, s_{k_{\mathtt{a}}+1} \in \mathbb{N}_0$ *with* $\sum_{i=1}^{k_{\mathtt{a}}+1} s_i = n - k_{\mathtt{a}}$. *Then* $\binom{w}{\mathtt{a}^\ell\mathtt{b}} \in \left[\binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}})\right]_0$.

Proof. The claim follows directly from Equation (3.3.9) and Lemma 3.3.12. □

The following lemma shows some cases in which $k_{\mathtt{a}^\ell\mathtt{b}}$ is unique.

3.3.14 Lemma. *Let* $k_{\mathtt{a}} \in [n]$, $\ell \in [k_{\mathtt{a}}]_0$ *and* $s_1, \ldots, s_{k_{\mathtt{a}}+1} \in \mathbb{N}_0$ *with* $\sum_{i=1}^{k_{\mathtt{a}}+1} s_i = n - k_{\mathtt{a}}$. *If* $k_{\mathtt{a}^\ell\mathtt{b}} \in [\ell]_0 \cup \{\binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}})\}$ *or* $k_{\mathtt{a}^\ell\mathtt{b}} = \binom{k_{\mathtt{a}}-1}{\ell}r + \binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}}-r)$ *for* $r \in [k_{\mathtt{b}}]_0$ *then* $k_{\mathtt{a}^\ell\mathtt{b}}$ *is unique.*

Proof. First, consider $k_{\mathtt{a}^\ell\mathtt{b}} \in [\ell]_0$. By Remark 3.3.10, we have $c_{\ell+1} = 1$ and $c_{\ell+2} = \ell+1$. By $c_i < c_{i+1}$, we obtain immediately $s_i = 0$ for $i \in [k_{\mathtt{a}}+1]\backslash[\ell+1]$. By setting $s_{\ell+1} = k_{\mathtt{a}^\ell\mathtt{b}}$, the claim is proven. If $k_{\mathtt{a}^\ell\mathtt{b}} = \binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}})$, $s_{k_{\mathtt{a}}+1} = (n-k_{\mathtt{a}})$ and $s_i = 0$ for $i \in [k_{\mathtt{a}}]_0$ is the only possibility. Second, let be $r \in [k_{\mathtt{b}}]_0$ and $k_{\mathtt{a}^\ell\mathtt{b}} = \binom{k_{\mathtt{a}}-1}{\ell}r + \binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}}-r)$ and suppose that $k_{\mathtt{a}^\ell\mathtt{b}}$ is not unique. This implies $s_{k_{\mathtt{a}}+1} < n-k_{\mathtt{a}}-r$. Assume that $s_{k_{\mathtt{a}}+1} = n-k_{\mathtt{a}}-r'$ for $r' \in [k_{\mathtt{b}}]_{>r}$. Thus, there exists $x \in \mathbb{N}$ with $\binom{k_{\mathtt{a}}}{\ell}(n-k_{\mathtt{a}}-r') + x = \frac{(k_{\mathtt{a}}-1)!(k_{\mathtt{a}}(n-k_{\mathtt{a}})-\ell r)}{\ell!(k_{\mathtt{a}}-\ell)!}$, i.e., $x = \frac{(k_{\mathtt{a}}-1)!(k_{\mathtt{a}}r'-\ell r)}{\ell!(k_{\mathtt{a}}-\ell)!}$. By $k_{\mathtt{b}} = n-k_{\mathtt{a}}$, we have $x \leqslant \binom{k_{\mathtt{a}}-1}{\ell}r' = \frac{(k_{\mathtt{a}}-1)!(k_{\mathtt{a}}r'-\ell r')}{\ell!(k_{\mathtt{a}}-\ell)!}$ (we only have r' occurrences of $\mathtt{b}$ left to distribute). By $r' > r$, we have $\frac{(k_{\mathtt{a}}-1)!(k_{\mathtt{a}}r'-\ell r)}{\ell!(k_{\mathtt{a}}-\ell)!} = x < \frac{(k_{\mathtt{a}}-1)!(k_{\mathtt{a}}r'-\ell r)}{\ell!(k_{\mathtt{a}}-\ell)!}$ - a contradiction. □

Since we are not able to fully characterise the uniquely determined values for each $k_{\mathtt{a}^\ell\mathtt{b}}$ for arbitrary n and ℓ, the following proposition gives the characterisation for $\ell \in \{0,1\}$. Notice that we use $k_{\mathtt{a}}$ immediately since it is determinable by n and $k_{\mathtt{a}^0\mathtt{b}} = k_{\mathtt{b}}$.

3.3.15 Proposition. *The word $w \in \Sigma^n$ is uniquely determined by $k_{\mathtt{a}}$ and $k_{\mathtt{ab}}$ iff one of the following occurs*

▷ $k_{\mathtt{a}} = 0$ *or* $k_{\mathtt{a}} = n$ *(and obviously* $k_{\mathtt{ab}} = 0$*),*

▷ $k_{\mathtt{a}} = 1$ *or* $k_{\mathtt{a}} = n-1$ *and* $k_{\mathtt{ab}}$ *is arbitrary,*

▷ $k_{\mathtt{a}} \in [n-2]_{\geqslant 2}$ *and* $k_{\mathtt{ab}} \in \{0, 1, k_{\mathtt{a}}(n-k_{\mathtt{a}})-1, k_{\mathtt{a}}(n-k_{\mathtt{a}})\}$*.*

Proof. Let us first prove that w is uniquely determined in these cases. The claim is obvious if $k_{\mathtt{a}} = 0$ or $k_{\mathtt{a}} = n$ since the word is composed of the same letter repeated n times. If $k_{\mathtt{a}} = 1$, then $w = \mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{n-1-s_1}$ and $\binom{w}{\mathtt{ab}} = n-1-s_1 = k_{\mathtt{ab}}$. Therefore, w is uniquely determined. If $k_{\mathtt{a}} = n-1$, then $w = \mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}\cdots\mathtt{a}\mathtt{b}^{s_n}$ with exactly one of the s_i being non zero and, in fact, equal to one. We have $\binom{w}{\mathtt{ab}} = \sum_{i=2}^{n}(i-1)s_i$ and, if $k_{\mathtt{ab}}$ is given (between 0 and $n-1$), then $s_{k_{\mathtt{ab}}+1} = 1$ is the only non zero exponent. Consider now $k_{\mathtt{a}} \in [n-2]_{\geqslant 2}$, i.e., $w = \mathtt{b}^{s_1}\mathtt{a}\mathtt{b}^{s_2}\ldots\mathtt{b}^{s_{k_{\mathtt{a}}}}\mathtt{a}\mathtt{b}^{s_{k_{\mathtt{a}}+1}}$. Thus, $k_{\mathtt{ab}} = 0$ implies $s_1 = n-k_{\mathtt{a}}$ and $s_2 = 0, \ldots, s_{k_{\mathtt{a}}+1} = 0$ while $k_{\mathtt{ab}} = 1$ implies $s_2 = 1$, $s_1 = n-k_{\mathtt{a}}-1$ and $s_3 = 0, \ldots, s_{k_{\mathtt{a}}+1} = 0$. By Lemma 3.3.12, we know that (3.3.9) is maximal if and only if $s_{k_{\mathtt{a}}+1} = n-k_{\mathtt{a}}$ and all the other s_i are equal to zero. In that case, the value of the sum equals $k_{\mathtt{a}}(n-k_{\mathtt{a}})$. Therefore, if $\binom{w}{\mathtt{ab}} = k_{\mathtt{a}}(n-k_{\mathtt{a}})$, the word w is uniquely determined. Finally, if $k_{\mathtt{ab}} = k_{\mathtt{a}}(n-k_{\mathtt{a}})-1$, we must have $s_{k_{\mathtt{a}}+1} \leqslant n-k_{\mathtt{a}}-1$. If we choose $s_{k_{\mathtt{a}}+1} = n-k_{\mathtt{a}}-1$, it remains that $\sum_{i=1}^{k_{\mathtt{a}}} s_i = 1$ and $\sum_{i=2}^{k_{\mathtt{a}}}(i-1)s_i = k_{\mathtt{a}}-1$. We must have $s_{k_{\mathtt{a}}} = 1$ and the other ones equal to zero. In fact, choosing $s_{k_{\mathtt{a}}+1} = n-k_{\mathtt{a}}-1$ is the only possibility: if otherwise $s_{k_{\mathtt{a}}+1} = n-k_{\mathtt{a}}-\ell$ with $\ell > 1$, we obtain that $\sum_{i=2}^{k_{\mathtt{a}}}(i-1)s_i \geqslant \ell k_{\mathtt{a}}-1$ with $\sum_{i=1}^{k_{\mathtt{a}}} s_i = \ell$. It is easy to check with Lemma 3.3.12 that these conditions are incompatible.

We now need to prove that w cannot be uniquely determined if $k_{\mathtt{a}} \in [n-2]_{\geqslant 2}$ and $k_{\mathtt{ab}} \in [k_{\mathtt{a}}(n-k_{\mathtt{a}})-2]_{\geqslant 2}$. To this aim we will give two different sets of values for the s_i. The first decomposition is the greedy one. Let us put $s_{k_{\mathtt{a}}+1} = \lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \rfloor$, $s_{(k_{\mathtt{ab}} \bmod k_{\mathtt{a}})+1} = 1$ and the other s_i equal to 0. Let us finally modify the value of s_1 (which is, at this stage, equal to 0 or 1) by adding the value needed. By $\sum_{i=1}^{k_{\mathtt{a}}+1} s_i = n - k_{\mathtt{a}}$, we get $s_1 \leftarrow s_1 + (n - k_{\mathtt{a}}) - \lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \rfloor - 1$. This implies $\sum_{i=1}^{k_{\mathtt{a}}+1} s_i = 1 + (n - k_{\mathtt{a}}) - \left\lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \right\rfloor - 1 + \left\lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \right\rfloor = n - k_{\mathtt{a}}$ and $s_i \geqslant 0$ for all i. Moreover, we have $\sum_{i=2}^{k_{\mathtt{a}}+1} (i-1)s_i = (k_{\mathtt{ab}} \bmod k_{\mathtt{a}}) + k_{\mathtt{a}} \left\lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \right\rfloor = k_{\mathtt{ab}}$.

Now we provide a second decomposition for the s_i. First, let us assume that $2 \leqslant k_{\mathtt{ab}} < k_{\mathtt{a}}$. In that case, the greedy algorithm sets $s_{k_{\mathtt{ab}}+1} = 1$, $s_1 = n - k_{\mathtt{a}} - 1$ and the other s_i to 0. Let us now set $s_1 = n - k_{\mathtt{a}} - 2$ and all the other s_i to 0. Then, update $s_{k_{\mathtt{ab}}} \leftarrow s_{k_{\mathtt{ab}}} + 1$ and $s_2 \leftarrow s_2 + 1$ (in the case where $k_{\mathtt{ab}} = 2$, s_2 will be equal to 2 after these manipulations). We have that the sum in (3.3.9) is equal to $1 + (k_{\mathtt{ab}} - 1)$ as needed. Finally, if $k_{\mathtt{ab}} \geqslant k_{\mathtt{a}}$, then $s_{k_{\mathtt{a}}+1}$ was non zero in the greedy decomposition, and the idea is to reduce it by 1. Let us set $s_{k_{\mathtt{a}}+1} = \lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \rfloor - 1$ and the other s_i to 0. Then, let us update some values: $s_{(k_{\mathtt{ab}} \bmod k_{\mathtt{a}})+2} \leftarrow s_{(k_{\mathtt{ab}} \bmod k_{\mathtt{a}})+2} + 1$ and $s_{k_{\mathtt{a}}} \leftarrow s_{k_{\mathtt{a}}} + 1$ if $(k_{\mathtt{ab}} \bmod k_{\mathtt{a}}) \neq k_{\mathtt{a}} - 1$, and $s_{k_{\mathtt{a}}} = 2$, $s_2 = 1$ otherwise. Finally, set s_1 to the right value, i.e., $n - k_{\mathtt{a}} - \sum_{i=2}^{k_{\mathtt{a}}+1} s_i$. It can be easily checked that, in both cases, $s_1 \geqslant 0$ (notice that $(k_{\mathtt{ab}} \bmod k_{\mathtt{a}}) = k_{\mathtt{a}} - 1$ implies that $\lfloor \frac{k_{\mathtt{ab}}}{k_{\mathtt{a}}} \rfloor \leqslant n - k_{\mathtt{a}} - 2$) and that all s_i sum up to $n - k_{\mathtt{a}}$. Similarly, we can check that $\sum_{i=2}^{k_{\mathtt{a}}+1} (i-1)s_i$ is equal to $k_{\mathtt{ab}}$ in both cases.

To sum up, we gave two different decompositions for the s_i in cases where $k_{\mathtt{a}} \in [n-2]_{\geqslant 2}$ and $k_{\mathtt{ab}} \in [k_{\mathtt{a}}(n - k_{\mathtt{a}}) - 2]_{\geqslant 2}$. That implies that w cannot be uniquely determined in those cases. □

In all cases not covered by Proposition 3.3.15, the word cannot be uniquely determined by $\binom{w}{\mathtt{a}}$ and $\binom{w}{\mathtt{ab}}$. The following theorem combines the reconstruction of a word with the binomial coefficients of right-bounded-block words.

3.3.16 Theorem. *Let $j \in [k_{\mathtt{a}}]_0$. If $k_{\mathtt{a}^j\mathtt{b}}$ is unique, then the word $w \in \Sigma^n$ is uniquely determined by $\{\mathtt{b}, \mathtt{ab}, \mathtt{a}^2\mathtt{b}, \ldots, \mathtt{a}^j\mathtt{b}\}$.*

Proof. If $k_{\mathtt{a}^j\mathtt{b}}$ is unique, the coefficients $s_{j+1}, \ldots, s_{k_{\mathtt{a}}+1}$ are uniquely determined. Substituting backwards the known values in the first $j-1$ equations (3.3.9) (for $\ell = 1, \ldots, j-1$) we can now obtain successively the values for $s_j, \ldots, s_1$. □

3.3.17 Corollary. *Let ℓ be minimal such that $k_{\mathtt{a}^\ell\mathtt{b}}$ is unique. Then w is uniquely determined by $\{\mathtt{a}, \mathtt{ab}, \mathtt{a}^2\mathtt{b}, \dots, \mathtt{a}^\ell\mathtt{b}\}$ and not uniquely determined by any $\{\mathtt{a}, \mathtt{ab}, \mathtt{a}^2\mathtt{b}, \dots, \mathtt{a}^i\mathtt{b}\}$ for $i < \ell$.*

Proof. The claim follows directly from Theorem 3.3.16. □

By [72], an upper bound on the number of binomial coefficients to uniquely reconstruct the word $w \in \Sigma^n$ is given by the amount of the binomial coefficients of the $(\lfloor \frac{16}{7}\sqrt{n} \rfloor + 5)$-spectrum. Notice that implicitly the full spectrum is assumed to be known. By Proposition 3.1.3, Lyndon words up to this length suffice. Since there are $\frac{1}{n}\sum_{d|n}\mu(d) \cdot 2^{\frac{n}{d}}$ Lyndon words of length n, the combination of both results presented in [72, 99] states that, for $n > 6$,

$$\sum_{i=1}^{\lfloor \frac{16}{7}\sqrt{n} \rfloor + 5} \frac{1}{i}\sum_{d|i}\mu(d) \cdot 2^{\frac{i}{d}} \tag{3.3.18}$$

binomial coefficients are sufficient for a unique reconstruction with the Möbius function μ. Up to now, it was the best known upper bound.

Theorem 3.3.16 shows that $\min\{k_\mathtt{a}, k_\mathtt{b}\} + 1$ binomial coefficients are enough for reconstructing a binary word uniquely. By Proposition 3.3.15, we need exactly one binomial coefficient if $n \in [3]$ and at most two if $n = 4$. For $n \in \{5, 6\}$ we need at most $n - 2$ different binomial coefficients. The following theorem shows that, by Theorem 3.3.16, we need strictly less binomial coefficients for $n > 6$.

3.3.19 Theorem*. *Let $w \in \Sigma^n$. We have that $\min\{k_\mathtt{a}, k_\mathtt{b}\} + 1$ binomial coefficients suffice to uniquely reconstruct w. If $k_\mathtt{a} \leqslant k_\mathtt{b}$, then the set of sufficient binomial coefficients is $S = \{\mathtt{b}, \mathtt{ab}, \mathtt{a}^2\mathtt{b}, ..., \mathtt{a}^h\mathtt{b}\}$ where $h = \lfloor \frac{n}{2} \rfloor$. If $k_\mathtt{a} > k_\mathtt{b}$, then the set is $S = \{\mathtt{a}, \mathtt{ba}, \mathtt{b}^2\mathtt{a}, ..., \mathtt{b}^h\mathtt{a}\}$. This bound is strictly smaller than (3.3.18).*

Proof. Assume w.l.o.g. $k_\mathtt{a} \leqslant k_\mathtt{b}$. Then $k_\mathtt{a} \leqslant \frac{n}{2}$ and Theorem 3.3.16 shows that words in the set $\{\mathtt{b}, \mathtt{ab}, \dots, \mathtt{a}^{\lfloor \frac{n}{2} \rfloor}\mathtt{b}\}$ can reconstruct w uniquely. If $k_\mathtt{a} > k_\mathtt{b}$, the set S is obtained by replacing the letter $\mathtt{a}$ by $\mathtt{b}$ and vice-versa.

Set $N_2(i) := \frac{1}{i}\sum_{d|i}\mu(d)2^{\frac{i}{d}}$ for all $i \in [\lfloor \frac{16}{7}\sqrt{n} \rfloor + 5]$, i.e., $\sum_{i=1}^{\lfloor \frac{16}{7}\sqrt{n} \rfloor + 5} N_2(i)$, which is Equation (3.3.18), binomial coefficients suffice. By [43, Lemma 2.4], we have

$$N_2(i) \geqslant \frac{1}{i}\left(2^i - \frac{2^{\frac{i}{2}}-1}{2-1}\right) = \frac{1}{i}\left(2^i - 2^{\frac{i}{2}} + 1\right) = \frac{1}{i}\left(2^{\frac{i}{2}}(2^{\frac{i}{2}}-1)+1\right) \geqslant \frac{2^{\frac{i}{2}}}{i}.$$

This results in

$$\sum_{i=1}^{\lfloor \frac{16}{7}\sqrt{n}\rfloor+5} N_2(i) \geqslant \sum_{i=1}^{\lfloor \frac{16}{7}\sqrt{n}\rfloor+5} \frac{2^{\frac{i}{2}}}{i} \geqslant \frac{1}{\frac{16}{7}\sqrt{n}+5}\,\frac{\sqrt{2}^{\frac{16}{7}\sqrt{n}+5}-1}{\sqrt{2}-1}.$$

We want to show that this quantity is at least equal to $\frac{n+1}{2}$. Let us define

$$f(x) = \frac{1}{\frac{16}{7}\sqrt{x}+5}\,\frac{\sqrt{2}^{\frac{16}{7}\sqrt{x}+5}-1}{\sqrt{2}-1} - \frac{x+1}{2}$$

for all $x > 0$, which is the continuous extension on $\mathbb{R}^+$ of the quantity we are interested in. It is easy to verify by hand that $f(1), f(2), f(3)$ and $f(4)$ are positive. Let us formally show that $f(x) > 0$ for all $x \geqslant 5$. Since this function is differentiable, we get with $y = \frac{16}{7}\sqrt{x}+5$

$$f'(x) = \frac{1}{y}\sqrt{2}^y\,\frac{\ln(\sqrt{2})}{\sqrt{2}-1}\,\frac{8}{7\sqrt{x}} - \frac{1}{y^2}\,\frac{8}{7\sqrt{x}}\,\frac{\sqrt{2}^y-1}{\sqrt{2}-1} - \frac{1}{2}.$$

Thus, we have $\sqrt{x} = \frac{7y-35}{16}$ and $y \geqslant 7$ for all $x \geqslant 1$. By injecting y in the previous expression, and reducing to the common denominator, we have to show that

$$\begin{aligned}
&2y\sqrt{2}^y 128\ln(\sqrt{2}) - 256(\sqrt{2}^y - 1) - 7(7y-35)y^2(\sqrt{2}-1)\\
=&\sqrt{2}^y(128\ln(2)y - 256) + 256 - 49y^3(\sqrt{2}-1) + 245y^2(\sqrt{2}-1)\\
\geqslant&\sqrt{2}^y 365 + 256 - 49y^3(\sqrt{2}-1) + 245y^2(\sqrt{2}-1)
\end{aligned}$$

is strictly positive. Let us call the last quantity $g(y)$. We will show that it is positive for all $y \geqslant 10.05$, which means that $f(x)$ is positive for all x such that $\frac{16}{7}\sqrt{x}+5 \geqslant 10.05$, i.e., for all $x \geqslant 5$. We have

$$\begin{aligned}
g'(y) &= 365\sqrt{2}^y\ln(\sqrt{2}) - 147(\sqrt{2}-1)y^2 + 490(\sqrt{2}-1)y,\\
g''(y) &= 365\sqrt{2}^y(\ln(\sqrt{2}))^2 - 294(\sqrt{2}-1)y + 490(\sqrt{2}-1),
\end{aligned}$$

$$g'''(y) = 365\sqrt{2}^y(\ln(\sqrt{2}))^3 - 294(\sqrt{2}-1),$$

and $g'''(7) > 50$, $g''(8.5) > 2$, $g'(10.05) > 8$ and finally $g(10.05) > 1787$. Since $g'''(y)$ is increasing and positive in 7, $g''(y)$ is increasing for $y \geqslant 7$. Therefore, $g'(y)$ is increasing for $y \geqslant 8.5$ and finally $g(y)$ is increasing for $y \geqslant 10.05$ and positive. □

3.3.20 Remark. By Lemma 3.3.14, we know that $k_{\mathtt{a}^\ell\mathtt{b}}$ is unique if it is in $[\ell]_0$ or exactly $\binom{k_\mathtt{a}}{\ell}(n-k_\mathtt{a})$. The probability for the latter is $\frac{1}{2^n}$ for $w \in \{\mathtt{a},\mathtt{b}\}^n$. If $k_{\mathtt{a}^\ell\mathtt{b}} = m \in [\ell]_0$ we get, by (3.3.9), immediately $s_{\ell+1} = m$ and $s_i = 0$ for $\ell+2 \leqslant i \leqslant k_\mathtt{a}+1$. Hence, the values for s_j for $j \in [\ell]$ are not determined. By $\sum_{i\in[\ell]} s_i = n - k_\mathtt{a} - m$, there are $d = \sum_{i\in[\ell]_0}\binom{\ell}{\ell-i}\binom{n-k_\mathtt{a}-m-1}{i-1}$ possibilities to fulfil the constraints, i.e., we have a probability of $\frac{d}{2^n}$ to have such a word.

3.3.2 Reconstruction for Arbitrary Alphabets

In this section we address the problem of reconstructing words over arbitrary alphabets from their scattered factors. We begin with a series of results of algorithmic nature. Let $\Sigma = \{\mathtt{a}_1, \dots, \mathtt{a}_q\}$ be an alphabet equipped with the ordering $\mathtt{a}_i < \mathtt{a}_j$ for $1 \leqslant i < j \leqslant q \in \mathbb{N}$.

3.3.21 Definition. Let $w_1, \dots, w_k \in \Sigma^*$ for $k \in \mathbb{N}$, and $K = (k_\mathtt{a})_{\mathtt{a}\in\Sigma}$ a sequence of $|\Sigma|$ natural numbers. A *K–valid marking* of $w_1, \dots, w_k$ is a mapping $\psi : [k] \times \mathbb{N} \to \mathbb{N}$ such that for all $j \in [k]$, $i, \ell \in [|w_j|]$, and $\mathtt{a} \in \Sigma$ it holds

- ▻ if $w_j[i] = \mathtt{a}$ then $\psi(j,i) \leqslant k_\mathtt{a}$,
- ▻ if $i < \ell \leqslant |w_j|$ and $w_j[i] = w_j[\ell] = \mathtt{a}$ then $\psi(j,i) < \psi(j,\ell)$.

A *K-valid marking* of $w_1, \dots, w_k$ is represented as the string w_1^ψ, $w_2^\psi, \dots, w_k^\psi$, where $w_j^\psi[i] = (w_j[i])_{\psi(j,i)}$ for fresh letters $(w_j[i])_{\psi(j,i)}$.

For instance, let $k = 2$, $\Sigma = \{\mathtt{a},\mathtt{b}\}$, and $w_1 = \mathtt{aab}$, $w_2 = \mathtt{abb}$. Let $k_\mathtt{a} = 3, k_\mathtt{b} = 2$ define the sequence K. A K-valid marking of w_1, w_2 would be $w_1^\psi = (\mathtt{a})_1(\mathtt{a})_3(\mathtt{b})_1, w_2^\psi = (\mathtt{a})_2(\mathtt{b})_1(\mathtt{b})_2$ defining ψ implicitly by the indices. We used parentheses in the marking of the letters in order to avoid confusions.

We recall that a topological sorting of a directed graph $G = (V, E)$, with $V = \{v_1, \ldots, v_n\}$, is a linear ordering $v_{\sigma(1)} < v_{\sigma(2)} < \ldots < v_{\sigma(n)}$ of the nodes, defined by the permutation $\sigma : [n] \to [n]$, such that there exists no edge in E from $v_{\sigma(i)}$ to $v_{\sigma(j)}$ for any $i > j$ (i.e., if v_a comes after v_b in the linear ordering, for some $a = \sigma(i)$ and $b = \sigma(j)$, then we have $i > j$ and there should be no edge between v_a and v_b). It is a folklore result that any directed graph G has a topological sorting if and only if G is acyclic.

3.3.22 Definition. Let $w_1, \ldots, w_k \in \Sigma^*$ for $k \in \mathbb{N}$, $K = (k_{\mathtt{a}})_{\mathtt{a} \in \Sigma}$ a sequence of $|\Sigma|$ natural numbers, and ψ a *K−valid marking* of $w_1, \ldots, w_k$. Let G_ψ be the graph that has $\sum_{\mathtt{a} \in \Sigma} k_{\mathtt{a}}$ nodes, labelled with the letters $(\mathtt{a})_1, \ldots, (\mathtt{a})_{k_{\mathtt{a}}}$, for all $\mathtt{a} \in \Sigma$, and the directed edges $((w_j[i])_{\psi(j,i)}, (w_j[i+1])_{\psi(j,i+1)})$, for all $j \in [k]$, $i \in [|w_j|]$, and $((\mathtt{a})_i, (\mathtt{a})_{i+1})$, for all occurring i and $\mathtt{a} \in \Sigma$. We say that there exists *a valid topological sorting* of the ψ-marked letters of the words $w_1, \ldots, w_k$ if there exists a topological sorting of the nodes of G_ψ, i.e., G_ψ is a directed acyclic graph.

The graph associated with the K-valid marking of w_1, w_2 from above would have the five nodes $(\mathtt{a})_1, (\mathtt{a})_2, (\mathtt{a})_3, (\mathtt{b})_1, (\mathtt{b})_2$ and the six directed edges $((\mathtt{a})_1, (\mathtt{a})_3)$, $((\mathtt{a})_3, (\mathtt{b})_1)$, $((\mathtt{a})_2, (\mathtt{b})_1)$, $((\mathtt{b})_1, (\mathtt{b})_2)$, $((\mathtt{a})_1, (\mathtt{a})_2)$, $((\mathtt{a})_2, (\mathtt{a})_3)$ (where the direction of the edge is from the left node to the right node of the pair defining it). This graph has the topological sorting $(\mathtt{a})_1(\mathtt{a})_2(\mathtt{a})_3(\mathtt{b})_1(\mathtt{b})_2$.

3.3.23 Theorem*. *For $w_1, \ldots, w_k \in \Sigma^*$ and a sequence $K = (k_{\mathtt{a}})_{\mathtt{a} \in \Sigma}$ of $|\Sigma|$ natural numbers, there exists a word w such that w_i is a scattered factor of w with $|w|_{\mathtt{a}} = k_{\mathtt{a}}$, for all $i \in [k]$ and all $\mathtt{a} \in \Sigma$, if and only if there exist a K-valid marking ψ of the words $w_1, \ldots, w_k$ and a valid topological sorting of the ψ-marked letters of the words $w_1, \ldots, w_k$.*

Proof. Let us now assume that there exists a K-valid marking ψ of the words $w_1, \ldots, w_k$, and there exists a valid topological sorting of the ψ-marked letters of the words $w_1, \ldots, w_k$. Let w' be the word obtained by writing the nodes of G_ψ in the order given by its topological sorting and removing their markings. It is clear that w' has w_i as a scattered factor, for all $i \in [k]$, and that $|w'|_{\mathtt{a}} \leqslant k_{\mathtt{a}}$, for all $\mathtt{a} \in \Sigma$. Let now $w = w' \prod_{\mathtt{a} \in \Sigma} \mathtt{a}^{k_{\mathtt{a}} - |w'|_{\mathtt{a}}}$, where $\prod_{\mathtt{a}\in\Sigma} \mathtt{a}^{k_{\mathtt{a}}-|w'|_{\mathtt{a}}}$ is the concatenation of the factors $\mathtt{a}^{k_{\mathtt{a}}-|w'|_{\mathtt{a}}}$, for $\mathtt{a} \in \Sigma$ in some fixed order. Now w has w_i as a scattered factor, for all $i \in [k]$, and $|w|_{\mathtt{a}} = k_{\mathtt{a}}$, for all $\mathtt{a} \in \Sigma$. □

Next we show that in Theorem 3.3.23 uniqueness propagates in the $\Leftarrow$-direction.

3.3.24 Corollary*. *Let $w_1, \ldots, w_k \in \Sigma^*$ and $K = (k_{\mathtt{a}})_{\mathtt{a}\in\Sigma}$ a sequence of $|\Sigma|$ natural numbers. If the following statements hold*

▻ *there exists a unique K-valid marking ψ of the words $w_1, \ldots, w_k$,*

▻ *in the unique K-valid marking ψ we have that for each $\mathtt{a} \in \Sigma$ and $\ell \in [k_{\mathtt{a}}]$ there exists $i \in [k]$ and $j \in [|w_i|]$ with $\psi(i, j) = \ell$, and*

▻ *there exists a unique valid topological sorting of the ψ-marked letters of the words $w_1, \ldots, w_k$*

then there exists a unique word w such that w_i is a scattered factor of w, for all $i \in [k]$ and $|w|_{\mathtt{a}} = k_{\mathtt{a}}$ for all $\mathtt{a} \in \Sigma$.

Proof. Let w be the word obtained by writing in order the letters of the unique valid topological sorting of the ψ-marked letters of the words $w_1, \ldots, w_k$ and removing their markings. It is clear that w' has w_i as a scattered factor, for all $i \in [k]$, and that $|w|_{\mathtt{a}} = k_{\mathtt{a}}$, for all $\mathtt{a} \in \Sigma$. The word w is uniquely defined (as there is no other K-valid marking nor valid topological sorting of the ψ-marked letters), and $|w|_{\mathtt{a}} = k_{\mathtt{a}}$, for all $\mathtt{a} \in \Sigma$. □

In order to state the second result, we need the projection $\pi_S(w)$ of a word $w \in \Sigma^*$ on $S \subseteq \Sigma$: $\pi_S(w)$ is obtained from w by removing all letters from $\Sigma \backslash S$.

3.3.25 Theorem*. *Set $W = \{w_{\mathtt{a},\mathtt{b}} \mid \mathtt{a} < \mathtt{b} \in \Sigma\}$ such that*

▻ *$w_{\mathtt{a},\mathtt{b}} \in \{\mathtt{a}, \mathtt{b}\}^*$ for all $\mathtt{a}, \mathtt{b} \in \Sigma$,*

▻ *for all $w, w' \in W$ and all $\mathtt{a} \in \Sigma$, if $|w|_{\mathtt{a}} \cdot |w'|_{\mathtt{a}} > 0$, then $|w|_{\mathtt{a}} = |w'|_{\mathtt{a}}$.*

Then there exists at most one $w \in \Sigma^*$ such that $w_{\mathtt{a},\mathtt{b}}$ is $\pi_{\{\mathtt{a},\mathtt{b}\}}(w)$ for all $\mathtt{a}, \mathtt{b} \in \Sigma$.

Proof. Notice $|W| = \frac{q(q-1)}{2}$. Let $k_{\mathtt{a}} = |w_{\mathtt{a},\mathtt{b}}|_{\mathtt{a}}$, for $\mathtt{a} < \mathtt{b} \in \Sigma$. These numbers are clearly well defined, by the second item in our hypothesis. Let $K = (k_{\mathtt{a}})_{\mathtt{a}\in\Sigma}$. It is immediate that there exists a unique K-valid marking ψ of the words $(w_{\mathtt{a},\mathtt{b}})_{\mathtt{a}<\mathtt{b}\in\Sigma}$. As each two marked letters $(\mathtt{a})_i$ and $(\mathtt{b})_j$ (i.e., each two nodes $(\mathtt{a})_i$ and $(\mathtt{b})_j$ of G_ψ) appear in the marked word $w^{\psi}_{\mathtt{a},\mathtt{b}}$, we know the order in which these two nodes should occur in a topological sorting of G_ψ. This means that, if G_ψ is acyclic, then it has a unique topological sorting. Our statement follows now from Corollary 3.3.24. □

3.3.26 Remark. Given the set $W = \{w_{\mathtt{a},\mathtt{b}} \mid \mathtt{a} < \mathtt{b} \in \Sigma\}$ as in the statement of Theorem 3.3.25, with $k_{\mathtt{a}} = |w_{\mathtt{a},\mathtt{b}}|_{\mathtt{a}}$, for $\mathtt{a} < \mathtt{b} \in \Sigma$, and $K = (k_{\mathtt{a}})_{\mathtt{a}\in\Sigma}$, we can produce the unique K-valid marking ψ of the words $(w_{\mathtt{a},\mathtt{b}})_{\mathtt{a}<\mathtt{b}\in\Sigma}$ in linear time $O(\sum_{\mathtt{a}<\mathtt{b}\in\Sigma} |w_{\mathtt{a},\mathtt{b}}|) = O((q-1)\sum_{\mathtt{a}\in\Sigma} k_{\mathtt{a}})$: just replace the i^{th} letter $\mathtt{a}$ of $w_{\mathtt{a},\mathtt{b}}$ by $(\mathtt{a})_i$, for all $\mathtt{a}$ and i. The graph G_ψ has $O((q-1)\sum k_{\mathtt{a}})$ edges and $O(\sum k_{\mathtt{a}})$ vertices and can be constructed in linear time $O((q-1)\sum k_{\mathtt{a}})$. Sorting G_ψ topologically takes $O((q-1)\sum k_{\mathtt{a}})$ time (see, e.g., the handbook [21]). As such, we conclude that reconstructing a word $w \in \Sigma^*$ from its projections over all two-letter-subsets of Σ can be done in linear time w.r.t. the total length of the respective projections.

Theorem 3.3.25 is in a sense optimal: in order to reconstruct a word over Σ uniquely, we need all its projections on two-letter-subsets of Σ. That is, it is always the case that for a strict subset U of $\{\{\mathtt{a},\mathtt{b}\} \mid \mathtt{a} < \mathtt{b} \in \Sigma\}$, with $|U| = \frac{q(q-1)}{2} - 1$, there exist two words $w' \neq w$ such that $\{\pi_p(w') \mid p \in U\} = \{\pi_p(w) \mid p \in U\}$. We can, in fact, show the following results:

3.3.27 Theorem*. *Let $S_1, \ldots, S_k$ be subsets of Σ. The following hold:*

1. *If each pair $\{\mathtt{a},\mathtt{b}\} \subseteq \Sigma$ is included in at least one of the sets S_i, then we can reconstruct any word uniquely from its projections $\pi_{S_1}(\cdot), \ldots, \pi_{S_k}(\cdot)$.*

2. *If there exists a pair $\{\mathtt{a},\mathtt{b}\}$ that is not contained in any of the sets $S_1, \ldots, S_k$, then there exist two words w and w' such that $w \neq w'$ and $\pi_{S_1}(w) = \pi_{S_1}(w'), \ldots, \pi_{S_k}(w) = \pi_{S_k}(w')$.*

Proof. The first part is, once again, a consequence of Corollary 3.3.24. The second part can be shown by assuming that $\Sigma = \{\mathtt{a}_1, \ldots, \mathtt{a}_q\}$ and the pair $\{\mathtt{a}_1, \mathtt{a}_2\}$ is not contained in any of the sets $S_1, \ldots, S_k$. Then, for $w =$

$\mathtt{a}_1\mathtt{a}_3\mathtt{a}_4 \ldots \mathtt{a}_q$ and $w' = \mathtt{a}_2\mathtt{a}_3\mathtt{a}_4 \ldots \mathtt{a}_q$, we have that $\pi_{S_1}(w) = \pi_{S_1}(w'), \ldots,$ $\pi_{S_k}(w) = \pi_{S_k}(w')$. □

In this context, we can ask how efficiently can we decide if a word is uniquely reconstructible from the projections $\pi_{S_1}(\cdot), \ldots, \pi_{S_k}(\cdot)$ for $S_1, \ldots, S_k \subset \Sigma$.

3.3.28 Theorem*. *Given the sets $S_1, \ldots, S_k \subset \Sigma$, we decide whether we can reconstruct any word uniquely from its projections $\pi_{S_1}(\cdot), \ldots, \pi_{S_k}(\cdot)$ in $O(q^2k)$ time. Moreover, under the* Strong Exponential Time Hypothesis *(see the survey [8] and the references therein), there is no $O(q^{2-d}k^c)$ algorithm for solving the above decision problem, for any $d, c > 0$.*

Proof. We begin with a series of preliminaries. Let us recall the *Orthogonal Vectors* problem: Given sets A, B consisting of n vectors in $\{0,1\}^k$, decide whether there are vectors $a \in A$ and $b \in B$ which are orthogonal (i.e., for any $i \in [k]$ we have $a[i]b[i] = 0$). This problem can be solved naïvely in $O(n^2k)$ time, but under the *Strong Exponential Time Hypothesis* there is no $O(n^{2-d}k^c)$ algorithm for solving it, for any $d, c > 0$ (once more, see the survey [8] and the references therein).

We show that our problem is equivalent to the *Orthogonal Vectors* problem.

Let us first assume that we are given the sets $S_1, \ldots, S_k \subset \Sigma$, and we want to decide whether we can reconstruct any word uniquely from its projections $\pi_{S_1}(\cdot), \ldots, \pi_{S_k}(\cdot)$. This is equivalent, according to Theorem 3.3.27, to checking whether each pair $\{\mathtt{a}, \mathtt{b}\} \subseteq \Sigma$ is included in at least one of the sets S_i. For each letter $\mathtt{a}$ of Σ we define the k-dimensional vectors $x_\mathtt{a}$ where $x_\mathtt{a}[i] = 1$ if $\mathtt{a} \in S_i$ and $x_\mathtt{a}[i] = 0$ if $\mathtt{a} \notin S_i$. This can be clearly done in $O(qk)$ time. Now, there exists a pair $\{\mathtt{a}, \mathtt{b}\}$ that is not contained in any of the sets $S_1, \ldots, S_k$ if and only if there exists a pair of vectors $\{x_\mathtt{a}, x_\mathtt{b}\}$ such that $x_\mathtt{a}[i]x_\mathtt{b}[i] = 0$ for all $i \in [k]$. We can check whether there exists a pair of vectors $\{x_\mathtt{a}, x_\mathtt{b}\}$ such that $x_\mathtt{a}[i]x_\mathtt{b}[i] = 0$ for all $i \in [k]$ by solving the *Orthogonal Vectors* problem by using for both input sets of vectors the set $\{x_\mathtt{a} \mid \mathtt{a} \in \Sigma\}$. As such, we can check whether there exists a pair $\{\mathtt{a}, \mathtt{b}\}$ that is not contained in any of the sets $S_1, \ldots, S_k$ in $O(q^2k)$ time.

Let us now assume that we are given two sets A, B consisting of n vectors in $\{0,1\}^k$, and we want to decide whether there are vectors $a \in A$

and $b \in B$ which are orthogonal (i.e., for any $i \in [k]$ we have $a[i]b[i] = 0$). We can compute the set of $(k+2)$-dimensional vectors A' containing the vectors of A extended with two new positions (position $k+1$ and position $k+2$) set to 10 and the vectors of B extended with two new positions (position $k+1$ and position $k+2$) set to 01. To decide whether there are vectors $a \in A$ and $b \in B$ which are orthogonal is equivalent to decide whether there are vectors $a, b \in A'$ which are orthogonal (if two such vectors exist, they must be different on their last two positions, so one must come from A and one from B). Assume that $A' = \{x_1, x_2, \ldots, x_{2n}\}$. Now we define an alphabet $\Sigma = \{\mathtt{a}_1, \ldots, \mathtt{a}_{2n}\}$ of size $2n$ and the sets $S_1, \ldots, S_k$, where $\mathtt{a}_j \in S_i$ if and only if $x_j[i] = 1$. Computing A' and then the alphabet Σ and the sets S_i, for $i \in [k]$, takes $O(nk)$ time. Now, to decide whether there are vectors $x_i, x_j \in A'$ which are orthogonal is equivalent to decide whether there exists a pair of letters $\{\mathtt{a}_i, \mathtt{a}_j\}$ of Σ that is not contained in any of the sets $S_1, \ldots, S_k$. The conclusion of the theorem now follows. □

Coming now back to combinatorial results, we use the method developed in Section 3.3.1 to reconstruct a word over an arbitrary alphabet. We show that we need at most $\sum_{i \in [q]} |w|_i (q+1-i)$ different binomial coefficients to reconstruct w uniquely for the alphabet $\Sigma = \{1, \ldots, q\}$. In fact, following the results from the first part of this section, we apply this method on all combinations of two letters. Consider for an example that for $w \in \{\mathtt{a}, \mathtt{b}, \mathtt{n}\}^6$ the following binomial coefficients $\binom{w}{\mathtt{a}^0\mathtt{b}} = 1$, $\binom{w}{\mathtt{a}^0\mathtt{n}} = 2$, $\binom{w}{\mathtt{a}^1\mathtt{b}} = 0$, $\binom{w}{\mathtt{a}^1\mathtt{n}} = 3$, $\binom{w}{\mathtt{b}^1\mathtt{n}} = 2$, and $\binom{w}{\mathtt{a}^2\mathtt{n}} = 1$ are given. By $|w| = 6$, $|w|_\mathtt{b} = 1$, and $|w|_\mathtt{n} = 2$, we get $|w|_\mathtt{a} = 3$. Applying the method from Section 3.3.1 for $\{\mathtt{a}, \mathtt{b}\}$, $\{\mathtt{a}, \mathtt{n}\}$, and $\{\mathtt{b}, \mathtt{n}\}$, we obtain the scattered factors $\mathtt{ba}^3$, $\mathtt{anana}$, and $\mathtt{bn}^2$. Combining all these three scattered factors gives us uniquely $\mathtt{banana}$. Notice that in this example we only needed six binomial coefficients instead of ten, which is the worst case.

3.3.29 Remark. As seen in the example we have not only the word length but also $\binom{w}{\mathtt{x}}$ for all but one $\mathtt{x} \in \Sigma$. Both information give us the remaining single letter binomial coefficient and, hence, we will assume that we know all of them.

For convenience in the following theorem consider $\Sigma = \{1, \ldots, q\}$ for $q > 2$ and set $\alpha := \lfloor \frac{16}{7}\sqrt{n} \rfloor + 5$. In the general case the results by [99] and [72] yield that

$$\sum_{i \in [\alpha]} \frac{1}{i} \frac{(q+1)^{\frac{i}{2}} - 1}{q} \tag{3.3.30}$$

is smaller than the best known upper bound on the number of binomial coefficients sufficient to reconstruct a word uniquely.

The following theorem generalises Theorem 3.3.19 on an arbitrary alphabet.

3.3.31 Theorem. *For uniquely reconstructing a word $w \in \Sigma^*$ of length at least $q-1$, $\sum_{i \in [q]} |w|_i (q+1-i)$ binomial coefficients suffice, which is strictly smaller than (3.3.30).*

Proof. The claim that $\sum_{i \in [q]} |w|_i (q+1-i)$ binomial coefficients suffice to reconstruct w uniquely follows by Theorem 3.3.25: for each pair of letters we apply the method of the binary case. Thus, we are going to reconstruct words $w_{\mathtt{a},\mathtt{b}}$ for all pairs of letters $\mathtt{a} < \mathtt{b}$. If $\mathtt{a}$ is the i^{th} letter in the alphabet, there are $q - i$ such pairs. To determine $w_{\mathtt{a},\mathtt{b}}$ uniquely, $\min(k_{\mathtt{a}}, k_{\mathtt{b}}) + 1 \leqslant k_{\mathtt{a}} + 1$ binomial coefficients from the set $\{k_{\mathtt{b}}, k_{\mathtt{ab}}, \ldots, k_{\mathtt{a}^{|w|_{\mathtt{a}}}\mathtt{b}}\}$ suffice. In total, thus, we need the binomial coefficients of the set

$$\{k_{\mathtt{a}^j\mathtt{b}} : \mathtt{a} < \mathtt{b}, j \in [|w|_{\mathtt{a}}]\} \cup \{k_{\mathtt{b}} : \mathtt{b} \in \Sigma \backslash \{1\}\}.$$

There are $\sum_{i \in [q]} |w|_i (q-i) + (q-1)$ such coefficients. This quantity is less than or equal to $\sum_{i \in [q]} |w|_i (q+1-i)$ for every w of length at least $q-1$.

We show the second claim about the bound by induction on q where the binary case in Theorem 3.3.19 serves as induction basis. This implies

$$\begin{aligned}
\sum_{i \in [q]} |w|_i (q+1-i) &= \left(\sum_{i \in [q-1]} |w|_i (q+1-i) \right) + |w|_q (q+1-q) \\
&= \left(\sum_{i \in [q-1]} |w|_i (q-i) \right) + \sum_{i \in [q-1]} |w|_i + |w|_q
\end{aligned}$$

and therefore

$$\begin{aligned}\sum_{i\in[q]} |w|_i(q+1-i) &\leqslant \left(\sum_{i\in[\alpha]} \frac{1}{i}\frac{q^{\frac{i}{2}}-1}{q-1}\right)+n\\ &= \left(\sum_{i\in[\alpha]} \frac{1}{i}\frac{q(q^{\frac{i}{2}}-1)}{q(q-1)}\right)+n.\end{aligned}$$

On the other hand, we have to compare this quantity with (3.3.30) which can be rewritten as

$$\sum_{i\in[\alpha]} \frac{1}{i}\frac{(q+1)^{\frac{i}{2}}-1}{q} = \sum_{i\in[\alpha]} \frac{1}{i}\frac{(q-1)((q+1)^{\frac{i}{2}}-1)}{q(q-1)}.$$

Thus, the claim is proven, if the substraction of the latter one and the previous one is greater than zero, i.e., we show that

$$\left(\sum_{i\in[\alpha]} \frac{1}{i}\frac{(q-1)((q+1)^{\frac{i}{2}}-1)-q(q^{\frac{i}{2}}-1)}{q(q-1)}\right)-n>0, \text{ i.e.,} \qquad (3.3.32)$$

$$\left(\sum_{i\in[\alpha]} \frac{1}{i}\frac{(q-1)(q+1)^{\frac{i}{2}}-qq^{\frac{i}{2}}+1}{q(q-1)}\right)-n>0. \qquad (3.3.33)$$

With $f(i)=\frac{(q-1)(q+1)^{\frac{i}{2}}-qq^{\frac{i}{2}}+1}{iq(q-1)}$ for all $i\in[\alpha]$, the proof of (3.3.33) contains the following steps

1. For all $i\geqslant 2$ we have $f(i)\geqslant 0$,

2. $f(5)+f(1)\geqslant 0$,

3. $f(\alpha)-n>0$.

ad 1. For $i=2$ we have

$$f(2)=\frac{1}{2}\frac{(q-1)(q+1)-q^2+1}{q(q-1)}=\frac{q^2-1-q^2+1}{2q(q-1)}=0.$$

For $i = 3$ we have

$$f(3) = \frac{1}{3}\frac{(q-1)(q+1)\sqrt{q+1} - q^2\sqrt{q} + 1}{q(q-1)}.$$

Consider the function $g : \mathbb{R} \to \mathbb{R}; q \mapsto q^4 - 2q^3 - 2q^2 + q + 1$. This function has two minima (between -0.75 and -0.5 as well as between 1.75 and 2) and one maximum (between 0.125 and 0.25). Since g has only two inflexion points and g is strictly greater than zero at the first minimum, g has only two roots. The first root is between 0.7 and 0.8 and the second root is between 2.5 and 2.75. Thus, for all $q \geqslant 2.75$, we have $g(q) > 0$. This implies $q^5 + q^4 - 2q^3 - 2q^2 + q + 1 > q^5$. Hence, equivalently we get $(q+1)(q^4 - 2q^2 + 1) > q^5$, i.e., $(q+1)(q^2-1)^2 > qq^4$. This implies $\sqrt{q+1}(q^2-1) > \sqrt{q}q^2$ which proves that the numerator of $f(3)$ is positive and, hence, $f(3) > 0$. Before we prove the claim for $i \geqslant 4$, we will prove that $(q-1)(q+1)^j \geqslant q^{j+1}$ for $j \geqslant 2$. Thus, we get

$$(q-1)(q+1)^j = \left(\sum_{k\in[j]}\left(\binom{j}{k-1} - \binom{j}{k}\right)q^k\right) + q^{j+1} - 1.$$

Due to the central symmetry of each row of the Pascal triangle and since the distribution of the binomial coefficient is unimodal, for $k \leqslant \lfloor j/2 \rfloor$, we have

$$\binom{j}{j-k} - \binom{j}{j-k+1} = -\left(\binom{j}{k-1} - \binom{j}{k}\right) > 0$$

and, thus,

$$(q-1)(q+1)^j = \left(\sum_{k\in[\lfloor j/2\rfloor]}\left(\binom{j}{k} - \binom{j}{k-1}\right)(q^{j-k+1} - q^k)\right) + q^{j+1} - 1.$$

Since $k \leqslant \lfloor j/2 \rfloor$, we have $j - k + 1 > k$ and each term of the above sum is, thus, positive. This shows that $(q-1)(q+1)^j \geqslant q^{j+1}$. This leads to the following estimations for $f(i)$. For $i = 2j$ and $j \geqslant 2$ we get

$$f(i) = \frac{(q-1)(q+1)^j - qq^j + 1}{iq(q-1)} \geqslant \frac{q^{j+1} - q^{j+1} + 1}{iq(q-1)} > 0.$$

Finally, for $i = 2j + 1$ and $j \geqslant 2$ we get

$$\begin{aligned} f(i) &= \frac{(q-1)(q+1)^j\sqrt{q+1} - qq^j\sqrt{q} + 1}{iq(q-1)} \\ &\geqslant \frac{q^{j+1}(\sqrt{q+1} - \sqrt{q}) + 1}{iq(q-1)} > 0. \end{aligned}$$

ad 2. Notice that $\alpha \geqslant 7$ holds and, thus, $f(5)$ is always a summand. For $f(5) + f(1)$ we have to prove

$$\frac{(q-1)(q+1)^2\sqrt{q+1} - qq^2\sqrt{q} + 1}{5q(q-1)} + \frac{(q-1)\sqrt{q+1} - q\sqrt{q} + 1}{q(q-1)} \geqslant 0$$

Thus, we get for the numerator

$$\begin{aligned} &(q-1)(q+1)^2\sqrt{q+1} - qq^2\sqrt{q} + 1 + 5(q-1)\sqrt{q+1} - 5q\sqrt{q} + 5 \\ &= (q-1)\sqrt{q+1}((q+1)^2 + 5) - q\sqrt{q}(q^2 + 5) + 6 \\ &= (q-1)\sqrt{q+1}(q^2 + 2q + 6) - q\sqrt{q}(q^2 + 5) + 6 \\ &= q^3\sqrt{q+1} + q^2\sqrt{q+1} + 4q\sqrt{q+1} - 6\sqrt{q+1} - q^3\sqrt{q} - 5q\sqrt{q} + 6. \end{aligned}$$

We have $q^3\sqrt{q+1} > q^3\sqrt{q}$ and, since $q \geqslant 3$,

$$q^2\sqrt{q+1} \geqslant (6+q)\sqrt{q+1}.$$

Therefore, $q^2\sqrt{q+1} + 4q\sqrt{q+1} \geqslant 6\sqrt{q+1} + 5q\sqrt{q}$ and the numerator is positive.

ad 3. Notice that for fixed i, $f(i)$ is monotonically increasing for increasing q. This implies

$$f(\alpha) \geqslant \frac{2 \cdot 4^{\frac{\alpha}{2}} - 3 \cdot 3^{\frac{\alpha}{2}} + 1}{6\alpha} = \frac{2^{\alpha+1} - 3^{\frac{\alpha}{2}+1} + 1}{6\alpha}.$$

We are going to prove that

$$2^{\alpha+1} - 3^{\frac{\alpha}{2}+1} + 1 > 6\alpha n. \tag{3.3.34}$$

Recall that α is a function of n, given by $\alpha = \lfloor \frac{16}{7}\sqrt{n} \rfloor + 5$.

First, we have $$2^{\alpha+1} - 3^{\frac{\alpha}{2}+1} > 2^{\alpha-1} - 2^{\frac{\alpha}{2}}.$$

Indeed, this inequality is equivalent to

$$2^{\frac{\alpha}{2}}\left(3\cdot 2^{\frac{\alpha}{2}-1}+1\right) > 3^{\frac{\alpha}{2}+1} \quad\Leftrightarrow\quad 2^{\frac{\alpha}{2}-1}+\frac{1}{3} > \left(\frac{3}{2}\right)^{\frac{\alpha}{2}}.$$

We can check that this last inequality is true by taking the logarithm of both sides, since $\alpha > 5$.

Therefore, it is sufficient for (3.3.34) to show that

$$2^{\alpha-1} - 2^{\frac{\alpha}{2}} = 2^{\frac{\alpha}{2}}(2^{\frac{\alpha}{2}-1} - 1) > 6\alpha n.$$

Note that $2^{\frac{\alpha}{2}} > n$ (indeed, $\lfloor \frac{16}{7}\sqrt{n} \rfloor + 5 > 2\sqrt{n} + 5$, thus, $2^{\frac{\alpha}{2}} > 2^{\frac{5}{2}} \cdot 2^{\sqrt{n}}$). Once again, taking the logarithms, one can check that $2^{\frac{5}{2}} \cdot 2^{\sqrt{n}} > n$ holds.

To verify (3.3.34), it remains to show that $2^{\frac{\alpha}{2}-1} - 1 \geqslant 6\alpha$ or that $2^{\frac{\alpha}{2}-1} > 6\alpha$. Taking the logarithms, it is equivalent to

$$\begin{aligned} &\frac{\alpha}{2} - 1 > \log(6) + \log(\alpha) \\ \Leftrightarrow &\alpha - 2\log(\alpha) > 2\log(6) + 2, \end{aligned}$$

which is true for $\alpha \geqslant 15$, that is for $n \geqslant 16$. Equation (3.3.34) can be verified by a computer for $q - 1 \leqslant n < 16$.

By 1., 2., and 3., Equation (3.3.33) is proven and this proves the claim.

□

3.3.35 Remark. Since the estimation in Theorem 3.3.31 depends on the distribution of the letters in contrast to the method of reconstruction, it is wise to choose an order $<$ on Σ such that $x < y$ if $|w|_x \leqslant |w|_y$. In the example we have chosen the *natural* order $\mathtt{a} < \mathtt{b} < \mathtt{n}$ which leads in the worst case to fourteen binomial coefficients that have to be taken into consideration. If we chose the order $\mathtt{b} < \mathtt{n} < \mathtt{a}$ the formula from Theorem 3.3.31 provides that ten binomial coefficients suffice. This observation leads also to the fact that less binomial coefficients suffice for a unique determinism if the letters are not distributed equally but some letters occur much more often than others.

3.3.36 Remark. Let's note that the number of binomial coefficients we need is at most qn. Indeed, we will prove that $\sum_{i\in[q]} |w|_i(q+1-i) \leqslant qn$. We have $qn = qn + n - n = q\sum_{i\in[q]} |w|_i + \sum_{i\in[q]} |w|_i - \sum_{i\in[q]} |w|_i \geqslant q\sum_{i\in[q]} |w|_i +$

$\sum_{i\in[q]} |w|_i - \sum_{i\in[q]}(|w|_i i) = \sum_{i\in[q]} |w|_i(q+1-i)$.

This result finishes not only the reconstruction section but also the chapter about scattered factors.

Chapter 4

Patterns and k-Locality

The results of this chapter are mainly based on [27, 28, 15]. Recall that in the context of patterns, we have a finite alphabet Σ of letters and a possibly infinite alphabet X of variables.

The structure of 1-local and 2-local words is characterised in [27]. The simplest 1-local words (patterns) are repetitions of one letter (variable) x^k for some $k \geqslant 0$ and $x \in \Sigma$ or $x \in X$. Furthermore, if a pattern α is 1-local, then $y^\ell \alpha y^r$ is 1-local, where $y \notin \text{alph}(\alpha), \ell, r \geqslant 0$. Marking sequences for 1-local words can be obtained by going from the "inner-most" letters to the "outer-most" ones. The English words *radar*, *refer*, *blender*, or *rotator* are all 1-local. Generally, in order to have a high locality number, a word needs to contain many alternating occurrences of (at least) two letters. For instance, $(x_1x_2)^n$ is n-local for $x_1, x_2 \in X$ and $n \in \mathbb{N}$. The number of occurrences of a letter alone is not always a good indicator of the locality of a word. The German word *Einzelelement* (basic component of a construction) has 5 occurrences of e, but is only 3-local, as witnessed by marking sequence (*l,m,e,i,n,z,t*). Nevertheless, a repetitive structure often leads to high locality. The Finnish word *tutustuttu* (perfect passive of *tutustua* - to become aquainted with) is nearly a repetition and 4-local, while *pneumonoultramicroscopicsilicovolcanoconiosis* is an (English) 8-local word, and *lentokonesuihkuturbiinimoottoriapumekaanikkoaliupseerioppilas* is a 10-local (Finnish) word.

In this chapter we investigate the complexity of Loc, the problem to determine the locality number, as well as the behaviour of the locality for palindromes and under repetitions. In before we present two helpful definitions: reversing a marking sequence and the notion of near-optimality.

4.0.1 Definition. Given a marking sequence $\sigma = (x_1, \dots, x_n)$ with $x_i \in X$ for $i \in [n]$, $n \in \mathbb{N}$, let σ^R be the marking sequence obtained by reversing σ, i.e., $\sigma^R(i) = \sigma(n - i + 1)$ for $i \in [n]$).

4.0.2 Definition. Let α be the skeleton of a pattern β. We say that a marking sequence σ is *near-optimal* (for α) if $\mathrm{loc}_\sigma(\alpha) \in \{\mathrm{loc}(\alpha), \mathrm{loc}(\alpha) + 1\}$.

4.1 The Hardness of Computing the Locality Number

In this section we prove that Loc is NP-complete. The first result shows that, given two letters x_i, x_j of a word α, it is guaranteed that there exists a near-optimal marking sequence which marks x_i before x_j.

4.1.1 Lemma*. *Let α be a word over the alphabet $X = \{x_1, x_2, \dots, x_n\}$. Let $\sigma : \{1, 2, \dots, |X|\} \to X$ be a marking sequence. Then $|\,\mathrm{loc}_\sigma(\alpha) - \mathrm{loc}_{\sigma^R}(\alpha)| \leqslant 1$.*

Proof. Let $1 \leqslant i \leqslant |X|$ and consider the marking of the first i letters in α according to σ^R. Note that these letters are exactly the last i letters to be marked according to σ. In particular, the number of marked blocks after stage i of marking α according to σ^R corresponds exactly to the number of unmarked blocks – or gaps – after stage $|X| - i$ of marking α according to σ. Since the number of unmarked blocks/gaps can be at most one higher, and at most one lower than the number of marked blocks, the lemma follows immediately. □

We show now, via a many-one reduction, that Loc is NP-hard. To this end, we devise a reduction from the well-known NP-complete Clique problem, i. e., the problem to decide, for a given graph $\mathcal{G} = (V, E)$ and $\ell \in \mathbb{N}$, whether $\mathcal{G}$ contains a clique (i. e., a complete subgraph) of size ℓ.

Let $\mathcal{G} = (V, E)$ be an undirected graph with $V = \{v_1, v_2, \dots, v_n\}$ and let $\ell \in \mathbb{N}$ with $\ell \leqslant n$. Note that the number of edges in a clique of size ℓ is exactly $\mu_\ell = \frac{\ell(\ell-1)}{2}$. We define the alphabet $X = \{x_1, x_2, \dots, x_n, z_1, z_2, z_3\}$ containing a unique letter for each vertex of the graph, along with three extra 'control' letters. Let $d(i)$ denote the degree of each vertex v_i, and let

$\Delta = \max_{1 \leqslant i \leqslant n} \{d(i)\}$. Next, we define the word $\alpha = \alpha_1 \alpha_2 \alpha_3$, where

$$\alpha_1 = (z_1 z_2 z_3 z_2)^{\gamma_1},$$
$$\alpha_2 = (z_1 z_2)^{\gamma_2 (n-\ell)} (x_1 z_2)^{\gamma_2} (x_2 z_2)^{\gamma_2} \dots (x_n z_2)^{\gamma_2} (z_3 z_2)^{\ell \gamma_2} z_3,$$
$$\alpha_3 = \left(\prod_{\{v_i, v_j\} \in E \wedge i<j} (x_i x_j)^{\gamma_3} z_3 \right) \left(\prod_{1 \leqslant i \leqslant n} (x_i z_3)^{\gamma_3 (\Delta - d(i))} \right),$$

and γ_1, γ_2 and γ_3 are chosen such that $\gamma_1 > |\alpha_2 \alpha_3|$, $\gamma_2 > |\alpha_3| + 1$, and $\gamma_3 > 2$. Finally, let $\rho = \gamma_1 + n\gamma_2 + (\ell\Delta - 2\mu_\ell)\gamma_3 + \mu_\ell + 1$.

4.1.2 Lemma. *The word α is ρ-local if and only if $\mathcal{G}$ contains a clique of size ℓ.*

Proof. We first consider some general observations on the k-locality of α. For clarity, and to avoid counting marked blocks more than once, we use the convention that a marked block which starts in α_1 and ends in α_2 (or α_3) belongs to α_2 (or α_3, respectively), and *not* to α_1.

Claim 1. α is $(\gamma_1 + n\gamma_2 + |\alpha_3|)$-local.

Proof. (Claim 1) Consider the marking sequence $z_1, x_1, x_2, \dots, x_\ell, z_2, z_3, x_{\ell+1}, \dots, x_n$. After marking the first letter, z_1, we have $\gamma_1 + \gamma_2(n-\ell)$ blocks. Marking the letters x_i, $1 \leqslant i \leqslant \ell$, introduces exactly $\ell\gamma_2$ additional blocks in α_2 (each single x_i accounting for γ_2 blocks), and altogether, they introduce fewer than $|\alpha_3|$ additional blocks in α_3, resulting always in a total of less than $\gamma_1 + n\gamma_2 + |\alpha_3|$ blocks. Marking z_2 introduces no new blocks in α_1 (the last occurrence is adjacent to the first z_1 in α_2), and joins together $n\gamma_2$ blocks in α_2 while simultaneously introducing $n\gamma_2$ more, giving a net increase of one. Since z_2 does not occur in α_3, no new blocks are introduced there. Thus, we have at most $\gamma_1 + n\gamma_2 + |\alpha_3|$ blocks. Marking z_3 joins all the γ_1 blocks in α_1, and α_1 is completely marked. Since no more than $|\alpha_2\alpha_3|$ blocks can exist elsewhere, and since $\gamma_1 > |\alpha_2\alpha_3|$, all further steps will have less than $\gamma_1 + 1$ marked blocks, so the maximum used is less than $\gamma_1 + n\gamma_2 + |\alpha_3| + 1$ as claimed. □

Claim 2. In any optimal marking sequence, z_2 is marked between z_1 and z_3. Consequently, there exists a near-optimal marking sequence in which z_1 is marked before z_2, which in turn is marked before z_3.

Proof. (Claim 2) If z_2 were the first (resp. last) out of the three to be marked, then α_1 would contain $2\gamma_1 > \gamma_1 + n\gamma_2 + |\alpha_3|$ marked blocks and, thus, by Claim 1, the marking sequence is not optimal. The second statement follows from Lemma 1. □

For the rest of the proof, consider a near-optimal marking sequence in which z_1 is marked before z_2, and z_2 is marked before z_3. Such a sequence exists, by Claim 2. Let ℓ' be the number of x_is which are marked before z_2. If $\ell' < \ell$, then exactly after z_2 is marked, we have γ_1 marked blocks in α_1. The number of marked blocks in the suffix $(z_3 z_2)^{\ell\gamma_2} z_3$ of c α_2 is $\ell\gamma_2$. To count the marked blocks in the rest of α_2 (i.e., the prefix $(z_1 z_2)^{\gamma_2(n-\ell)}(x_1 z_2)^{\gamma_2}(x_2 z_2)^{\gamma_2} \ldots (x_n z_2)^{\gamma_2}$), note that since both ends of this factor are marked, the number of marked blocks is exactly one more than the number of gaps. Moreover, since z_1 and z_2 are marked, the only gaps come from occurrences of the unmarked x_is. Since no two occurrences of these are adjacent, this means that each occurrence is a unique gap so there are $(n - \ell')\gamma_2$ gaps in total. Consequently, α_2 contains exactly $(n - \ell' + \ell)\gamma_2 + 1$ marked blocks. Since $\gamma_2(n - \ell' + \ell) \geqslant \gamma_2(n+1)$, and $\gamma_2 > |\alpha_3| + 1$, this means we have more than $\gamma_1 + n\gamma_2 + |\alpha_3| + 1$ blocks in total. By Claim 1, this contradicts our assumption that the sequence is near optimal. Similarly, if $\ell' > \ell$, then exactly before z_2 is marked, we have γ_1 marked blocks in α_1 and $\gamma_2(n - \ell + \ell') \geqslant \gamma_2(n+1)$ marked blocks in α_2. Again this implies that we have more than $\gamma_1 + n\gamma_2 + |\alpha_3| + 1$ marked blocks altogether, contradicting the assumption that our sequence is near-optimal. Consequently, $\ell' = \ell$, and there exist $i_1, i_2, \ldots, i_\ell$ such that the set of letters marked before z_2 is $\{x_{i_1}, x_{i_2}, \ldots, x_{i_\ell}, z_1\}$.

Now, we observe that after z_2 is marked, the number of marked blocks is never increased. To see why, suppose $z_1, z_2, x_{i_1}, x_{i_2}, \ldots, x_{i_\ell}$ are marked (note that we do not exclude the case that more letters may also be marked). Suppose we mark z_3. Then γ_1 marked blocks will be joined together in α_1, and, hence, decrease the number of marked blocks in α_1 by $\gamma_1 - 1$ (and α_1 is completely marked). Since $\gamma_1 > |\alpha_2\alpha_3|$, the total number of blocks cannot increase overall. Similarly, suppose we mark some x_j, $1 \leqslant j \leqslant n$. Then γ_2 marked blocks are joined together in α_2, thus, reducing the number of marked blocks by $\gamma_2 - 1$. The number of blocks in α_1 remains the same, and since $\gamma_2 > |\alpha_3|$, the total number of marked blocks cannot increase overall.

It is reasonably straightforward to observe that until z_2 is marked, the total number of marked blocks is never decreased (in order to be fully precise, one can make an argument symmetric to the above). Thus, the maximum number of marked blocks in our sequence is obtained (not necessarily for the first time) when z_2 is marked. In other words, if exactly $z_1, z_2, x_{i_1}, x_{i_2}, \ldots, x_{i_\ell}$ are marked, we have the maximal number of blocks. Clearly, this implies that there are γ_1 blocks in α_1 and $n\gamma_2 + 1$ blocks in α_2.

We now consider the number of marked blocks in α_3, which is given by $\gamma_3\,(\Delta\ell - 2t) + t$, where $t = |\{(j, j') \mid 1 \leqslant j < j' \leqslant \ell \wedge \{v_{i_j}, v_{i_{j'}}\} \in E\}|$. To see this, first suppose there are gaps (or a new unmarked letter #) between all adjacent letters. This hypothetical situation would give a total of $\Delta\gamma_3\ell$ blocks. Then consider how many blocks are lost or joined by removing the gaps (or #s). In particular, precisely $2\gamma_3$ blocks are joined together for each pair $x_{i_j}, x_{i_{j'}}$ such that $\{v_{i_j}, v_{i_{j'}}\} \in E$. No further blocks are joined together - so for each such pair we must subtract $2\gamma_3 - 1$ from the total.

Note that t can be at most μ_ℓ and is exactly μ_ℓ if and only if the vertices $v_{i_1}, v_{i_2}, \ldots, v_{i_\ell}$ form a clique. Consequently, if G contains a size-ℓ clique, a (near-optimal) marking sequence can be chosen such that the maximum number of blocks used is $\gamma_1 + n\gamma_2 + \gamma_3(\Delta\ell - 2\mu_\ell) + \mu_\ell + 1 = \rho$. Hence, in this case, α is ρ-local. On the other hand, if G does not contain a size-ℓ clique, then regardless of the choice of $x_{i_1}, x_{i_2}, \ldots, x_{i_\ell}$, we have $t \leqslant \mu_\ell - 1$, and any near optimal marking sequence requires at least

$$\begin{aligned}\gamma_1 + n\gamma_2 + \gamma_3\,(\Delta\ell - 2\mu_\ell + 2) + \mu_\ell &= \gamma_1 + n\gamma_2 + (\Delta\ell - 2\mu_\ell)\,\gamma_3 + 2\gamma_3 + \mu_\ell \\ &> \rho\end{aligned}$$

marked blocks, meaning α is not ρ-local. Thus, α is ρ-local if and only if G contains a size-ℓ clique. Since α and ρ can be constructed in polynomial time, the theorem follows. □

4.1.3 Theorem. *Since* Loc *is obviously in* NP*, we get that* Loc *is* NP*-complete.*

4.2 Locality of Palindromes and Repetitions

In this section we investigate the locality of palindromes and repetitions motivated by the fact that both sets of words follow a structure which may help to determine the locality more easily.

Recall that in any case it suffices to consider condensed words since any factor x^k for $x \in X$ and $k \in \mathbb{N}$ would be marked in a single stage simultaneously.

4.2.1 Remark. An important observation is that condensed palindromes of even length do not exist (the even length would imply that the both letters surrounding the middle are the same). Thus, only palindromes of odd length are of interest when determining the locality number.

4.2.2 Lemma. *Define the morphism $f : X \cup \overline{X} \to \{0,1\}$ by*

$$f(x) = \begin{cases} 0 & \textit{if } x \in X, \\ 1 & \textit{if } x \in \overline{X}. \end{cases}$$

If w is a palindrome and σ a marking sequence for w then $f(w_i)$ is a palindrome for all $i \in [|\operatorname{var}(w)|]$.

Proof. Let $w = uxu^R$ be a palindrome with $u \in X^*$ and $x \in X \cup \overline{X}$ and $|w| = n \in \mathbb{N}$. Moreover, let σ be a marking sequence for w and $i \in [|\operatorname{var}(w)|]$. Since w is a palindrome, $w[j] = w[n-j]$. This implies $w_i[j], w_i[n-j]$ are both either in X or in $\overline{X}$. Thus, either are both mapped to 0 or to 1. Consequently, $f(w_i)$ is a palindrome. □

Before we can present our results, we need to recall the definition of border priority markable from [27].

4.2.3 Definition. A strictly k-local word $w = avb \in XX^*X$ is called *border priority markable* (bpm) if there exists a marking sequence σ of w such that in every stage $i \in [|\operatorname{var}(w)|]$ of σ where k blocks are marked, a and b are marked as well. Analogously *right-border priority markable* and *left-border priority markable* are defined: A strictly k-local word $w = avb \in XX^*X$ is called right-border priority markable (rbpm) if there exists a marking sequence σ of w such that in every stage $i \in [|\operatorname{var}(w)|]$ of σ where k blocks are marked, b is marked as well - respectively, for left-border priority markable, a is marked as well.

4.2.4 Remark. If $w \in X^*$ is right-border priority markable, then w^R is left-border priority markable.

4.2.5 Lemma. *Let $w = u\mathtt{a}u^R$ be an odd-length condensed palindrome with $u \in X^*$ and $\mathtt{a} \in X$. Let u be strictly k-local witnessed by the marking sequence σ.*

▷ *If u is rbpm then* $\operatorname{loc}(w) = 2k - 1$,

▷ *if u is not rbpm and* $\mathtt{a} \notin \operatorname{var}(u)$ *then* $\operatorname{loc}(w) = 2k$,

▷ *if u is not rbpm and* $\mathtt{a} \in \operatorname{var}(u)$ *and for all optimal marking sequences for u there exists a stage $i \in [|\operatorname{var}(u)|]$ such that* $\mathtt{a}$ *is marked, k blocks are marked, and $u[|u|]$ is unmarked then* $\operatorname{loc}(w) = 2k + 1$, *and*

▷ *else* $\operatorname{loc}(w) = 2k$.

Proof. Let σ be an optimal marking sequence of u. If $\mathtt{a} \in \operatorname{var}(u)$ then σ is a marking sequence for w. Marking w w.r.t. σ leads to $\operatorname{loc}_\sigma(w) \leqslant 2k+1$ since there are at most k blocks marked each in u and u^R, and, additionally, the single $\mathtt{a}$ in the middle. If $\mathtt{a} \notin \operatorname{var}(u)$ then $\sigma' = \sigma \cup \{(|u|+1, \mathtt{a}\}$ is a marking sequence for w with $\operatorname{loc}_{\sigma'}(w) \leqslant 2k$, since by marking w.r.t. σ maximal k blocks are marked by σ each in u and u^R and afterwards on marking a two blocks are joined. Thus, in any case $\operatorname{loc}(w) \leqslant 2k+1$.

case 1: Consider u to be rbpm. Thus, in every stage $i \in [|\operatorname{var}(u)|]$ where k blocks are marked, $u[|u|]$ is marked. This implies that $\operatorname{loc}_\sigma(w) \leqslant 2k-1$ or $\operatorname{loc}_{\sigma'}(w) \leqslant 2k-1$ with σ' defined as above.

Supposition: $\operatorname{loc}(w) =: \ell < 2k-1$

Let μ be an optimal marking sequence for w. Then μ is also a marking sequence for u and, thus, $\operatorname{loc}_\mu(u) \geqslant k$. By $\operatorname{loc}(u) = k$, there exists a stage $i \in [|\operatorname{var}(w)|]$ of μ such that k blocks are marked in u, or more precisely $|\operatorname{cond}(f(u_i))|_1 = k$. On the other hand $|\operatorname{cond}(f(w_i))|_1 \leqslant \ell$. Since u is rbpm $u[|u|]$ is marked. If x is not marked, $|\operatorname{cond}(f(u_i))|_1 \leqslant \frac{\ell}{2} < \frac{2k-1}{2} = k - \frac{1}{2}$. If x is marked, $|\operatorname{cond}(f(u_i))|_1 \leqslant \frac{\ell-1}{2} < \frac{2k-2}{2} = k-1$. This is in both cases a contradiction to $|\operatorname{cond}(f(u_i))|_1 = k$.

case 2: Consider now that u is not rbpm. Thus, there exists a stage $i \in [|\operatorname{var}(u)|]$ in which k blocks are marked but $u[|u|]$ is unmarked. If $\mathtt{a}$ is not in $\operatorname{var}(u)$ marking $\mathtt{a}$ before stage i leads to $2k+1$ blocks for the largest such i. Considering σ' then at the beginning u and u^R are completely marked and in the end two blocks are joined by marking $\mathtt{a}$. This leads to $\operatorname{loc}(w) \leqslant 2k$.

Supposition: $\operatorname{loc}(w) < 2k$

As described, $\mathtt{a}$ needs to be marked after the last stage where in u k blocks are marked without $u[|u|]$ being marked. But this sums up to k blocks

marked in u and k blocks marked in u^R, hence, overall $2k$ blocks. This concludes the case $\mathtt{a} \notin \mathrm{var}(u)$.
Consider $\mathtt{a} \in \mathrm{var}(u)$ and assume that $\mathtt{a}$ is marked by σ when k blocks are marked in u and $u[|u|]$ is unmarked. Thus, $\mathrm{loc}_\sigma(w) = 2k+1$.
Supposition: $\mathrm{loc}(w) =: \ell < 2k+1$
Let μ be an optimal marking sequence for w.
Additional supposition: μ not optimal for u
Then there exists a stage $i \in [|\mathrm{var}(w)|]$ such that $|\mathrm{cond}(f(u_i))|_1 = k+1$. If $\mathtt{a}$ is unmarked in this stage, $|\mathrm{cond}(f(w_i))|_1 = 2k+2 > \ell$ which contradicts the first supposition. If $\mathtt{a}$ is marked in this stage $|\mathrm{cond}(f(w_i))|_1 = 2k+1$ which contradicts the first supposition.
Thus, μ is optimal for u. By assumption, there exists a stage $i \in [|\mathrm{var}(u)|]$ such that $\mathtt{a}$ is marked, k blocks are marked, and $u[|u|]$ is unmarked. This implies since $\mathrm{cond}(f(w_i))$ is a palindrome that at most $\frac{\ell-1}{2}$ blocks are marked in u. Thus, $k \leqslant \frac{\ell-1}{2} < \frac{2k+1-1}{2} = k$.
case 3: In the remaining case u is not rbpm, $\mathtt{a} \in \mathrm{var}(u)$, and there exists an optimal marking sequence for u such that in every stage $\mathtt{a}$ is unmarked or less than k blocks are marked or $u[|u|]$ is marked. Let σ be such a marking sequence. Then $\mathrm{loc}_\sigma(w) = 2k$.
Supposition: $\mathrm{loc}(w) =: \ell < 2k$
Let μ be an optimal marking sequence for w. Since u is not rbpm there exists a stage $i \in [|\mathrm{var}(u)|]$ such that $|\mathrm{cond}(f(u_i))|_1 = k$ and $u[|u|]$ is unmarked. If $\mathtt{a}$ were unmarked in stage i, $k = |\mathrm{cond}(f(u_i))|_1 \leqslant \frac{\ell}{2} < k$ and if $\mathtt{a}$ were marked in stage i, $k = |\mathrm{cond}(f(u_i))|_1 \leqslant \frac{\ell-1}{2} < \frac{2k-1}{2} = k - \frac{1}{2}$. Thus, $2k+1 \leqslant \ell < 2k$ would hold. □

The following lemma investigates the behaviour of the locality if a word is repeated.

4.2.6 Lemma. *Let $w = u^i$ be the i-times repetition for $u \in X^*$ and $i \in \mathbb{N}$. If u is strictly k-local then*

$$\mathrm{loc}(w) = \begin{cases} ik - i + 1, & \text{if } u \text{ is bpm,} \\ ik, & \text{otherwise.} \end{cases}$$

Proof. Let σ be a marking sequence with $\mathrm{loc}_\sigma = \mathrm{loc}(u) = k$. By $\mathrm{alph}(u) = \mathrm{alph}(u^i)$, for all $i \in \mathbb{N}$, σ is also a marking sequence for w. If u is not bpm, there exists a stage during the marking in which k blocks are marked by σ and at least one of $u[1]$ or $u[|u|]$ is unmarked. Thus, marking w according to the sequence σ leads to $\mathrm{loc}_{\sigma}(w) = ik$.

If u is bpm, in any stage in which k blocks are marked, $u[1]$ and $u[|u|]$ are marked and, thus, in w, while being marked according to σ, the last marked block of an occurrence of u and the first marked block of the next occurrence of u coincide, as soon as the prefix of length $|u|$ of w contains k marked blocks. So, we get $\mathrm{loc}_{\sigma}(w) = ik - i + 1$.

For proving $\mathrm{loc}(w) = ik$ or $\mathrm{loc}(w) = ik - i + 1$ respectively, first consider $i = 2$. Assume first that w is bpm. Suppose $\mathrm{loc}(w) = \ell < 2k - 1$. Let σ' be the marking sequence witnessing $\mathrm{loc}(w) = \ell$. Since u is strictly k-local, there exists a stage in marking w by σ' in which u has k marked blocks. The second u has exactly as many marked blocks as the first one, so also k. In the best case, in w the last marked block of the first u and the first marked block of the second u are connected. Anyway, the number of marked blocks of w is, in that case, exactly $2k - 1$. A contradiction to the assumption $\mathrm{loc}(w) = \ell < 2k - 1$. If u is not bpm, then, once again, there exists a stage in marking w by σ' in which u has k marked blocks. The second u has also exactly k marked blocks. But, in this case, in w the last marked block of the first u and the first marked block of the second u do not touch (as either the last letter of u or its first letter are not marked). So w has $2k$ marked blocks, a contradiction.

This reasoning can be trivially extended for $i > 2$. □

The well-known *Zimin words* [80] also have high locality numbers compared to their lengths. These words are important in the domain of avoidability, as it was shown that a terminal-free pattern is unavoidable (i.e., it occurs in every infinite word over a large enough finite alphabet) if and only if it occurs in a Zimin word.

4.2.7 Definition. The Zimin words Z_i, for $i \in \mathbb{N}$, are inductively defined by $Z_1 = x_1$ and $Z_{i+1} = Z_i x_{i+1} Z_i$.

Clearly, $|Z_i| = 2^i - 1$ for all $i \in \mathbb{N}$. Regarding the locality of Z_i, note that marking x_2 leads to 2^{i-2} marked blocks; further, marking x_1 first and then the remaining symbols in an arbitrary order only extends or joins marked blocks. Thus, we obtain a sequence with locality 2^{i-2}. In fact, we have $\mathrm{loc}(Z_i) = \frac{|Z_i|+1}{4} = 2^{i-2}$ for $i \in \mathbb{N}_{\geqslant 2}$. Notice that both Zimin words and 1-local words have an obvious palindromic structure. However, in the Zimin words the letters occur multiple times, but not in large blocks, while in 1-local words there are at most 2 blocks of each letter.

4.2.8 Proposition. *For all $i \in \mathbb{N}_{\geqslant 2}$ we have* $\mathrm{loc}(Z_i) = \frac{|Z_i|+1}{4} = 2^{i-2}$.

Proof. Clearly, x_1 and $x_1x_2x_1$ are 1-local. Consider a fixed $i \in \mathbb{N}$ and the marking sequence $(x_2, x_1, y_1, y_2, \ldots, y_{i-2})$ for $i \geqslant 3$ and $\{y_1, \ldots, y_{i-2}\} = \{x_3, \ldots, x_i\}$. Notice that for all $j \in \mathbb{N}$, x_j occurs 2^{i-j} times in Z_i. Thus, by marking x_2, there are 2^{i-2} marked blocks. Since all occurrences of x_1 are adjacent to occurrences of x_2, marking x_1 does not change the number of marked blocks. As marking the remaining variables only leads to the merging of some pairs of consecutive blocks into one, we never have more than 2^{i-2} marked blocks.

In the following we will show the converse. More precisely, we show that if a sequence is optimal for Z_i then it starts with x_2, x_1. Let us note first that, for $2 \leqslant p < r$, between two consecutive occurrences of x_r in Z_i there is one occurrence of x_p. More precisely, each occurrence of a variable x_p, with $p \geqslant 2$, is directly between two occurrences of x_1. Also, notice that x_j has 2^{i-j} occurrences in Z_i. Now, if x_1 is marked before x_2, because Z_i starts with x_1x_2 and ends with x_2x_1, it is immediate that after the marking of x_1 we will have at least $2^{i-2}+1$ marked blocks in the word (separated by the 2^{i-2} unmarked occurrences of x_2). This is, thus, a marking sequence that is not optimal. So x_2 is marked before x_1 in an optimal sequence. Assume that there exists x_j, with $j > 2$, which is also marked before x_1 in an optimal sequence. Let w be a word such that $Z_i = x_1wx_1$. There are $2^{i-1}-2$ occurrences of x_1 in w, and w starts with x_2x_1 and ends with x_1x_2. As each two consecutive (marked) occurrences of the letters x_2 and x_j are separated by unmarked occurrences of x_1 we have that, just before marking x_1, there are at least $\min\{2^{i-1}-1, 2^{i-2}+2^{i-j}\}$ marked blocks in w (and the same number in Z_i). This again shows that this is not an optimal marking sequence. So, before x_1 is marked, only x_2 should be marked. This concludes the proof of our claim, and of the proposition. □

4.3 The Hardness of Matching Repetitions

In this last section of Chapter 4 we present again an NP hardness proof regarding patterns. Whereas for regular patterns, i.e., each variable occurs only once, there exist quite efficient matching algorithms (cf. [39]), we prove the rather strong negative result that even to decide whether $\beta_1\beta_2$ for patterns $\beta_1, \beta_2 \in (\Sigma \cup X)^*$ matches a word $w \in \Sigma^*$ is NP-hard. We are going to reduce the *perfect code problem* for 3-regular graphs to the matching problem of this specific kind (similar to reductions in [41, 84]).

4.3.1 Definition. An undirected graph is defined by $G = (V, E)$ with the set of vertices $V = \{t_1, \ldots, t_n\}$, $n \in \mathbb{N}_0$, and $E \subseteq \{U \subseteq V \mid |U| = 2\}$. The closed neighbourhood of a vertex $v \in V$ for a graph G is defined by $N_G[v] = \{u \in V \mid \{u, v\} \in E\} \cup \{v\}$ and the edge-degree in G for a given vertex $v \in V$ is defined by $\deg_G(v) = |N_G[v]| - 1$. A graph is called 3-regular if $\deg_G(v) = 3$ holds for all $v \in V$. A vertex set $C \subseteq V$ is a perfect code for G if, for every $v \in V$, $|N_G[v] \cap C| = 1$. Notice that the cardinality of a perfect code is given by $\frac{n}{4}$. Let $G = (V, E)$ with $V = \{t_1, \ldots, t_n\}$ be a 3-regular graph. To get a convenient access to the neighbours of a given $v \in V$ define for $r \in [4]$ the (not unique) mappings $\wp_r : [n] \to [n]$ where $\wp_r(i) = j$ indicates that the r$^{\text{th}}$ neighbour of t_i is t_j (they are assumed to be arbitrary but fixed).

Define for a given graph G a pattern matching instance: let

$$X = \{x_{i,j} \mid i, j \in [n]\} \cup \{y_i, y'_i \mid i \in [n]\}$$

be the set of variables and $\Sigma = \{\mathtt{a}, \#, \star\}$ be the set of terminal symbols. For all $i \in [n]$ set

$$\begin{aligned}
\alpha_i &= x_{\wp_1(i),i} \cdots x_{\wp_4(i),i},\\
\alpha'_i &= y_i \# x_{i,\wp_1(i)} \cdots x_{i,\wp_4(i)} \# y'_i,\\
w_i &= \mathtt{a}^5, w'_i = (\#\mathtt{a}^8)^2\#(\mathtt{a}^4\#)^2,\\
\beta_1 &= \alpha_1 \star \cdots \star \alpha_n \star y_1 \cdots y_n y'_1 \cdots y'_n \star,\\
\beta_2 &= \alpha'_1 \star \cdots \star \alpha'_n,\\
v_1 &= w_1 \star \cdots \star w_n \star (\#\mathtt{a}^8)^{2n - \frac{n}{4}} (\mathtt{a}^4\#)^{n + \frac{n}{4}} \star,\\
v_2 &= w'_1 \star \cdots \star w'_n.
\end{aligned}$$

Finally, set $\beta = \beta_1\beta_2$ and $v = v_1v_2$. Notice that, in contrast to β, v contains only terminal symbols and, hence, (β, v) is a pattern matching instance.

By definition, β_1 and β_2 are regular and, since, for every $i, j \in [n]$, $x_{i,j}$ occurs in α_j if and only if it occurs in α'_i, and all variables y_i, y'_i, $i \in [n]$ occur in both β_1 and β_2, we get $\mathrm{var}(\beta_1) = \mathrm{var}(\beta_2)$, i. e., the skeleton of β_1 is a permutation of β_2's skeleton. This, alongside the next result, allows us to reach the conclusion.

4.3.2 Lemma. *The patterns β_1 and β_2 are regular and their projections onto the set of variables are abelian equivalent, i.e., they are permutations of each other.*

Proof. If $x_{i,j}$ for $i, j \in [n]$ is a variable of β_1 then there exists $k \in [4]$ with $i = \wp_k(j)$. On the other hand, $\alpha'_{\wp_k(j)}$ is a factor of β_2 and, consequently, $x_{\wp_k(j),\wp_{k'}(\wp_k(j))}$ is a variable in β_2 for all $k' \in [4]$. Since i and j are adjacent there exists $\hat{k} \in [4]$ with $\wp_{\hat{k}}(i) = j$. Thus, $x_{i,j} = x_{i,\wp_{\hat{k}}(i)} = x_{\wp_k(j),\wp_{\hat{k}}(\wp_k(j))}$ occurs in β_2. Since in both β_1 and β_2 the number of occurrences of x's (with some subscript) is each $4n$ and each occurrence in β_1 is also in β_2, both contain exactly the same variables x_{ij} for some i, j. Additionally, both patterns have each exactly one occurrence of y_i and y'_i for $i \in [n]$. Thus, the skeletons of β_1 and β_2 are abelian equivalent. □

4.3.3 Lemma. *The graph G has a perfect code if and only if β matches v.*

Proof. First, assume that G has a perfect code C. Define the substitution $h : (\Sigma \cup X)^* \to \Sigma^+$ by $h(\ell) = \ell$ for all $\ell \in \Sigma$ and

$$x_{i,\wp_r(i)} \mapsto \begin{cases} \mathtt{a}^2 & \text{if } t_i \in C, \\ \mathtt{a} & \text{otherwise,} \end{cases}$$

$$y_i \mapsto \begin{cases} \#\mathtt{a}^8 & \text{if } t_i \in C, \\ \#\mathtt{a}^8\#\mathtt{a}^8 & \text{otherwise,} \end{cases} \qquad y'_i \mapsto \begin{cases} \mathtt{a}^4\#\mathtt{a}^4\# & \text{if } t_i \in C, \\ \mathtt{a}^4\# & \text{otherwise,} \end{cases}$$

for all $i \in [n], r \in [4]$. Since C is a perfect code, exactly for one $r \in [4]$, $x_{\wp_r(i),i}$ is mapped to $\mathtt{a}^2$ for each $i \in [n]$. Thus, w_i matches α_i for all $i \in [n]$. Moreover, the definition of h implies immediately $h(x_{i,\wp_1(i)} \cdots x_{i,\wp_4(i)}) \in \{\mathtt{a}^8, \mathtt{a}^4\}$ for all $i \in [n]$. In the first case y_i is mapped to $\#\mathtt{a}^8$ and y'_i to $(\mathtt{a}^4\#)^2$ whereas in the second case y_i is mapped to $(\#\mathtt{a}^8)^2$ and y'_i to $\mathtt{a}^4\#$ for all $i \in [n]$. Hence, α'_i matches w'_i. Finally, due to the fact that exactly $|C|$ many of the y_i are set to $\#\mathtt{a}^8$ (and the remaining ones to $\#\mathtt{a}^8\#\mathtt{a}^8$) and $|C|$ many of the y'_i are set to $\mathtt{a}^4\#\mathtt{a}^4\#$ (and the remaining ones to $\mathtt{a}^4\#$), and $|C| = \frac{n}{4}$, we can conclude that $(\#\mathtt{a}^8)^{2n-\frac{n}{4}}(\mathtt{a}^4\#)^{n+\frac{n}{4}}$ matches $y_1 \cdots y_n y'_1 \cdots y'_n$. Overall this proves that β matches v.

Assume now that (β, v) is a yes-instance of the pattern matching problem, i.e., there exists a substitution $h : (\Sigma \cup X)^* \to \Sigma^+$ with $h(\ell) = \ell$ for all $\ell \in \Sigma$ and $h(\beta) = v$. Since there are as many occurrences of $\star$ in β as in v, we can conclude that, for every $i \in [n]$, $h(\alpha_i) = w_i$ and $h(\alpha'_i) = w'_i$. Consequently, for all $i \in [n]$ there exists exactly one $r \in [n]$ with $h(x_{\wp_r(i),i}) = \mathtt{a}^2$. Thus, $h(x_{\wp_s(i),i}) = \mathtt{a}$ follows for all $s \in [4]\backslash\{r\}$. Moreover, since $h(\alpha'_i) = \#\mathtt{a}^8\#\mathtt{a}^8\#\mathtt{a}^4\#\mathtt{a}^4\#$, $i \in [n]$, we can also conclude that $h(x_{i,\wp_1(i)} \cdots x_{i,\wp_4(i)}) \in \{\mathtt{a}^4, \mathtt{a}^8\}$, which implies that $h(y_i) \in \{\#\mathtt{a}^8, (\#\mathtt{a}^8)^2\}$ and $h(y'_i) \in \{\mathtt{a}^4\#, (\mathtt{a}^4\#)^2\}$ for all $i \in [n]$. Set

$$C = \{t_i \in V \mid i \in [n], \ell \in [4], h(x_{i,\wp_\ell(i)}) = \mathtt{a}^2\}.$$

Since $h(x_{\wp_1(i),i} \cdots x_{\wp_4(i),i}) = \mathtt{a}^5$ holds for all $i \in [n]$, for all these $i \in [n]$ exists exactly one $r \in [4]$ with $h(x_{\wp_r(i),i}) = \mathtt{a}^2$ and for all $s \in [4]\backslash\{r\}$ follows $h(x_{\wp_s(i),i}) = \mathtt{a}$. Thus, $h(x_{\wp_r(i),\wp_1(\wp_r(i))} \cdots x_{\wp_r(i),\wp_4(\wp_r(i))}) = \mathtt{a}^8$ holds and $t_{\wp_r(i)}$ is in C whereas $t_{\wp_s(i)}$ is not in C for all $s \in [4]\backslash\{r\}$. This proves that C is a perfect code of G.

□

4.3.4 Theorem. *Deciding whether a pattern $\beta_1\beta_2$, with* $\mathrm{var}(\beta_1) = \mathrm{var}(\beta_2)$ *and regular patterns β_1, β_2, matches a word w is* NP*-hard.*

Proof. Follows directly by Lemma 4.3.3. □

4.3.5 Remark. The reduction from above can easily be modified for erasing substitutions: Set $w_i = \mathtt{a}$ and $w'_i = \#\mathtt{a}^4\#\#$, and the factor matched against $y_1 \ldots y_n y'_1 \ldots y'_n$ is then $(\#\mathtt{a}^4)^{n-\frac{n}{4}}\#^{\frac{n}{4}}$. The proof is analogous.

Chapter 5

Prefix Normal Words

This chapter is mainly based on [46]. Recall that for prefix normality we only consider $\Sigma = \{0, 1\}$ and that a word is called prefix normal if $p_w = f_w$, i.e., every factor of a given length has at most as many 1s as the prefix of the same length. The least representative is the lexicographically smallest element within the class of equivalent words, i.e., words with the same maximum-ones function.

5.1 Least Representatives and Prefix Normal Palindromes

Before we present specific properties of the least representatives for a given word length, we mention some useful properties of the maximum-ones, prefix-ones, and suffix-ones functions (for the basic properties we refer to [44, 14] and the references therein). Since we are investigating only words of a specific length, we fix $n \in \mathbb{N}_0$.

Beyond the relation $p_w = s_{w^R}$ the mappings p_w and s_w are determinable from each other. Counting the 1s in a suffix of length i and adding the 1s in the corresponding prefix of length $(n - i)$ of a word w, gives the overall amount of 1s of w, namely

$$p_w(n) = p_w(n - i) + s_w(i) \quad \text{and} \quad s_w(n) = p_w(i) + s_w(n - i).$$

For suffix (resp. prefix) normal words this leads to $p_w(i) = f_w(n) - f_w(n - i)$ resp. $s_w(i) = f_w(n) - f_w(n - i)$ witnessing the fact $p_w = s_w$ for palindromes (since both equation hold). Before we show that indeed prefix normal palindromes form a singleton class w.r.t. $\equiv_n$, we need the relation between the lexicographical order and prefix and suffix normality.

5.1.1 Lemma. *The prefix normal form of a class is the lexicographically largest element in the class and the suffix-normal form is the least representative.*

Proof. Let $w \in \Sigma^n$ be the prefix normal form of the class $[w]_\equiv$. Suppose there existed $v \in [w]_\equiv$ with $v > w$. Let $i \in [n]$ be the smallest index with $v[i] \neq w[i]$. Since we are only considering binary alphabets we get $v[i] = 1$ and $w[i] = 0$. By the prefix normality of w, we have $f_w(i) = p_w(i) = p_w(i-1)$ but on the other hand, $v \in [w]_\equiv$ and the minimality of i implies

$$f_v(i) = f_w(i) = p_w(i-1) = p_v(i-1) = p_v(i) - 1 \leqslant f_v(i) - 1 < f_v(i).$$

This contradiction shows that the prefix normal form of a class is the lexicographically largest element. The reverse w^R is, thus, the lexicographically smallest element of the class which is, by definition, the least representative. By Remark 2.4.3, follows

$$s_{w^R} = p_w = f_w = f_{w^R}$$

and, hence, the suffix normality of the least representative.. □

Lemma 5.1.1 implies that a word being prefix and suffix normal forms a singleton class w.r.t. $\equiv_n$. As mentioned $p_w = s_w$ only holds for palindromes.

5.1.2 Proposition. *For a word $w \in \Sigma^n$ it holds that $|[w]_\equiv| = 1$ iff $w \in \mathrm{NPal}(n)$.*

Proof. Already in [14], the authors proved that $|[w]_\equiv| = 1$ implies $w \in \mathrm{NPal}(n)$ for $w \in \Sigma^n$. The other direction follows from Lemma 5.1.1: if w is a prefix normal palindrome it is, by definition, prefix normal and, by $w = w^R$, w is the lexicographically largest and smallest element of the class. This implies that the class is a singleton. □

The general part of this section is concluded by a somewhat artificial equation which is nevertheless useful for prefix normal palindromes : by $s_w(i) = p_w^R(i) - p_w^R(i+1) + s_w(i-1)$ with $p_w^R(n+1) = 0$ for $i \in [n]$ and $s_w = p_{w^R}$, we get

$$p_{w^R}(i) = p_w^R(i) - p_w^R(i+1) - p_{w^R}(i-1).$$

The rest of the section will cover properties of the least representatives of a class.

5.1.3 Remark. For completeness, we mention that 0^n is the only even least representative w.r.t. $\equiv_n$ and the only prefix normal palindrome starting with 0. Moreover, 1^n is the largest least representative. As we show later in this work 0^n and 1^n are of minor interest in the recursive process to determine representatives for the classes due to their speciality.

The following lemma is an extension of [14, Lemma 1] for the suffix-one function by relating the prefix and the suffix of the word s_w for a least representative. Intuitively the suffix normality implies that the 1s are more at the end of the word w rather than at the beginning: consider for instance $s_w = 1123345$ for $w \in \Sigma^7$. The associated word w cannot be suffix normal since the suffix of length two has only one 1 ($s_w(2) = 1$) but, by $s_w(5) = 3$, $s_w(6) = 4$, and $s_w(7) = 5$ we get that within two letters two 1s are present and, consequently, $f_w(2) \geqslant 2$. Thus, a word w is only least representative if the amount of 1s at the end of s_w does not exceed the amount of 1s at the beginning of s_w.

5.1.4 Lemma. *Let $w \in \Sigma^n$ be a least representative. Then we have*

$$s_w(i) \geqslant \begin{cases} s_w(n) - s_w(n-i+1) & \text{if } s_w(n-i+1) = s_w(n-i), \\ s_w(n) - s_w(n-i+1) + 1 & \text{otherwise.} \end{cases}$$

Proof. Since $w \in \Sigma^n$ is least representative we have $f_w(i) = s_w(i) = |\operatorname{Suff}_i(w)|_1$ and $|\operatorname{Suff}_i(w)|_1 \geqslant |\operatorname{Pref}_i(w)|_1$ for all $i \in [n]$. Let $i \in [n]$, $|\operatorname{Suff}_{n-i}(w)|_1 = s$, and $|\operatorname{Pref}_i(w)|_1 = r$. This implies $s_w(n-i) = s$ and $s_w(n) = s + r$. If $s_w(n-i+1) = s_w(n-i)$ then $w[i] = 0$ and $s_w(n-i+1) = s$. This implies

$$s_w(i) = |\operatorname{Suff}_i(w)|_1 \geqslant |\operatorname{Pref}_i(w)|_1 = r = s + r - s = s_w(n) - s_w(n-i+1).$$

If $s_w(n-i+1) \neq s_w(n-i)$ then $w[i] = 1$, $s_w(n-i+1) = s + 1$ and

$$\begin{aligned} s_w(i) &= |\operatorname{Suff}_i(w)|_1 \geqslant |\operatorname{Pref}_i(w)|_1 = r = s + r - s = s_w(n) - s_w(n-i) \\ &= s_w(n) - s_w(n-i+1) + 1. \end{aligned}$$

This concludes the proof. □

The remaining part of this section presents results for prefix normal palindromes. Notice that for $w \in \operatorname{NPal}(n)$ with $w = \mathrm{x}v\mathrm{x}$ with $\mathrm{x} \in \Sigma$, v is not necessarily a prefix normal palindrome; consider for instance $w = 10101$

with 010 $\in$ Pal(3)\NPal(3). The following lemma shows a result for prefix normal palindromes which is folklore for palindromes substituting f_w by p_w or s_w.

5.1.5 Lemma*. *For $w \in \mathrm{NPal}(n)\backslash\{0^n\}$, $v \in \mathrm{Pal}(n)$ with $w = 1v1$ we have*

$$f_w(k) = \begin{cases} 1 & \text{if } k = 1, \\ f_v(k-1)+1 & \text{if } 1 < k \leqslant |w|-1, \\ f_w(|v|+1)+1 & \text{if } k = |w|. \end{cases}$$

Proof. For $k = 1$ we have $f_w(1) = |\mathrm{Pref}_1(w)|_1 = 1$ since $w \neq 0^n$. If $k \in [|w|-1]_{>1}$ then we get

$$f_v(k-1)+1 = |\mathrm{Pref}_{k-1}(v)|_1 + 1 = |\mathrm{Pref}_k(w)|_1 = p_w(k).$$

Finally, we have

$$\begin{aligned} f_w(|v|+1)+1 &= |\mathrm{Pref}_{|v|+1}(w)|_1 + 1 = |\mathrm{Pref}_{k-1}(w)|_1 + 1 \\ &= f_w(k-1)+1 = f_w(k) \end{aligned}$$

for $k = |w|$. □

In the following we give a characterisation of when a palindrome w is prefix normal depending on its maximum-ones function f_w and a derived function $\overline{f_w}$. In particular we observe that $f_w = \overline{f_w}^R$ if and only if w is a prefix normal palindrome. Intuitively $\overline{f_w}$ captures the progress of f_w in reverse order. This is an intriguing result because it shows that properties regarding prefix and suffix normality can be observed when f_w, s_w, p_w are considered in their serialised representation.

5.1.6 Definition. For $w \in \Sigma^n$ define $\overline{f}_w : [n] \to [n]$ by

$$\overline{f}_w(k) = \overline{f}_w(k-1) - (f_w(k-1) - f_w(k-2))$$

with the extension $f_w(-1) = f_w(0) = 0$ of f and $\overline{f}_w(0) = f_w(n)$. Define $\overline{p}_w$ and $\overline{s}_w$ analogously.

5.1.7 Example. Consider the prefix normal palindrome $w = 11011$ with $f_w = 12234$. Then $\overline{f}_w$ is 43221 and we have $f_w = \overline{f}_w^R$. On the other hand for $v = 101101 \in \mathrm{Pal}(6)\backslash\mathrm{NPal}(6)$ we have $p_v = 112334$ and $f_v = 122334$ and $\overline{f}_v = 432211$ and, thus, $\overline{f}_v^R \neq f_v$.

The following lemma shows a connection between the reversed prefix-ones function and the suffix-ones function that holds for all palindromes.

5.1.8 Lemma. *For $w \in \mathrm{Pal}(n)$ we have $s_w \equiv_n \overline{p}^R_w$.*

Proof. Let $w \in \mathrm{Pal}(n)$. We get $\overline{p}^R_w(n) = p^R_w(1) = |w|_1 = s_w(n)$. Now let $i \in [n]_0$. Assume that $s_w(n-i+1) = \overline{p}^R_w(n-i+1)$ holds. We have, by induction,

$$\begin{aligned}
\overline{p}^R_w(n-i) &= \overline{p}_w(n-(n-i)+1) = \overline{p}_w(i+1) \\
&= \overline{p}_w(i) - (p_w(i) - p_w(i-1)) \\
&= \overline{p}^R_w(n-i+1) + (-s_w(i) + s_w(i-1)) \\
&= s_w(n-i+1) - w[i] \\
&= s_w(n-i) + w[n-i+1] - w[i] \\
&= s_w(n-i).
\end{aligned}$$

This proves the claim. □

By Lemma 5.1.8, we get $p_w \equiv \overline{p}^R_w$ since $p_w \equiv s_w$ for a palindrome w. As advocated earlier, our main theorem of this part (Theorem 5.1.9) gives a characterisation of prefix normal palindromes. The theorem allows us to decide if a word is a prefix normal palindrome by only looking at the maximum-ones-function, thus, a comparison of all factors is not required.

5.1.9 Theorem*. *Let $w \in \Sigma^n \backslash \{0^n\}$. Then w is a prefix normal palindrome if and only if $f_w = \overline{f}^{\mathsf{R}}_w$.*

Proof. Let $w \in \Sigma^n$. By definition of $\mathrm{NPal}(n)$, w is prefix normal and a palindrome, i.e., $s_w = f_w$. By Lemma 5.1.8 and Definition 5.1.6, we get $f_w = s_w = \overline{p}^R_w = \overline{f}^R_w$. This proves ⇒. Let $w \in \Sigma^n \backslash \{0^n\}$ such that $f_w = \overline{f}^{\mathsf{R}}_w$. If $w = \varepsilon$, then obviously $w \in \mathrm{NPal}(n)$ holds. Otherwise, if $w \neq \varepsilon$, there exists a least representative $v \in \Sigma^n$ with $w \in [v]_\equiv$. This implies $f_v = f_w = \overline{f}^R_w = \overline{f}^R_v$, therefore the assumption also holds for v. First, we will prove that v is a palindrome. Let $x \in \{0, 1\}$ and $i \in [n]$. Thus, we have $f_v(i-1) + x = f_v(i)$. Since v is least representative this implies $v[i] = x$. By the assumption, we get $\overline{f}^R_v(i-1) + x = \overline{f}^R_v(i)$ and applying the definition of the reversal and

$\overline{f}_v$ leads to

$$\begin{aligned}\overline{f}_v(n-i+1) = \overline{f}_v^R(i) &= \overline{f}_v^R(i-1) + x \\ &= \overline{f}_v(n-i+2) + x \\ &= \overline{f}_v(n-i+1) - (f_v(n-i+1) - f_v(n-i)) + x.\end{aligned}$$

Hence, we get $f_v(n-i+1) = f_v(n-i) + x$, i.e., $v[n-i+1] = x$. Thus, $v[n-i+1] = v[i]$ and therefore v is a palindrome. As proven in [14], prefix normal (and, thus, suffix normal) palindromes are not prefix normal-equivalent to any different word. Consequently, $v = w$ and $w \in \mathrm{NPal}(n)$. □

Table 5.1 presents the amount of prefix normal palindromes up to length 30. These results support the conjecture in [14] that there is a different behaviour for even and odd length of the word.

Table 5.1. Number of prefix normal palindromes. [61] (A308465)

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
#	2	2	3	3	5	4	8	7	12	11	21	18	36	31	57	55	104	91	182	166	308	292	562	512	1009	928	1755	1697	3247	2972

5.2 Recursive Construction of Prefix Normal Equivalence Classes

In this section we investigate how to generate least representatives of length $n+1$ using the least representatives of length n. Our approach is similar to the work of [44] except that the authors of [44] investigated appending a letter to prefix normal words while we explore the behaviour on prepending letters to least representatives. Consider the words $v = 1001$ and $w = 0011$, both being (different) least representatives of length 4. Prepending a 1 to them leads to 11001 and 10011 which are prefix-normal equivalent. We say that v and w *collapse* and denote it by $v \leftrightarrow w$. Hence, for determining the index of $\equiv_n$ based on the least representatives of length $n-1$, only the least representative of one class matters.

5.2.1 Definition. Two words $w, v \in \Sigma^n$ *collapse* if $1w \equiv_{n+1} 1v$ holds. This is denoted by $w \leftrightarrow v$.

Prepending a 1 to a non least representative will never lead to a least representative. Therefore, it is sufficient to only look at least representatives. Since collapsing is an equivalence relation, denote the equivalence class w.r.t. $\leftrightarrow$ of a word $w \in \Sigma^*$ by $[w]_{\leftrightarrow}$. Next, we present some general results regarding the connections between the least representatives of lengths n and $n+1$. As mentioned in Remark 5.1.3, 0^n and 1^n are for all $n \in \mathbb{N}$ least representatives. This implies that they do not have to be considered in the recursive process.

5.2.2 Remark. By [44], a word $w0 \in \Sigma^{n+1}$ is prefix-normal if w is prefix-normal. Consequently, we know that if a word $w \in \Sigma^n$ is suffix normal, $0w$ is suffix normal as well. This leads in accordance to the naïve upper bound of $2^n + 1$ to a naïve lower bound of $|\Sigma^n / \equiv_n|$ for $|\Sigma^{n+1} / \equiv_{n+1}|$.

5.2.3 Remark. The maximum-ones functions for $w \in \Sigma^*$ and $0w$ are equal on all $i \in [|w|]$ and $f_{0w}(|w|+1) = f_w(|w|)$ since the factor determining the maximal number of 1's is independent of the leading 0. Prepending 1 to a word w may result in a difference between f_w and f_{1w}, but notice that since only one 1 is prepended, we always have $f_{1w}(i) \in \{f_w(i), f_w(i)+1\}$ for all $i \in [n]$. In both cases we have $s_w(i) = s_{\mathsf{x}w}(i)$ for $\mathsf{x} \in \{0,1\}$ and $i \in [|w|]$ and $s_{0w}(n+1) = s_w(n)$ as well as $s_{1w}(n+1) = s_w(n)+1$.

First, we improve the naïve upper bound to $2|\Sigma^n/\equiv_n|$ by proving that only least representatives in Σ^n can become least representatives in Σ^{n+1} by prepending 1 or 0.

5.2.4 Proposition. *Let $w \in \Sigma^n$ not be least representative. Neither $0w$ nor $1w$ is a least representative in Σ^{n+1}.*

Proof. Suppose $0w$ is a least representative, i.e., $f_{0w}(i) = s_{0w}(i)$ for all $i \in [n+1]$. By $s_{0w}(i) = s_w(i)$ and $f_{0w}(i) = f_w(i)$ for $i \in [n]$, we have $s_w(i) = f_w(i)$ and, thus, w would be a least representative. Now, suppose that $1w$ is a least representative. Since w is not a least representative there exists a $j \in [|w|]$ with $s_w(j) \neq f_w(j)$. Choose j minimal. Since $1w$ is a least representative, we get

$$f_{1w}(j) = s_{1w}(j) = s_w(j) \neq f_w(j).$$

By Remark 5.2.3, we have $f_{1w}(j) = f_w(j) + 1$ $s_w(j) \leqslant f_w(j)$ implies $f_w(j) \geqslant s_w(j) = f_{1w}(j) = f_w(j) + 1$ - a contradiction. □

By Proposition 5.1.2, prefix (and, thus, suffix) normal palindromes form a singleton class. This implies immediately that a word $w \in \Sigma^n$ such that $1w$ is a prefix normal palindrome, does not collapse with any other $v \in \Sigma^n \backslash \{w\}$. The next lemma shows that even prepending once a 1 and once a 0 to different words leads only to equivalent words in one particular case.

5.2.5 Lemma. *Let $w, v \in \Sigma^n$ be different least representatives. Then $0w \equiv_n 1v$ if and only if $v = 0^n$ and $w = 0^{n-1}1$.*

Proof. The equivalence of $00^{n-1}1 = 0^n1$ and 10^n is immediate. This proves the $\Leftarrow$-direction. For the other direction assume $0w \equiv_n 1v$. By definition, we get $f_{0w}(i) = f_{1v}(i)$ for all $i \in [n+1]$ and, moreover, by Remark 5.2.3, $f_w(i) = f_{1v}(i)$ for all $i \in [n]$. By $s_w(1) = f_w(1) = f_{1v}(1) = 1$, we get $w[|w|] = 1$ and, by $s_w(n) = f_{1v}(n) = |w|_1$, there exists $u \in \mathrm{Fact}_n(1v)$ with $|u|_1 = |w|_1$. The equivalence of $0w$ and $1v$ implies $|w|_1 = |v|_1 + 1$ and, thus, u has to be a prefix of $1v$. Hence, $u[2..n]$ is a prefix of v of length $n-1$ with $|w|_1 - 1$ 1s. Since this is the overall amount of 1 in v, $v[n] = 0$ follows. Lemma 5.1.3 implies immediately $v = 0^n$. By $s_w(1) = f_{1v}(1) = 1$, $w[n] = 1$ follows and the claim follows with $|0w|_1 = |1v|_1$. □

By Lemma 5.2.5 and Remark 5.2.2, it suffices to investigate the collapsing relation on prepending 1s. The following proposition characterises the least representative $1w$ among the elements $1v \in [1w]_{\equiv}$ for all least representatives $v \in \Sigma^n$ with $w \leftrightarrow v$ for $w \in \Sigma^n$.

5.2.6 Proposition. *Let $w \in \Sigma^n$ be a least representative. Then $1w \in \Sigma^{n+1}$ is a least representative if and only if $f_{1w}(i) = f_w(i)$ holds for $i \in [n]$ and $f_{1w}(n+1) = f_w(n) + 1$.*

Proof. Let $w \in \Sigma^n$ be a least representative. First, consider $1w \in \Sigma^{n+1}$ to be a least representative as well. Since w is a least representative we have $s_w(i) = f_w(i)$ for all $i \in [n]$. By Remark 5.2.3, $s_w(i) = s_{1w}(i)$ follows for all $i \in [n]$ and with $1w$ being a least representative we get $f_w(i) = f_{1w}(i)$ for all $i \in [n]$. By the same arguments, we get $f_{1w}(n+1) = s_{1w}(n+1) = s_w(n) + 1 = f_w(n) + 1$. Similarly we get for the second direction $f_{1w}(i) = f_w(i) = s_w(i) = s_{1w}(i)$ for all $i \in [n]$ and $f_{1w}(n+1) = f_w(n) + 1 = s_w(n) + 1 = s_{1w}(n+1)$. □

5.2.7 Corollary. *Let $w \in \mathrm{NPal}(n)$. Then $f_{w1}(i) = f_w(i)$ for $i \in [n]$ and $f_{w1}(n+1) = f_w(n)+1$. Moreover, $s_{w1}(i) = s_w(i)$ for $i \in [n]$ and $s_{w1}(n+1) = s_w(n) + 1$.*

Proof. Since w is a prefix normal palindrome, we have $(1w)^R = w^R 1 = w1$. This implies $f_w(i) = f_{1w}(i) = f_{(1w)^R}(i) = f_{w1}(i)$ for all $i \in [n]$ and $f_{w1}(n+1) = f_w(n) + 1$. If $s_{w1}(i) = s_w(i) + 1$ then $s_{w1}(i) = s_w(i) + 1 = f_w(i) + 1 = f_{w1}(i) + 1$ would contradict $s_{w1}(i) \leqslant f_{w1}(i)$ for some $i \in [n+1]$. This proves the claim for the suffix-ones function. □

This characterisation is unfortunately not convenient for determining either the number of least representatives of length $n+1$ from the ones from length n or the collapsing least representatives of length n. For a given word w, the maximum-ones function f_w has to be determined, f_w to be extended by $f_w(n) + 1$, and finally the associated word - under the assumption $f_{1w} \equiv s_{1w}$ - has to be checked for being suffix normal. For instance, given $w = 10101$ leads to $f_w = 11223$, and is extended to $f_{1w} = 112234$. This would correspond to 110101 which is not suffix normal and, thus, w is not extendable to a new least representative. The following two lemmata reduce the number of least representatives that need to be checked for extensibility.

5.2.8 Lemma. *Let $w \in \Sigma^n$ be a least representative such that $1w$ is a least representative as well. Then for all least representatives $v \in \Sigma^n \backslash \{w\}$ collapsing with w, $f_v(i) \leqslant f_w(i)$ holds for all $i \in [n]$, i.e., all other least representatives have a smaller maximal-one sum.*

Proof. Let $v \in \Sigma^n \backslash \{w\}$ a least representative with $w \leftrightarrow_n v$. By the property of being least representative, the definition of the maximum-ones and suffix-ones functions follows for all $i \in [n]$

$$f_v(i) = s_v(i) = s_{1v}(i) \leqslant f_{1v}(i) = f_{1w}(i) = s_{1w}(i) = s_w(i) = f_w(i).$$

By $v \neq w$, there exists at least one $j \in [n]$ with $s_w(j) > s_v(j)$. This implies

$$\begin{aligned}\sigma(v) &= \sum_{i \in [n]} f_v(i) = \sum_{i \in [n]} s_v(i)\\ &< s_w(j) + \sum_{i \in [n] \backslash \{j\}} s_w(j) = \sum_{i \in [n]} s_w(j) = \sum_{i \in [n]} f_w(j)\\ &= \sigma(w).\end{aligned}$$

□

5.2.9 Corollary. *If $w, v \in \Sigma^n$ and $1w \in \Sigma^{n+1}$ are least representatives with $w \leftrightarrow v$ and $v \neq w$ then $w \leqslant v$.*

Proof. By $v \neq w$, there exists an $i \in [n]$ minimal with $w[i] \neq v[i]$. Suppose $w[i] = 1$ and $v[i] = 0$. By $f_{1v} \equiv f_{1w}$, we get $|w[i+1..n]|_1 + 1 = |v[i+1..n]|_1$. Thus,

$$\begin{aligned}s_{1v}(n-i) &= s_v(n-i) = s_w(n-i) + 1 = f_w(n-i) + 1 = f_{1w}(n-i) + 1\\ &= f_{1v}(n-i) + 1.\end{aligned}$$

This contradicts $s_{1v}(n-i) \leqslant f_{1v}(n-i)$. □

5.2.10 Remark. By Corollary 5.2.9, the lexicographically smallest collapsing least representative w leads to the least representative of $[1w]$. Thus, if w is a least representative not collapsing with any lexicographically smaller word then $1w$ is least representative.

Before we present the theorem characterising exactly the collapsing words for a given word w, we introduce a symmetry property of the least representatives which are not extendable to least representatives, i.e., a property of words which collapse.

5.2.11 Lemma. *Let $w \in \Sigma^n$ be a least representative. Then $f_{1w}(i) \neq f_w(i)$ for some $i \in [n]$ iff $f_{1w}(n-i+1) \neq f_w(n-i+1)$.*

Proof. Since w is least representative, we have

$$f_w(n-i+1) = s_w(n-i+1) = |\operatorname{Suff}_{n-i+1}(w)|_1 = |w|_1 - |\operatorname{Pref}_{i-1}(w)|_1.$$

From $f_w(i) \neq f_{1w}(i)$, it follows $f_{1w}(i) = f_w(i) + 1 = s_w(i) + 1$. Thus, $1w$ has a factor of length i with $s_w(i) + 1$ 1s. The suffix normality of w implies that this factor needs to be the prefix of $1w$ of length i, i.e., $|\operatorname{Pref}_{i-1}(w)|_1 = s_w(i)$. Thus, we get $f_w(n-i+1) = |w|_1 - s_w(i)$. On the other hand we have

$$\begin{aligned} |\operatorname{Pref}_{n-i+1}(1w)|_1 &= |\operatorname{Pref}_{n-i}(w)|_1 + 1 = |w|_1 - |\operatorname{Suff}_i(w)|_1 + 1 \\ &= |w|_1 - s_w(i) + 1. \end{aligned}$$

Consequently, $f_{1w}(n-i+1) \geqslant |w|_1 - s_w(i) + 1 > f_w(n-i+1)$. The second direction follows immediately with $j := n-i+1$ and $f_{1w}(j) \neq f_w(j)$.

□

By [14, Lemma 10], a word $w1$ is prefix normal if and only if $|\operatorname{Suff}_k(w)|_1 < |\operatorname{Pref}_{k+1}(w)|_1$ for all $k \in \mathbb{N}$. The following theorem extends this result for determining the collapsing words w' for a given word w.

5.2.12 Theorem. *Let $w \in \Sigma^n$ be a least representative and $w' \in \Sigma^n \backslash \{w\}$ with $|w|_1 = |w'|_1 = s \in \mathbb{N}$. Moreover, let $v \not\leftrightarrow w$ for all $v \in \Sigma^*$ with $v \leqslant w$. Then $w \leftrightarrow w'$ iff*

1. $f_{w'}(i) \in \{f_w(i), f_w(i) - 1\}$ *for all* $i \in [n]$,

2. $f_{w'}(i) = f_w(i)$ *implies* $f_{1w'}(i) = f_w(i)$,

3. $$f_{w'}(i) \geqslant \begin{cases} f_{w'}(n) - f_{w'}(n-i+1) & \text{if } f_{w'}(n-i+1) = f_{w'}(n-i), \\ f_{w'}(n) - f_{w'}(n-i+1) + 1 & \text{otherwise.} \end{cases}$$

Proof. Notice that $|w|_1 = |w'|_1 = s \in \mathbb{N}$ implies immediately $f_w(1) = f_{w'}(1) = 1$ and $f_{w'}(n) = f_w(n) = s$. Moreover, for all $i \in [n]$ we have $f_{w'}(i) \in \{f_{w'}(i-1), f_{w'}(i-1)+1\}$ and $f_w(i) \in \{f_w(i-1), f_w(i-1)+1\}$ and, by Lemma 5.2.11, we get $f_{w'}(i) \neq f_{1w'}(i)$ iff $f_{w'}(n-i+1) \neq f_{1w'}(n-i+1)$.

First, consider the ⇐-direction, i.e., let $w' \in \Sigma^n$ with $|w|_1 = s$ and the properties 1, 2, and 3. We have to prove $w' \leftrightarrow w$, hence, we have to prove $f_{1w}(i) = f_{1w'}(i)$ for all $i \in [n]$. Since w does not collapse with any lexicographically smaller $v \in \Sigma^n$, $0w$ is a least representative by Remark 5.2.10. From Proposition 5.2.6, it follows $f_w(i) = f_{1w}(i)$ for all $i \in [n]$. Obviously we have $f_{1w}(1) = 1 = f_{1w'}(1)$ and, hence, the claim holds for $i = 1$. By $f_{w'}(n) = s$,we get $f_{1w'}(n) \in \{s, s+1\}$. If $f_{1w'}(n)$ were $s+1$ then, by $f_{1w'}(n) \neq f_{w'}(n)$ and, consequently, by Lemma 5.2.11, we would have $1 = f_{1w'}(1) \neq f_{w'}(1) = 1$. Hence, the claim holds for $i = n$. Let $i \in [n-1]_{>1}$. The claim holds by Property 1 and Proposition 5.2.6 if $f_{w'}(i) = f_w(i)$. Hence, assume $f_{w'}(i) \neq f_w(i)$ for an $i \in [n-1]_{>1}$, i.e., $f_{w'}(i) = f_w(i) - 1$ by Property 1. By Remark 5.2.3 we have $f_{1w'}(i) \in \{f_{w'}(i), f_{w'}(i)+1\}$.
case 1: $f_{w'}(i) = f_{1w'}(i)$
If w''s prefix of length $i-1$ had more (or equal) 1s than the suffix of length i, then the prefix of $1w'$ of length i would have strictly more 1s than the suffix of length i. This contradicts $f_{w'}(i) = f_{1w'}(i)$ and, thus, we have $|\operatorname{Pref}_{i-1}(w')|_1 < |\operatorname{Suff}_i(w')|_1$. By $f_{w'}(n-i+1) \geqslant |\operatorname{Suff}_{n-i+1}(w')|_1$ and $|\operatorname{Suff}_{n-i+1}(w')|_1 + |\operatorname{Pref}_{i-1}(w')|_1 = s$, we get

$$\begin{aligned} s - f_{w'}(n-i+1) &\leqslant s - |\operatorname{Suff}_{n-i+1}(w')|_1 = |\operatorname{Pref}_{i-1}(w')|_1 \\ &< |\operatorname{Suff}_i(w')|_1 \leqslant f_{w'}(i). \end{aligned}$$

This is a contradiction to property 3.
case 2: $f_{w'}(i) + 1 = f_{1w'}(i)$
In this case we get immediately

$$f_{1w'}(i) = f_{w'}(i) + 1 = f_w(i) - 1 + 1 = f_w(i) = f_{1w}(i).$$

Thus, $f_{1w'}(i) = f_w(i)$ for all $i \in [n]$ which means that $f_{1w'}$ and f_{1w} are identical, i.e., w and w' collapse.

For the ⇒-direction, assume $w \leftrightarrow w'$, i.e., $f_{1w'} = f_{1w}$. Proposition 5.2.4

implies that w' can be assumed as a least representative since $1w$ is a least representative and by w and w' collapsing, $1w'$ is one as well. By Proposition 5.2.6, we have $f_{1w}(i) = f_w(i)$ for all $i \in [n]$ and, thus, $f_{1w'}(i) = f_w(i)$ which proves (2). Since $f_w(i) = f_{1w'}(i) \in \{f_{w'}(i), f_{w'}(i) + 1\}$ for all $i \in [n]$ we get property 1. Since w' is a least representative, Lemma 5.1.4 implies property 3. □

Theorem 5.2.12 allows us to construct the equivalence classes w.r.t. the least representatives of the previous length but more tests than necessary have to be performed: Consider, for instance $w = 11101100111011111$ of length 17 which is the shortest least representative of length 17 not collapsing with any lexicographically smaller least representative. For w we have $f_w = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 8 \cdot 8 \cdot 9 \cdot 10 \cdot 10 \cdot 11 \cdot 12 \cdot 13$ where the dots just act as separators between letters. Thus, we know for any w' collapsing with w, that $f_{w'}(1) = 1$ and $f_{w'}(17) = 13$. The constraints $f_{w'}(2) \in \{f_{w'}(2), f_{w'}(2) + 1\}$ and $f_{w'}(2) \leqslant f_w(2)$ implies $f_{w'}(2) \in \{1, 2\}$. First the check that $f_{w'}(10) = 4$ is impossible excludes $f_{w'}(2) = 1$. Since no collapsing word can have a factor of length 2 with only one 1, a *band* in which the possible values range can be defined by the unique greatest collapsing word w'. It is not surprising that this word is connected with the prefix normal form. The following two lemmata define the band in which the possible collapsing words f_w are.

5.2.13 Lemma. *Let $w \in \Sigma^n \backslash \{0^n\}$ be a least representative with $v \not\leftrightarrow w$ for all $v \in \Sigma^n$ with $v \leqslant w$. Set $u := (1w[1..n-1])^R$. Then $w \leftrightarrow u$ and for all least representatives $v \in \Sigma^n \backslash \{u\}$ with $v \leftrightarrow w$ and all $i \in [n]$ $f_v(i) \geqslant f_u(i)$, thus, $\sigma(u) = \sum_{i \in [n]} f_u(i) \leqslant \sum_{i \in [n]} f_v(i) = \sigma(v)$.*

Proof. Set $u = (1w[1..n-1])^R$. Then by w odd, it follows

$$\begin{aligned} f_{1w} &= f_{(1w)^R} = f_{w^R 1} = f_{w[n](w[1..n-1])^R 1} = f_{w[n](1w[1..n-1])^R} = f_{1(1w[1..n-1])^R} \\ &= f_{1u}, \end{aligned}$$

i.e., $w \leftrightarrow u$. Since w does not collapse with any lexicographically smaller word, $1w$ is a least representative by Remark 5.2.10. By Remark 2.4.3, $(1w)^R \in [1w]_\equiv$ and $w^R 1$ is lexicographically the largest element in the class. If there existed a $v \in [w]_\leftrightarrow$ with $v > w[n-1..1]1$, then

$$1v > 1w[n-1..1] = w^R 1 = (1w)^R$$

would hold which contradicts the maximality of $(1w)^R$. □

Notice that $w' = (1w[1..n-1])^R$ is not necessarily a least representative in $\Sigma^n / \equiv_n$ witnessed by the word from the last example. For w we get $u = 1110111001101111$ with $f_u(8) = f_w(8)$ and $f_u(10) = 7 \neq 8 = f_w(10)$ violating the symmetry property given in Lemma 5.2.11. The following lemma alters w' into a least representative which represents still the lower limit of the band.

5.2.14 Lemma. *Let $w \in \Sigma^n$ be a least representative such that $1w$ is also a least representative. Let $w' \in \Sigma^n$ with $w \leftrightarrow w'$, and I the set of all $i \in [\lfloor \frac{n}{2} \rfloor]$ with*

$$(f_{w'}(i) = f_w(i) \wedge f_{w'}(n-i+1) \neq f_w(n-i+1)) \text{ or}$$
$$(f_{w'}(i) \neq f_w(i) \wedge f_{w'}(n-i+1) = f_w(n-i+1))$$

and $f_w(j) = f_{w'}(j)$ for all $j \in [n]\backslash I$. Then $\widehat{w}$ defined such that $f_{\widehat{w}}(j) = f_{w'}(j)$ for all $j \in [n]\backslash I$ and $f_{\widehat{w}}(n-i+1) = f_{w'}(n-i+1)+1$ ($f_{\widehat{w}}(i) = f_{\widehat{w}}(i)+1$ resp.) for all $i \in I$ holds, collapses with w.

Proof. Let $k \in [n]$. Since $1w$ is least representative, we have $f_{1w}(k) \geqslant f_{1\widehat{w}}(k)$. If $k \notin I$ we get

$$f_{1w}(k) = f_w(k) = f_{w'}(k) = f_{\widehat{w}}(k) \leqslant f_{1\widehat{w}}(k)$$

and, thus, $f_{1w}(k) = f_{1\widehat{w}}(k)$. If $k \in I$ we get in the first case

$$\begin{aligned} f_{1w}(n-k+1) = f_w(n-k+1) = f_{w'}(n-k+1)+1 = f_{\widehat{w}}(n-k+1) \\ \leqslant f_{1\widehat{w}}(n-k+1) \end{aligned}$$

and, thus, $f_{1w}(n-k+1) = f_{1\widehat{w}}(n-k+1)$. The second case holds analogously. □

5.2.15 Remark. Lemma 5.2.14 applied to $(1w[1..n-1])^R$ gives the lower limit of the band. Let $\widehat{w}$ denote the output of this application for a given $w \in \Sigma^n$ according to Lemma 5.2.14.

Continuing with the example, we first determine $\widehat{w}$ for

$$w = 11110111001101111.$$

We get $u = w[n-1..1]1$. Since for all collapsing $w' \in \Sigma^n$ we have $f_{\widehat{w}}(i) \leqslant f_{w'}(i) \leqslant f_w(i)$, w' is determined for $i \in [17]\backslash\{5, 9, 13\}$. Since the value for 5 determines the one for 13 there are only two possibilities, namely $f_{w'}(5) = 5$ and $f_{w'}(9) = 7$ and $f_{w'}(5) = 4$ and $f_{w'}(9) = 8$. Notice that the words w' corresponding to the generated words $f_{w'}$ are not necessarily least representatives of the shorter length as witnessed by the one with $f_{w'}(5) = 5$ and $f_{w'}(9) = 7$. In this example this leads to at most three words being not only in the class but also in the list of former representatives. Thus, we are able to produce an upper bound for the cardinality of the class. Notice that in any case we only have to test the first half of w''s positions by Lemma 5.2.11. This leads to the following definition.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
f_w	1	2	3	4	5	5	6	7	8	8	8	9	10	10	11	12	13
f_u	1	2	3	4	4	5	6	7	7	7	8	9	9	10	11	12	13
$f_{\hat{w}}$	1	2	3	4	4	5	6	7	7	8	8	9	9	10	11	12	13

Table 5.2. f for $w = 11110111001101111$.

5.2.16 Definition. Let $h_d : \Sigma^* \times \Sigma^* \to \mathbb{N}_0$ be the Hamming-distance. The *palindromic distance* $p_d : \Sigma^* \to \mathbb{N}_0$ is defined by

$$p_d(w) = h_d(w[1..\lfloor \frac{n}{2} \rfloor], (w[\lceil \frac{n}{2} \rceil + 1..|w|])^R).$$

Define the *palindromic prefix length* $p_\ell : \Sigma^* \to \mathbb{N}_0$ by

$$p_\ell(w) = \max \{ k \in [|w|] \mid \exists u \in \mathrm{Pref}_k(w) : \; p_d(u) = 0 \}.$$

The palindromic distance gives the minimal number of positions in which a bit has to be flipped for obtaining a palindrome. Thus, $p_d(w) = 0$ for all palindromes w, and, for instance, $p_d(110011001) = 2$ since the first half of w and the reverse of the second half mismatch in two positions. The palindromic prefix length determines the length of w's longest prefix being a palindrome. For instance $p_\ell(1101) = 2$ and $p_\ell(01101) = 4$. Since a least representative w determines the upper limit of the band and $w[n-1..1]1$ the lower limit, the palindromic distance of $ww[n-1..1]1$ is in relation to the positions of f_w in which collapsing words may differ from w.

5.2.17 Theorem. *If $w \in \Sigma^n$ and $1w$ are both least representatives then* $|[w]_\leftrightarrow| \leqslant 2^{\lceil \frac{p_d(ww[n-1..1]1)}{2} \rceil}$.

Proof. By Lemma 5.2.13, $w[n-1..1]1$ determines the lower bound of the band for collapsing words. Let $s_1, \dots, s_\ell \in [n]$ with $s_i < s_{i+1}$ for $i \in [\ell], \ell \in [n]$ be the positions with $w[s_i] \neq (w[n-1..1]1)[s_i]$. Thus, for all odd $i \in [\ell - 1]$, f_w and $f_{w[n-1..1]1}$ are different between s_i and $s_{i+1} - 1$, since a different bit leads to a different number of 1s. By the same argument, f_w and $f_{w[n-1..1]1}$ are identical between s_i and $s_{i+1} - 1$ for all even $i \in [\ell - 1]$. This implies that only the differences in odd positions lead to different values of the corresponding maximum-ones function. Since each difference in the maximum-ones functions can be altered independently for obtaining a potential collapsing word, the number of collapsing words is exponential in half the palindromic distance. □

For an algorithmic approach to determine the least representatives of length n, we want to point out that the search for collapsing words can also be reduced using the palindromic prefix length. Let $w_1, \dots, w_m$ be the least representatives of length $n-1$. For each w we keep track of $|w| - p_\ell(w)$. For each w_i we check first if $|w_i| - p_\ell(w_i) = 1$ since in this case the prepended 1 leads to a palindrome. Only if this is not the case, $[w_i]_\leftrightarrow$ needs to be determined. All collapsing words computed within the band of w_i and $\hat{w}_i$ are deleted in $\{w_{i+1}, \dots, w_m\}$.

In the remaining part of the section we investigate the set $\mathrm{NPal}(n)$ w.r.t. $\mathrm{NPal}(\ell)$ for $\ell < n$. This leads to a second formula for an upper bound and a refinement for determining the least representatives of $\Sigma^n / \equiv_n$ faster.

5.2.18 Lemma. *If $w \in \mathrm{NPal}(n) \backslash \{1^n\}$ then $1w$ is not a least representative but $w1$ is a least representative.*

Proof. It suffices to prove that $w1$ is a least representative. Then $(w1)^R = 1w$ is prefix normal and since $w1$ is not a palindrome, $1w$ is not a least representative. By Corollary 5.2.7, follows immediately that $w1$ is a least representative. □

5.2.19 Remark. By Lemma 5.2.18, it follows that all words $w \in \mathrm{NPal}(n)$ collapse with a smaller least representative. Thus, for all $n \in \mathbb{N}$, an upper bound for $|\Sigma^{n+1} / \equiv_{n+1}|$ is given by $2|\Sigma^n / \equiv_n| - \mathrm{npal}(n)$.

For a closed recursive calculation of the upper bound in Remark 5.2.19, the exact number $\mathrm{npal}(n)$ is needed. Unfortunately we are not able to determine $\mathrm{npal}(n)$ for arbitrary $n \in \mathbb{N}$. The following results show relations between prefix normal palindromes of different lengths. For instance, if $w \in \mathrm{NPal}(n)$ then $1w1$ is a prefix normal palindrome as well. The importance of the the prefix normal palindromes is witnessed by the following estimation.

5.2.20 Theorem. *For all $n \in \mathbb{N}_{\geqslant 2}$ and $\ell = |\Sigma^n / \equiv_n|$ we have*

$$\ell + \mathrm{npal}(n-1) \leqslant |\Sigma^{n+1} / \equiv_{n+1}| \leqslant \ell + \mathrm{npal}(n+1) + \frac{\ell - \mathrm{npal}(n+1)}{2}.$$

Proof. The lower bound follows by the fact that $1w1$ is a prefix normal palindrome if w is one. For the upper bound all $0w$, for w being a least representative in $\Sigma^n / \equiv_n$, have to be counted and then all prefix normal palindromes. All other elements collapse with at least one different element. □

The following results only consider prefix normal palindromes that are different from 0^n and 1^n. Notice for these special palindromes that 0^n0^n, 1^n1^n, 1^n11^n, 0^n00^n, 11^n1^n1, $10^n0^n1 \in \mathrm{NPal}(k)$ for an appropriate $k \in \mathbb{N}$ but $0^n10^n \notin \mathrm{NPal}(2n+1)$.

5.2.21 Lemma. *If $w \in \mathrm{NPal}(n) \backslash \{1^n, 0^n\}$ then neither ww nor $w1w$ are prefix normal palindromes.*

Proof. Let $k \in [n-1]$ be minimal with $f_w(k) = f_w(k+1)$ (exists by $w \notin \{1^n, 0^n\}$). Thus, we have $f_w(k+1) = k$. Since $w \in \mathrm{NPal}(n)$ we have $|\mathrm{Suff}_k(w)|_1 = |\mathrm{Pref}_k(w)|_1 = k$ and $|\mathrm{Suff}_{k+1}(w)|_1 = |\mathrm{Pref}_{k+1}(w)|_1 = k$. This implies $f_{ww}(k+1) = k+1$ and, hence, $f_{ww}(k+1) \neq s_{ww}(k+1)$. The proof of $w1w \notin \mathrm{NPal}(2n+1)$ is similar to the proof of Lemma 5.2.21: in the middle of $w1w$ is a larger block of 1's than at the end. □

5.2.22 Lemma. *Let $w \in \mathrm{NPal}(n) \backslash \{0^n\}$ with $n \in \mathbb{N}_{\geqslant 3}$. If $w0w$ is also a prefix normal palindrome then $w = 1^k$ or $w = 1^k01u101^k$ for some $u \in \Sigma^*$ and $k \in \mathbb{N}$.*

Proof. By $w \neq 0^n$, it follows $w = 1u1$ for an $u \in \mathrm{Pal}(n)$. If $w \neq 1^k$, there exists k minimal with $w[k] = 0$. Suppose $w = 1^k0^\ell u0^\ell 1^k$ for some $k \in \mathbb{N}$ and $\ell \in \mathbb{N}_{>1}$. Then $p_{w0w}(k+2) = k$ but $|w0w[n-k, n+2]|_1 = k+1$. This is a contradiction to $w0w \in \mathrm{NPal}(2n+1)$. This implies $w = 1^k01u101^k$. □

A characterisation for $w1w$ being a prefix normal palindrome is more complicated. By $w \in \mathrm{NPal}(n)$, it follows that a block of 1s contains at most the number of 1s of the previous block. But if such a block contains strictly less 1s the number of 0s in between can increase by the same amount the number of 1s decreased.

5.2.23 Lemma. *Let* $w \in \mathrm{NPal}(n) \backslash \{1^n, 0^n\}$*. If* $1ww1$ *is also a prefix normal palindrome then* $10 \in \mathrm{Pref}(w)$*.*

Proof. Let $1ww1 \in \mathrm{NPal}(2n+2)$. Since $w \neq 0^n$, there exists $u \in \Sigma^*$ with $w = 1u1$. Since $w \neq 1^n$ there exists a minimal $k \in \mathbb{N}$ with $u[k] = 0$. If $k > 1$, then $1ww1 = 1^k01^{2k}01^k$ or $1ww1 = 1^k0v01^{2k}0v01^k$. In both cases we have a contradiction to $1ww1 \in \mathrm{NPal}(2n+2)$. □

Lemma 5.2.21, 5.2.22, and 5.2.23 indicate that a characterization of prefix normal palindromes based on smaller ones is hard to determine.

Chapter 6

Conclusion

In this thesis we investigated three different subfields of the domain combinatorics on words, namely scattered factors, locality, and prefix normal words.

In Chapter 3 we looked into three different aspects of the domain of scattered factors: the cardinality of k-spectra of weakly c-balanced words, the scattered factor universality, and the reconstruction of words by the binomial coefficients of right-bounded block words.

The idea of the first part was to give insights about the index of Simon congruence by weakening it: instead of dealing with the index itself (which is a problem we were not able to solve), we grouped words by the cardinality of different spectra and investigated properties of the words having the same cardinality of a k-spectrum for a given k. In particular, we gave several insights into the structure of the set of all k-spectra of weakly-0-balanced words of length $2k$ by considering for which numbers n there exists w such that the k-spectrum of w has cardinality n. In particular, we characterised the first two gaps in the possibilities for each k which are regular (in the sense that the first and second gaps are always from $k+2$ to $2k-1$ and $2k+1$ to $3k-4$ (inclusive)). On the other hand, we saw that the third gap is considerably less regular and, thus, resists a natural characterisation. As mentioned, some of the weakly-0-balanced words are θ-palindromes. Since the θ-palindromes of length $2k$ are constructible from the ones of length $2(k-1)$ (except for each even k exactly one θ-palindrome) we surmised that the structure and properties propagate. Moreover, we expected that the knowledge of the word's second half helps in finding the cardinalities of the k-spectra. Nevertheless, we were only able to get results for θ-palindromes in the same manner as for the other words, but we still believe that the structure of the θ-palindromes can reveal more insights with further work.

In the second part of Chapter 3 we looked in more detail into the words having the maximal cardinality of k-spectra for given ks, namely words whose k-spectrum is Σ^k. Following the notion of language theory we called these words k-universal. We have proven how this universality behaves if a word is repeated and how this characterisation can be exploited to obtain linear-time algorithms for obtaining an discommon scattered factor. Moreover, we set the universality of a palindrome into relation with its first half (minus one letter if the length is odd) as well as the generalised repetition $w\pi(w)$ for a morphic permutation π. The last part of Section 3.2 dealt with circular universality. Here we have proven the relation between universality and circular universality and we have shown that the characterisation in Theorem 3.2.17 does not hold for arbitrary alphabets. We conjecture that for an alphabet of cardinality σ the notion of circularity has to be generalised such that, assuming the word as a circle, not once but $\sigma - 1$ times the word has to be read before the universality is increased. Finally, in the last section we developed data structures that allow us to determine the universality of factors of a given word.

The last part of Chapter 3 is dedicated to the reconstruction problem. While in before the approaches to reconstructing a word uniquely by scattered factors where restricted to scattered factors of the same length, we relaxed the constraint on the length but therefore took only specific scattered factors - the right-bounded block words. This relaxation of the so far investigated reconstruction problem from scattered factors from k-spectra to arbitrary sets yields that less scattered factors than the best known upper bound are sufficient to reconstruct a word uniquely. Not only in the binary but also in the general case the distribution of the letters plays an important role: in the binary case the amount of necessary binomial coefficients is smaller the larger $|w|_{\mathtt{a}} - |w|_{\mathtt{b}}$ is. The same observation results from the general case - if all letters are equally distributed in w then we need more binomial coefficients than in the case where some letters rarely occur and others occur much more often. Nevertheless, the restriction to right-bounded-block words (that are intrinsically Lyndon words) shows that a word can be reconstructed by fewer binomial coefficients if scattered factors from different spectra are taken. Further investigations may lead into two directions: a better characterisation of the uniqueness of the $k_{\mathsf{a}^\ell\mathsf{b}}$ would be helpful to understand better in which cases less than the worst case amount of binomial coefficients suffices and other sets than the right-bounded-block words could be investigated for the reconstruction problem.

In Chapter 4 we investigated the locality of patterns as well as the hardness of matching repetitions. In Section 4.1 we have proven that Loc is NP-complete. Notice that in [15] in addition to the reduction to Clique also reductions to Cutwidth and Pathwidth are given. In Section 4.2 we investigated the locality of repetitions and palindromes. We have shown that the locality increases linear in the number of repetitions which also holds for palindromes. Moreover, we have shown that even though Zimin words have a *nice* and *easy* structure, the locality grows exponentially with the word's length. Chapter 4 ends with a proof that the matching problem may turn out to be hard if patterns are repeated although the underlying pattern can be matched efficiently. We have proven this by using regular patterns and the repetition of regular patterns.

Finally, in Chapter 5 based on the work in [44], we investigated prefix normal palindromes and gave a characterisation based on the maximum-ones function. Moreover, results for a recursive approach to determine prefix normal palindromes are given. These results show that easy connections between prefix normal palindromes of different lengths cannot be expected. By introducing the collapsing relation we were able to partition the set of extension-critical words introduced in [44]. This leads to a characterization of collapsing words which can be extended to an algorithm determining the corresponding equivalence classes. Moreover, we have shown that palindromes and the collapsing classes are related.

The concrete values for prefix normal palindromes and the index of the collapsing relation remain an open problem as well as the cardinality of the equivalence classes w.r.t. the collapsing relation. Further investigations of the prefix normal palindromes and the collapsing classes lead directly to the index of the prefix equivalence.

Bibliography

[1] A. Amir and I. Nor. "Generalized function matching". In: *Journal of Discrete Algorithms* 5.3 (2007), pp. 514–523.

[2] D. Angluin. "Finding patterns common to a set of strings". In: *Journal of Computer and System Sciences* 21.1 (1980), pp. 46–62.

[3] P. Balister and S. Gerke. "The asymptotic number of prefix normal words". In: *Jour. Comb. Theo.* (2019).

[4] P. Barceló, L. Libkin, A.W. Lin, and P.T. Wood. "Expressive languages for path queries over graph-structured data". In: *ACM Transactions on Database Systems* 37.4 (2012), pp. 1–46.

[5] L. Barker, P. Fleischmann, K. Harwardt, F. Manea, and D. Nowotka. "Scattered factor-universality of words". In: *DLT - 24th International Conference, 2020, Tampa, FL, USA, May 11-15, 2020, Proceedings*. Ed. by N. Jonoska and D. Savchuk. Vol. 12086. LNCS. Springer, 2020, pp. 14–28.

[6] J. Berstel and J. Karhumäki. "Combinatorics on words: a tutorial". In: *Bulletin of the EATCS* 79 (2003), p. 178.

[7] V. Berthé, J. Karhumäki, D. Nowotka, and J. Shallit. "Mini workshop: combinatorics on words". In: *Oberwolfach Rep.* **7** (2010), pp. 2195–2244. DOI: 10.4171/OWR/2010/37.vv.

[8] K. Bringmann. "Fine-grained complexity theory (tutorial)". In: *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019, March 13-16, 2019, Berlin, Germany*. Ed. by R. Niedermeier and C. Paul. Vol. 126. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, 4:1–4:7.

[9] K. Bringmann and M. Künnemann. "Multivariate fine-grained complexity of longest common subsequence". In: *Proc. SODA 2018*. SIAM, 2018, pp. 1216–1235.

[10] N.G. de Bruijn. "A combinatorial problem". In: *Koninklijke Nederlandse Akademie v. Wetenschappen* 49 (1946), pp. 758–764.

[11] P. Burcsi, F. Cicalese, G. Fici, and Z. Lipták. "Algorithms for jumbled pattern matching in strings". In: *Int. Jour. Found. CS* 23.2 (2012), pp. 357–374.

[12] P. Burcsi, G. Fici, Z. Lipták, F. Ruskey, and J. Sawada. "Normal, abby normal, prefix normal". In: *Proc FUN*. Springer. 2014, pp. 74–88.

[13] P. Burcsi, G. Fici, Z. Lipták, F. Ruskey, and J. Sawada. "On combinatorial generation of prefix normal words". In: *Proc CPM*. Springer. 2014, p. 60.

[14] P. Burcsi, G. Fici, Z. Lipták, F. Ruskey, and J. Sawada. "On prefix normal words and prefix normal forms". In: *TCS* 659 (2017), pp. 1–13.

[15] K. Casel, J. D. Day, P. Fleischmann, T. Kociumaka, F. Manea, and M.L. Schmid. "Graph and string parameters: connections between pathwidth, cutwidth and the locality number". In: *ICALP*. Ed. by Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi. Vol. 132. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2019, 109:1–109:16. ISBN: 978-3-95977-109-2.

[16] J. Cassaigne, G. Richomme, K. Saari, and L.Q. Zamboni. "Avoiding abelian powers in binary words with bounded abelian complexity". In: *Int. Jour. Found. CS* 22.04 (2011), pp. 905–920.

[17] T.M. Chan and M. Lewenstein. "Clustered integer 3sum via additive combinatorics". In: *47th ACM symp. TOC*. ACM. 2015, pp. 31–40.

[18] H.Z.Q. Chen, S. Kitaev, T. Mütze, and B.Y. Sun. "On universal partial words". In: *Electronic Notes in Discrete Mathematics* 61 (2017).

[19] F. Cicalese, Z. Lipták, and M. Rossi. "Bubble-flip - a new generation algorithm for prefix normal words". In: *LATA*. Vol. 10792. LNCS. Springer, 2018, pp. 207–219.

[20] F. Cicalese, Z. Lipták, and M. Rossi. "On infinite prefix normal words". In: *Proc. SOFSEM*. 2019, pp. 122–135.

[21] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to algorithms, 3rd edition*. MIT Press, 2009.

[22] E.M. Coven and G.A. Hedlund. "Sequences with minimal block growth". In: *TCS* 7.2 (1973), pp. 138–153.

[23] M. Crochemore, C. Hancart, and T. Lecroq. *Algorithms on strings*. Cambridge University Press, 2007. ISBN: 978-0-521-84899-2.

[24] J. Currie and N. Rampersad. "Recurrent words with constant abelian complexity". In: *Adv. Appl. Math.* 47.1 (2011), pp. 116–124.

[25] J. Dassow. "Parikh mapping and iteration". In: *LNCS* 2235 (2001), pp. 85–101. ISSN: 0302-9743.

[26] J.D. Day, P. Fleischmann, F. Manea, and D. Nowotka. "K-spectra of weakly-c-balanced words". In: *Proc. DLT 2019*. Vol. 11647. LNCS. Springer, 2019, pp. 265–277.

[27] J.D. Day, P. Fleischmann, F. Manea, and D. Nowotka. "Local patterns". In: *Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2017*. Ed. by Satya V. Lokam and R. Ramanujam. Vol. 93. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017, 24:1–24:14.

[28] J.D. Day, P. Fleischmann, F. Manea, D. Nowotka, and M.L. Schmid. "On matching generalised repetitive patterns". In: *Developments in Language Theory, DLT 2018*. Ed. by Mizuho Hoshi and Shinnosuke Seki. Vol. 11088. LNCS. Springer, 2018, pp. 269–281.

[29] D. P. Dobkin and R. J. Lipton. "On the complexity of computations under varying sets of primitives". In: *J. Comput. Syst. Sci.* 18.1 (1979), pp. 86–91.

[30] A. W.M. Dress and P. Erdös. "Reconstructing words from subwords in linear time". In: *Annals of Combinatorics* 8.4 (2004), pp. 457–462.

[31] X. Droubay, J. Justin, and G. Pirillo. "Episturmian words and some constructions of de luca and rauzy". In: *Theor. Comput. Sci.* 255.1-2 (2001), pp. 539–553.

[32] M. Dudík and L.J. Schulman. "Reconstruction from subsequences". In: *J. Comb. Theory Series A* 103.2 (2003). Ed. by Hélène Barcelo, pp. 337–348.

[33] T. Ehlers, F. Manea, R. Mercas, and D. Nowotka. "K-abelian pattern matching". In: *J. Dis. Alg.* 34 (2015), pp. 37–48.

[34] C.H. Elzinga, S. Rahmann, and H. Wang. "Algorithms for subsequence combinatorics". In: *Theor. Comput. Sci.* 409.3 (2008), pp. 394–404.

[35] P.L. Erdős, P. Ligeti, P. Sziklai, and D.C. Torney. "Subwords in reverse-complement order". In: 10.4 (2006).

[36] T. Erlebach, P. Rossmanith, H. Stadtherr, A. Steger, and T. Zeugmann. "Learning one-variable pattern languages very efficiently on average, in parallel, and by asking queries". In: *Theoretical Computer Science* 261.1 (2001), pp. 119–156.

[37] S. Z. Fazekas, F. Manea, R. Mercas, and K. Shikishima-Tsuji. "The pseudopalindromic completion of regular languages". In: *Information and Compution* 239 (2014), pp. 222–236.

[38] W. Feller. *An introduction to probability theory and its applications*. Vol. 1. New York: Wiley, 1956.

[39] H. Fernau, F. Manea, R. Mercas, and M.L. Schmid. "Pattern matching with variables: fast algorithms and new hardness results". In: *Symposium on Theoretical Aspects of Computer Science, STACS 2015*. Ed. by Ernst W. Mayr and Nicolas Ollinger. Vol. 30. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015, pp. 302–315.

[40] H. Fernau, F. Manea, R. Mercaş, and M.L. Schmid. "Revisiting Shinohara's algorithm for computing descriptive patterns". In: *Theoretical Computer Science* 733 (2018), pp. 44–54. DOI: 10.1016/j.tcs.2018.04.035.

[41] H. Fernau and M.L. Schmid. "Pattern matching with variables: a multivariate complexity analysis". In: *Information and Computation* 242 (2015), pp. 287–305. DOI: 10.1016/j.ic.2015.03.006.

[42] H. Fernau, M.L. Schmid, and Y. Villanger. "On the parameterised complexity of string morphism problems". In: *Theory of Computing Systems* 59.1 (2016), pp. 24–51.

[43] M. Ferov. 2008.

[44] G. Fici and Z. Lipták. "On prefix normal words". In: *Proc DLT*. Vol. 6795. 2011.

[45] L. Fleischer and M. Kufleitner. "Testing Simon's congruence". In: *Proc. MFCS 2018*. Vol. 117. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 62:1–62:13.

[46] P. Fleischmann, M. Kulczynski, D. Nowotka, and D.B. Poulsen. "On collapsing prefix normal words". In: *LATA 2020, Milan, Italy, March 4-6, 2020, Proceedings*. Ed. by A. Leporati, C. Martín-Vide, D. Shapira, and C. Zandron. Vol. 12038. LNCS. Springer, 2020, pp. 412–424.

[47] P. Fleischmann, M. Lejeune, F. Manea, D. Nowotka, and M. Rigo. "Reconstructing words from right-bounded-block words". In: *DLT - 24th International Conference, 2020, Tampa, FL, USA, May 11-15, 2020, Proceedings*. Ed. by N. Jonoska and D. Savchuk. Vol. 12086. LNCS. Springer, 2020, pp. 96–109.

[48] D.D. Freydenberger. "Extended regular expressions: succinctness and decidability". In: *TCS* 53.2 (2013), pp. 159–193.

[49] D.D. Freydenberger, P. Gawrychowski, J. Karhumäki, F. Manea, and W. Rytter. "Testing k-binomial equivalence". In: *CoRR* abs/-1509.00622 (2015).

[50] D.D. Freydenberger and M.L. Schmid. "Deterministic regular expressions with back-references". In: *Journal of Computer and System Sciences* (2019).

[51] J.E.F. Friedl. *Mastering regular expressions*. 3rd. O'Reilly, 2006.

[52] Harold N. Gabow and Robert Endre Tarjan. "A linear-time algorithm for a special case of disjoint set union". In: *Proc. 15th STOC*. 1983, pp. 246–251.

[53] P. Gawrychowski, M. Lange, N. Rampersad, J.O. Shallit, and M. Szykula. "Existential length universality". In: LIPIcs 154 (2020). Ed. by Christophe Paul and Markus Bläser, 16:1–16:14.

[54] B. Goeckner, C. Groothuis, C. Hettle, B. Kell, P. Kirkpatrick, R. Kirsch, and R.W. Solava. "Universal partial words over non-binary alphabets". In: *Theor. Comput. Sci* 713 (2018), pp. 56–65.

[55] S. Halfon, P. Schnoebelen, and G. Zetzsche. “Decidability, complexity, and expressiveness of first-order logic over the subword ordering”. In: *Proc. LICS*. 2017, pp. 1–12.

[56] F. Harary. “On the reconstruction of a graph from a collection of subgraphs”. In: *In Theory of Graphs and its Applications (Proc. Sympos. Smolenice, 1963). Publ. House Czechoslovak Acad. Sci., Prague* (1964), pp. 47–52.

[57] J.-J. Hebrard. “An algorithm for distinguishing efficiently bitstrings by their subsequences”. In: *TCS* 82.1 (1991), pp. 35–49.

[58] M. Holzer and M. Kutrib. “Descriptional and computational complexity of finite automata - A survey”. In: *Inf. Comput.* 209.3 (2011), pp. 456–470.

[59] L. van Iersel and V. Moulton. “Leaf-reconstructibility of phylogenetic networks”. In: *SIAM J. Discrete Math* 32.3 (2018), pp. 2047–2066.

[60] H. Imai and T. Asano. “Dynamic segment intersection search with applications”. In: *Proc. 25th Annual Symposium on Foundations of Computer Science, FOCS*. IEEE Computer Society, 1984, pp. 393–402.

[61] OEIS Foundation Inc. *The on-line encyclopedia of integer sequencess*. 2020. URL: `http://oeis.org/`.

[62] L. I. Kalashnik. “The reconstruction of a word from fragments”. In: *Numerical Mathematics and Computer Technology* (1973), pp. 56–57.

[63] P. Karandikar, M. Kufleitner, and P. Schnoebelen. “On the index of Simon’s congruence for piecewise testability”. In: *Inf. Process. Lett.* 115.4 (2015), pp. 515–519.

[64] P. Karandikar and P. Schnoebelen. “The height of piecewise-testable languages and the complexity of the logic of subwords”. In: *Logical Methods in Computer Science* 15.2 (2019).

[65] P. Karandikar and P. Schnoebelen. “The height of piecewise-testable languages with applications in logical complexity”. In: *Proc. CSL 2016*. Vol. 62. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 37:1–37:22.

[66] J. Karhumäki. "Generalized Parikh mappings and homomorphisms". In: *Information and Control* 47.3 (Dec. 1980).

[67] J. Karhumäki, F. Mignosi, and W. Plandowski. "The expressibility of languages and relations by word equations". In: *Journal of the ACM* 47.3 (2000), pp. 483–505. DOI: 10.1145/337244.337255.

[68] L. Kari and K. Mahalingam. "Watson-crick palindromes in dna computing". In: *Natural Computing* 9.2 (2010), pp. 297–316.

[69] M. Kearns and L. Pitt. "A polynomial-time algorithm for learning k-variable pattern languages from examples". In: *Computational Learning Theory, COLT 1989*. Ed. by Ronald L. Rivest, David Haussler, and Manfred K. Warmuth. Morgan Kaufmann, 1989, pp. 57–71.

[70] P. J. Kelly. "A congruence theorem for trees". In: *Pacific J. Math.* 7.1 (1957), pp. 961–968.

[71] V. Keränen. "Abelian squares are avoidable on 4 letters". In: *Int. Coll. Aut., Lang., and Prog.* Springer. 1992, pp. 41–52.

[72] I. Krasikov and Y. Roditty. "On a reconstruction problem for sequences". In: *J. Comb. Theory, Ser. A* 77.2 (1997), pp. 344–348.

[73] M. Krötzsch, T. Masopust, and M. Thomazo. "Complexity of universality and related problems for partially ordered NFAs". In: *Inf. Comput.* 255 (2017), pp. 177–192.

[74] D. Kuske and G. Zetzsche. "Languages ordered by the subword order". In: *Proc. FOSSACS 2019*. Vol. 11425. LNCS. Springer, 2019, pp. 348–364.

[75] L.-K. Lee, M. Lewenstein, and Q. Zhang. "Parikh matching in the streaming model". In: *SPIRE*. Vol. 7608. LNCS. Springer, 2012, pp. 336–341. ISBN: 978-3-642-34108-3.

[76] T. Leighton and S. Rao. "Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms". In: *Journal of the ACM* 46.6 (1999), pp. 787–832.

[77] M. Lejeune, J. Leroy, and M.s Rigo. "Computing the k-binomial complexity of the Thue-Morse word". In: *CoRR* abs/1812.07330 (2018).

[78] J. Leroy, M. Rigo, and M. Stipulanti. "Generalized Pascal triangle for binomial coefficients of words". In: *CoRR* abs/1705.08270 (2017).

[79] V.I. Levenshtein. "On perfect codes in deletion and insertion metric". In: *Discrete Math. Appl.* 2 (1992), pp. 241–258.

[80] M. Lothaire. *Algebraic combinatorics on words*. Cambridge University Press, 2002.

[81] M. Lothaire. *Combinatorics on words*. Cambridge University Press, 1997.

[82] A. de Luca, A. Glen, and L.Q. Zamboni. "Rich, sturmian, and trapezoidal words". In: *Theor. Comput. Sci.* 407.1-3 (2008), pp. 569–573.

[83] D. Maier. "The complexity of some problems on subsequences and supersequences". In: *J. ACM* 25.2 (Apr. 1978), pp. 322–336. ISSN: 0004-5411.

[84] F. Manea, D. Nowotka, and M.L. Schmid. "On the solvability problem for restricted classes of word equations". In: *DLT*. Vol. 9840. LNCS. Springer, 2016, pp. 306–318.

[85] J. Manŭch. "Characterization of a word by its subwords". In: *Developments in Language Theory*. World Scientific, 1999, pp. 210–219.

[86] B. Manvel, A. Meyerowitz, A.J. Schwenk, K. Smith, and P.K. Stockmeyer. "Reconstruction of sequences". In: *Discrete Mathematics* 94.3 (1991), pp. 209–219.

[87] B. Manvel and P. K. Stockmeyer. "On reconstruction of matrices". In: *Math. Mag.* 44.4 (1971), pp. 218–221.

[88] M.H. Martin. "A problem in arrangements". In: *Bull. Amer. Math. Soc.* 40.12 (Dec. 1934), pp. 859–864.

[89] A. Mateescu, A. Salomaa, K. Salomaa, and S. Yu. *On an extension of the Parikh mapping*.

[90] A. Mateescu, A. Salomaa, and S. Yu. "Subword histories and Parikh matrices". In: *Jour. Comp. and Sys. Sci.* 68.1 (2004), pp. 1–21.

[91] Y.K. Ng and T. Shinohara. "Developments from enquiries into the learnability of the pattern languages from positive data". In: *Theoretical Computer Science* 397.1–3 (2008), pp. 150–165.

[92] P. V. O'Neil. "Ulam's conjecture and graph reconstructions". In: *Amer. Math. Monthly* 77 (1970), pp. 35–43.

[93] R. J. Parikh. "On context-free languages". In: *Jour. ACM* 13 (1966).

[94] G. Pólya. "Eine wahrscheinlichkeitsaufgabe in der kundenwerbung". In: *ZAMM* Band 10 (1930), pp. 96–97.

[95] S. Puzynina and L.O. Zamboni. "Abelian returns in Sturmian words". In: *Jour. Comb. Theo.* 120.2 (2013), pp. 390–408.

[96] N. Rampersad, J.O. Shallit, and Z. Xu. "The computational complexity of universality problems for prefixes, suffixes, factors, and subwords of regular languages". In: *Fundam. Inf.* 116.1-4 (Jan. 2012), pp. 223–236. ISSN: 0169-2968.

[97] D. Reidenbach. "Discontinuities in pattern inference". In: *Theoretical Computer Science* 397.1–3 (2008), pp. 166–193.

[98] D. Reidenbach and M.L. Schmid. "Patterns with bounded treewidth". In: *Information and Computation* 239 (2014), pp. 87–99.

[99] C. Reutenauer. *Free lie algebras*. London Math. Soc. Monogr. (N.S.) 7. Oxford University Press, 1993.

[100] G. Richomme, K. Saari, and L.Q. Zamboni. "Abelian complexity of minimal subshifts". In: *Jour. London Math. Soc.* 83.1 (2010), pp. 79–95.

[101] G. Richomme, K. Saari, and L.Q. Zamboni. "Balance and abelian complexity of the tribonacci word". In: *Adv. Appl. Math.* 45.2 (2010), pp. 212–231.

[102] M. Rigo and P. Salimov. "Another generalization of abelian equivalence: binomial complexity of infinite words". In: *Theor. Comput. Sci.* 601 (2015), pp. 47–57.

[103] G. Rozenberg and A. Salomaa, eds. *Handbook of formal languages (3 volumes)*. Springer, 1997.

[104] A. Salomaa. "Connections between subwords and certain matrix mappings". In: *TCS* 340.2 (2005), pp. 188–203.

[105] M.L. Schmid. "Characterising REGEX languages by regular languages equipped with factor-referencing". In: *Information and Computation* 249 (2016), pp. 1–17.

[106] S. Seki. “Absoluteness of subword inequality is undecidable”. In: *Theor. Comput. Sci.* 418 (2012), pp. 116–120.

[107] T. Shinohara. “Polynomial time inference of pattern languages and its application”. In: *7th IBM Symposium on Mathematical Foundations of Computer Science*. 1982, pp. 191–209.

[108] I. Simon. “Piecewise testable events”. In: *Autom. Theor. Form. Lang., 2nd GI Conf.* Vol. 33. LNCS. Springer, 1975, pp. 214–222.

[109] A. Thue. “Über unendliche Zeichenreihen”. In: *Kra. Vidensk. Selsk. Skrifter, I. Mat. Nat. Kl.* 1906.7 (1906), pp. 1–22.

[110] R.A. Wagner and M.J. Fischer. “The string-to-string correction problem”. In: *J. ACM* 21.1 (Jan. 1974), pp. 168–173. ISSN: 0004-5411. DOI: 10.1145/321796.321811. URL: http://doi.acm.org/10.1145/321796.321811.

[111] G. Zetzsche. “The complexity of downward closure comparisons”. In: *Proc. ICALP 2016*. Vol. 55. LIPIcs. 2016, 123:1–123:14.