k-Abelian Equivalence and Rationality

Julien Cassaigne, Juhani Karhumäki, Svetlana Puzynina, Markus A. Whiteland

Fundamenta Informaticae XXXI (2017) 1–26
DOI 10.3233/FI-2016-0000
IOS Press

k-Abelian Equivalence and Rationality

Julien Cassaigne
Institut de mathématiques de Marseille
address info
Marseille, France
julien.cassaigne@math.cnrs.fr

Juhani Karhumäki
Department of Mathematics and Statistics
University of Turku
20014 University of Turku, Finland
karhumak@utu.fi *

Svetlana Puzynina
University of
Paris, France/
Sobolev Institute of Mathematics
Russia
svepuz@utu.fi †

Markus A. Whiteland
Department of Mathematics and Statistics
University of Turku
20014 University of Turku, Finland
mawhit@utu.fi ‡

Abstract. Two words u and v are said to be k-abelian equivalent if, for each word x of length at most k, the number of occurrences of x as a factor of u is the same as for v. We study some combinatorial properties of k-abelian equivalence classes. Our starting point is a characterization of k-abelian equivalence by rewriting, so-called k-switching. Using this characterization we show that the set of lexicographically least representatives of equivalence classes is a regular language. From this we infer that the sequence of the numbers of equivalence classes is $\mathbb{N}$-rational. Furthermore, we sharpen an earlier result by showing that the k-abelian complexity function is asymptotic to a polynomial which depends on k and the alphabet size.

Keywords: k-abelian equivalence, Regular languages, Rational sequences

Address for correspondence: mawhit@utu.fi
*Partially supported by the Academy of Finland, grant 257857.
†
‡Partially supported by the Academy of Finland, grant 257857.

1. Introduction

k-abelian equivalence has attracted quite a lot of interest recently, see, e.g., [1, 2, 3, 4, 5, 6]. It is an equivalence relation extending abelian equivalence and allowing an infinitary approximation of the equality of words defined as follows: for an integer k, two words u and v are *k-abelian equivalent*, denoted by $u \sim_k v$, if, for each word w of length at most k, w occurs in u and v equally often.

k-abelian equivalence, originally introduced in [7], has been studied, e.g., in the following directions: avoiding *k-abelian powers* [8, 6], estimating the number of k-abelian equivalence classes, that is, *k-abelian complexity* [9], analyzing the growth and the fluctuation of the *k-abelian complexity of infinite words* [1], analyzing *k-abelian palindromicity* [3], and studying *k-abelian singletons* [10]. We continue the approach of analyzing the structure of k-abelian equivalence classes. We also study some numerical properties of the equivalence classes.

Our starting point is a *k-switching* lemma, proved in [10], which allows a characterization of k-abelian equivalence in terms of rewriting. This is quite different from the other existing characterizations, so it is no surprise that it opens new perspectives of k-abelian equivalence. This is what we intend to explore here.

A fundamental observation from the characterization of k-abelian equivalence using k-switching is that certain languages related to k-abelian equivalence classes are *regular* (or *rational*). More precisely, the union of all singleton classes forms a regular language, for any parameter k, and any size m of the alphabet. Similarly, the set of lexicographically least (or greatest) representatives of k-abelian equivalence classes forms a regular language. Summing up all minimal elements of a fixed length we obtain the number of equivalence classes of words of this length. As a consequence, we conclude that the complexity function of k-abelian equivalence, that is, the function computing the number of the equivalence classes of all lengths, is a rational function.

Everything above is algorithmic. So, given the parameter k and the size m of the alphabet, we can algorithmically compute a rational generating function giving the numbers of all equivalence classes of words of length n. However, the automata involved are – due to the non-determinism and the complementation – so huge that in practice this can be done only for very small values of the parameters. We give explicitly the above automata for values $m = 2$ and $k = 2, 3$, and 4 and $m = 3$ and $k = 2$. We see that the automaton for $m = 2$ and $k = 4$ is too large for any reasonable analysis. Using other means we are able to compute a candidate for the complexity function.

Inspired by the connection to automata theory, we study k-switching in connection with regular languages. We show that regular languages are closed under the *k-switching operation*. On the other hand, we show that regular languages are not closed under the transitive closure of this operation. Using the former result, we conclude that the union of k-abelian equivalence classes of size two is regular. On the other hand, it remains open whether this extends, instead of classes of size two, to larger classes.

Finally, using the automata-theoretic characterization of the complexity function, we solve the open problem of the original conference version, by showing that for each m and k the complexity function $\mathcal{P}_{k,m}$ giving the number of equivalence classes of words of length n, is not only of order $\Theta\left(n^{m^{k-1}(m-1)}\right)$, but actually asymptotic to $Cn^{m^{k-1}(m-1)}$ for some rational constant C.

This paper is an extended full version of the conference presentation [11].

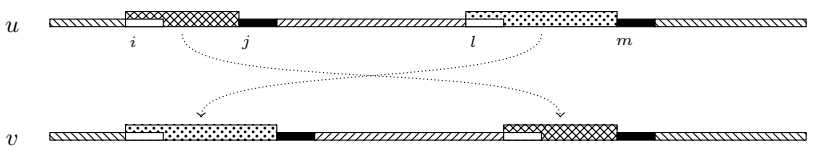

Figure 1. Illustration of a k-switching. Here $v = S_{k,u}(i, j, l, m)$; the white rectangles symbolize x and the black rectangles symbolize y.

2. Preliminaries and Notation

We recall some notation and basic terminology from the literature of combinatorics on words. We refer the reader to [12] for more on the subject.

The set of finite words over an *alphabet* Σ is denoted by Σ^* and the set of non-empty words is denoted by Σ^+. The empty word is denoted by ε. A set $L \subseteq \Sigma^*$ is called a *language*. We let $|w|$ denote the length of a word $w \in \Sigma^*$. By convention, we set $|\varepsilon| = 0$. The language of words of length n over the alphabet Σ is denoted by Σ^n.

For a word $w = a_1 a_2 \cdots a_n \in \Sigma^*$ and indices $1 \leqslant i \leqslant j \leqslant n$, we let $w[i, j]$ denote the factor $a_i \cdots a_j$. For $i > j$ we set $w[i, j] = \varepsilon$. Similarly, for $i < j$ we let $w[i, j)$ denote the factor $a_i \cdots a_{j-1}$, and we set $w[i, j) = \varepsilon$ when $i \geqslant j$. We say that a word $x \in \Sigma^*$ *has position i in w* if the word $w[i, |w|]$ has x as a prefix. For $u \in \Sigma^+$ we let $|w|_u$ denote the number of occurrences of u as a factor of w.

Two words $u, v \in \Sigma^*$ are *k-abelian equivalent*, denoted by $u \sim_k v$, if $|u|_x = |v|_x$ for all $x \in \Sigma^+$ with $|x| \leqslant k$. The relation $\sim_k$ is clearly an equivalence relation; we let $[u]_k$ denote the k-abelian equivalence class defined by u. A word u is called a *k-abelian singleton* if $|[u]_k| = 1$.

In [10], k-abelian equivalence is characterized in terms of rewriting, namely by *k-switching*. For this we define the following. Let $k \geqslant 1$ and let $u \in \Sigma^*$. Suppose that there exist $x, y \in \Sigma^{k-1}$, not necessarily distinct, and indices i, j, l and m, with $i < j \leqslant l < m$, such that x has positions i and l in u and y has positions j and m in u. In other words, we have

$$u = u[1, i) \cdot u[i, j) \cdot u[j, l) \cdot u[l, m) \cdot u[m, |u|],$$

where both $u[i, |u|]$ and $u[l, |u|]$ begin with x and both $u[j, |u|]$ and $u[m, |u|]$ begin with y. Furthermore, $u[i, j)$, $u[l, m) \neq \varepsilon$ but we allow $l = j$, in which case $y = x$ and $u[j, l) = \varepsilon$. We define a *k-switching* on u, denoted by $S_{u,k}(i, j, l, m)$, as

$$S_{u,k}(i, j, l, m) = u[1, i) \cdot u[l, m) \cdot u[j, l) \cdot u[i, j) \cdot u[m, |u|]. \tag{1}$$

A k-switching operation is illustrated in Figure 1.

Example 2.1. Let $u = aababababaaabab$ and $k = 4$. Let then $x = aba$, $y = bab$, $i = 2$, $j = 3$, $l = 4$ and $m = 11$. We then have

$$u = a \cdot a \cdot b \cdot ababaaa \cdot bab$$
$$S_{u,4}(i, j, l, m) = a \cdot ababaaa \cdot b \cdot a \cdot bab.$$

Note here that the occurrences of x are overlapping. With $i = 2$, $j = l = 4$, and $m = 10$ we obtain the same word as above:

$$u = a \cdot ab \cdot ababaa \cdot abab$$
$$S_{u,4}(i, j, j, m) = a \cdot ababaa \cdot ab \cdot abab.$$

In this example we have $j = l$, whence $x = y = aba$ and $u[j, l) = \varepsilon$.

Let us define a relation R_k of Σ^* by uR_kv if and only if v is obtained from u by a k-switching. Now R_k is clearly symmetric, so that the reflexive and transitive closure R_k^* of R_k is an equivalence relation on Σ^*. In [10], k-abelian equivalence is characterized using R_k^*:

Lemma 2.2. For $u, v \in \Sigma^*$, we have $u \sim_k v$ if and only if uR_k^*v.

We need a few basic properties of *regular* (or *rational*) languages, such as equivalent definitions of regular languages with various models of finite automata, e.g., non-deterministic finite automata which can read the empty word (ε-NFA), and some basic closure properties of regular languages. We refer to [13] for this knowledge. In addition to classical language theoretical properties, we use the theory of *languages with multiplicities*. This counts how many times a word occurs in a language. This leads to the theory of $\mathbb{N}$-*rational sets*. Using the terminology of [14], a multiset over Σ^* is called $\mathbb{N}$-*rational* if it is obtained from finite multisets by applying finitely many times the rational operations *product*, *union*, and taking *quasi-inverses*, i.e., *iteration* restricted to ε-free languages. Further, a unary $\mathbb{N}$-rational subset is referred to as an $\mathbb{N}$-*rational sequence*. We refer to [14] for more on this topic. The basic result we need is (see [14]):

Proposition 2.3. Let $\mathcal{A}$ be a non-deterministic finite automaton over the alphabet Σ. The function $f_{\mathcal{A}} : \Sigma^* \to \mathbb{N}$ defined as

$$f_{\mathcal{A}}(w) = \#\text{of accepting paths of } w \text{ in } \mathcal{A}$$

is $\mathbb{N}$-rational. In particular, the function $\ell_{\mathcal{A}} : \mathbb{N} \to \mathbb{N}$,

$$\ell_{\mathcal{A}}(n) = \#\text{of accepting paths of length } n \text{ in } \mathcal{A} \tag{2}$$

is an $\mathbb{N}$-rational sequence. Consequently, the *generating function* for $\ell_{\mathcal{A}}$ is a rational function.

3. Properties of k-Switchings

Our starting point for the study of structural properties of k-abelian equivalence classes is the characterization of k-abelian equivalence in terms of k-switchings. We proceed to describe a k-switching operation on languages. We show that this operation preserves regularity. That is, given a regular language L, the language obtained by this operation is also regular. This result will be used later on.

We now describe k-switchings on languages. For a language $L \subset \Sigma^*$, we define the k-switching of L, denoted by $R_k(L)$, as the language

$$R_k(L) = \{w \in \Sigma^* \mid wR_kv \text{ for some } v \in L\}.$$

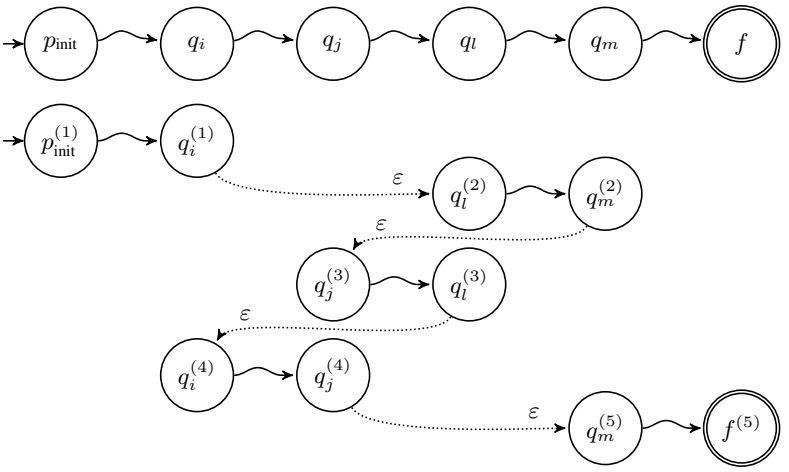

Figure 2. The computation of automaton $\mathcal{A}$ on an accepted word u (in continuous lines) and a computation of $\mathcal{A}'$ on $S_{k,u}(i,j,l,m)$ (in dotted lines). The automaton $\mathcal{A}'$ non-deterministically guesses the positions i,j,l, and m and jumps to the corresponding states non-deterministically. Verification is then easy: the factors of length $k-1$ starting at the first and third guess have to be equal, and so do the factors occurring at the second and fourth guesses.

Similarly, we define $R_k^*(L) = \bigcup_{n\in\mathbb{N}} R_k^n(L) = \bigcup_{w\in L}[w]_k$.

Note that, from a regular language L, it is straightforward to identify all words that admit a k-switching (i.e., the words on the top row of Figure 1). It is not so clear that, by performing all possible k-switchings on all words of L (i.e., taking the union of all words on the bottom row of Figure 1), the obtained language is also regular. We give a proof sketch here. For a full proof, see [11].

Theorem 3.1. Let L be a regular language. Then $R_k(L)$ is also regular.

Proof:
We start with a DFA $\mathcal{A}$ recognizing the language L and construct an NFA $\mathcal{A}'$ with ε-transitions. The automaton $\mathcal{A}'$ basically guesses at which states the switchings occur at. Afterwards, it verifies the guesses. The verification requires only finite memory, since all is needed to verify is the states the switching has occurred at, and that the factors of length $k-1$ occurring at the positions of the switchings are valid. The idea of the computation of $\mathcal{A}'$ is depicted in Figure 2.

□

Remark 3.2. It is worth noticing that, in a k-switching, the word v is obtained from u by changing the order of the factors $u[i,j)$ and $u[l,m)$. They are of unbounded length and hence cannot be remembered by a finite automaton. Instead, in the proof, we just remember the corresponding states at positions of i,j,l, and m in the automaton $\mathcal{A}$ recognizing u, and do simultaneously all the switchings where the automaton $\mathcal{A}$ performs these state transitions. We shall return to these considerations in our later discussions.

Remark 3.3. This result may also be proved using MSO logic for words, as suggested by one of the anonymous referees of the conference version.

The following example shows that the family of regular languages is not closed under the language operation R_k^*.

Example 3.4. Fix $k \geqslant 1$ and let $L = (ab^k)^+$. It is straightforward to verify by, e.g., comparing the number of occurrences of factors of length k that

$$R_k^*(L) = \left\{ab^{r_1}ab^{r_2}\cdots ab^{r_n} \mid n \geqslant 1, r_i \geqslant k-1, \textstyle\sum_{i=1}^n r_i = nk\right\}.$$

Let now h be a morphism defined by $h(a) = ab^{k-1}$ and $h(b) = b$. It is again straightforward to show that $h^{-1}(R_k^*(L)) = \{w \in a\{a,b\}^* \mid |w|_a = |w|_b\}$, which is clearly not regular. It follows that $R_k^*(L)$ is not regular.

4. On the Number of k-Abelian Equivalence Classes

In this section we focus on the number $\mathcal{P}_{k,m}(n)$ of k-abelian equivalence classes of words of length n over Σ, $|\Sigma| = m$, where k and an m are fixed. We first recall a result from [9]:

Theorem 4.1. We have, for k and m fixed, $\mathcal{P}_{k,m}(n) = \Theta(n^{m^{k-1}(m-1)})$, where the constants implied by Θ depend on k and m.

We are also interested in the number $\mathcal{S}_{k,m}(n)$ of k-abelian singletons of length n over Σ, $|\Sigma| = m$, where k and an m are fixed. We recall a result proved in [10].

Theorem 4.2. For k and m fixed, we have $\mathcal{S}_{k,m}(n) = \mathcal{O}(n^{N_m(k-1)-1})$, where the constant implied by $\mathcal{O}$ depends on k and m. Here $N_m(l) = \frac{1}{l}\sum_{d|l}\varphi(d)m^{l/d}$ is the number of conjugacy classes (or necklaces) of words of length l over an m-letter alphabet.

The main results of this section are the following:

Theorem 4.3. The sequences $\mathcal{P}_{k,m}(n)$ and $\mathcal{S}_{k,m}(n)$ are $\mathbb{N}$-rational.

In order to prove these, we define the following languages. Here $\trianglelefteq$ denotes a lexicographic ordering of Σ^* and m denotes the size of the alphabet $|\Sigma|$.

$$\begin{aligned} L_{k,\Sigma,\min} &= \{w \in \Sigma^* \mid w \trianglelefteq u \text{ for all } u \in [w]_k\};\\ L_{k,\Sigma,\max} &= \{w \in \Sigma^* \mid w \trianglerighteq u \text{ for all } u \in [w]_k\}; \text{ and}\\ L_{k,\Sigma,\text{sing}} &= \{w \in \Sigma^* \mid |[w]_k| = 1\}. \end{aligned}$$

In other words, $L_{k,\Sigma,\min}$ (resp., $L_{k,\Sigma,\max}$) is the language of lexicographically minimal (resp., maximal) representatives of k-abelian equivalence classes over Σ, while $L_{k,\Sigma,\text{sing}}$ is the language of k-abelian singletons over Σ. We shall often omit the subscripts k and Σ when they are clear from context.

We also recall a technical lemma from [10], a refinement of our Lemma 2.2.

Lemma 4.4. Let $u \sim_k v$ with $u \neq v$. Let p be the longest common prefix of u and v. Then there exists $z \in \Sigma^*$ such that zR_ku and the longest common prefix of z and v has length at least $|p| + 1$.

We are now ready for our main technical tool.

Lemma 4.5. The languages $L_{k,\Sigma,\min}$, $L_{k,\Sigma,\max}$, and $L_{k,\Sigma,\text{sing}}$ are regular for any $k \geqslant 1$ and Σ.

Proof:
Let u be the minimal element in $[u]_k$. If there exists a k-switching on u which yields a new element, it has to be lexicographically greater than u. In particular, u does not contain factors from the language

$$\big((xb\Sigma^* \cap \Sigma^* y)\, \Sigma^* \cap \Sigma^* x\big) a\Sigma^* \cap \Sigma^* y,$$

where $x, y \in \Sigma^{k-1},\ a, b \in \Sigma,\ a < b$. On the other hand, by the above lemma, any word u avoiding such factors is lexicographically least in $[u]_k$. We thus have

$$L_{k,\Sigma,\min} = \bigcap_{\substack{x,y\in\Sigma^{k-1}\\ a,b\in\Sigma,\ a\lhd b}} \overline{\Sigma^*\Big(\big((xb\Sigma^* \cap \Sigma^* y)\, \Sigma^* \cap \Sigma^* x\big) a\Sigma^* \cap \Sigma^* y\Big)\Sigma^*}, \tag{3}$$

where, for a regular expression R, $\overline{R}$ denotes the *complement* language $\Sigma^* \setminus L(R)$.

Similarly, for $L_{\max}$, by reversing $a \lhd b$ to $a \rhd b$ in (3), we obtain the claim.

Finally, $L_{\text{sing}} = L_{\min} \cap L_{\max}$ so that L_{sing} is regular. Another, perhaps more informative, way to see this is as follows: for k-abelian singletons, we are avoiding all possible k-switchings that give a different word. By requiring $a \neq b$, instead of $a \lhd b$, in (3), we obtain the expression for L_{sing}. □

Remark 4.6. Observe that the languages $L_{\min}$, $L_{\max}$, and L_{sing} are *factorial* (a language L is said to be factorial if, for every $u \in L$, any factor w of u is in L). Indeed, they are languages obtained by avoiding certain patterns. Another simple way to see this is as follows. Suppose that $u = u_1u_2u_3 \in L_{\min}$ and $u_2 \notin L_{\min}$. We may take u_2' for which $u_2' \sim_k u_2$ and $u_2' \lhd u_2$. By replacing u_2 by u_2' we obtain $u' = u_1u_2'u_3 \lhd u$ and $u' \sim_k u$, a contradiction.

We are now ready to prove Theorem 4.3.

Proof:
Consider first the language $L_{\min}$ and a DFA $\mathcal{A}$ recognizing it. We transform the automaton to a unary NFA $\mathcal{A}'$ by identifying all input letters. Since $\mathcal{A}$ is deterministic, the transformation is *faithful*, that is, for each word w accepted by $\mathcal{A}$, there exists a unique corresponding accepting path in $\mathcal{A}'$, and vice versa. By the construction of $\mathcal{A}'$, $\ell_{\mathcal{A}'}(n) = \mathcal{P}_{k,m}(n)$ for all $n \in \mathbb{N}$. The claim follows for $\mathcal{P}_{k,m}$. The case of $\mathcal{S}_{k,m}$ is identical. □

4.1. Automata and Complexities for Small Values of k and m

We now give some examples illustrating the results obtained above for small values of k and m. We also compute closed formulas for $\mathcal{P}_{k,m}$ and $\mathcal{S}_{k,m}$ for some small values of k and m.

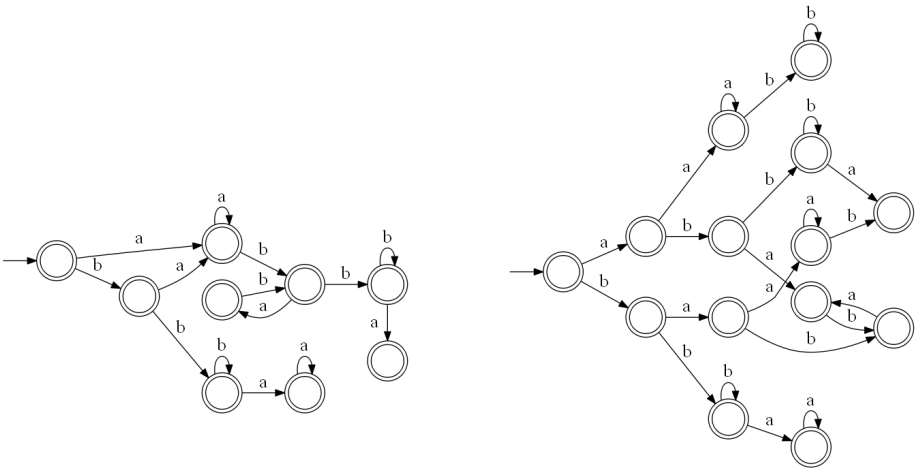

Figure 3. DFAs recognizing the minimal representatives of 2-abelian equivalence classes (left) and 2-abelian singletons (right) over the alphabet $\{a, b\}$ with ordering $a \lhd b$.

Example 4.7. In figure Figure 3, we have two minimal DFAs, one recognizing the minimal representatives of 2-abelian equivalence classes and the other recognizing 2-abelian singletons over $\Sigma = \{a, b\}$. In Figures 4 and 5 we illustrate the minimal DFA recognizing $L_{\min}$ for $m = 3$, $k = 2$ and $m = 2$, $k = 3$, respectively. In Figures 6 and 7 we illustrate the minimal DFA recognizing L_{sing} for $m = 2$, $k = 3$ and $m = 3$, $k = 2$, respectively. The sink states are not included in the figures. We also note that all other states are accepting, since the languages are factorial by Remark 4.6.

Proposition 4.8.

$$\begin{aligned}
&\text{For all } n \geqslant 1,\ \mathcal{P}_{2,2}(n) = n^2 - n + 2;\\
&\text{for all } n \geqslant 2,\ \mathcal{P}_{3,2}(n) = \tfrac{1}{18}n^4 - \tfrac{5}{18}n^3 + \tfrac{65}{36}n^2 - \tfrac{23}{6}n - \tfrac{1}{8}(-1)^n +\\
&\qquad\qquad + \tfrac{2}{27}e^{-\frac{\pi i}{3}}(e^{\frac{2\pi i}{3}})^n + \tfrac{2}{27}e^{\frac{\pi i}{3}}(e^{-\frac{2\pi i}{3}})^n + \tfrac{1307}{216};\ \text{and}\\
&\text{for all } n \geqslant 4,\ \mathcal{P}_{2,3}(n) = \tfrac{1}{960}n^6 + \tfrac{7}{320}n^5 + \tfrac{67}{384}n^4 - \tfrac{19}{32}n^3 + \tfrac{1457}{480}n^2 -\\
&\qquad\qquad - (\tfrac{1569}{640} + \tfrac{3}{128}(-1)^n)n + \tfrac{741}{256} + \tfrac{27}{256}(-1)^n.
\end{aligned}$$

Proposition 4.9.

$$\begin{aligned}
&\text{For all } n \geqslant 4,\ \mathcal{S}_{2,2}(n) = 2n + 4;\\
&\text{for all } n \geqslant 6,\ \mathcal{S}_{3,2}(n) = 3n^2 + 27n - 63;\ \text{and}\\
&\text{for all } n \geqslant 9,\ \mathcal{S}_{2,3}(n) = \frac{1}{2}n^2 + 16n + \frac{2}{3}(e^{\frac{2\pi i}{3}n} + (e^{-\frac{2\pi i}{3}})^n) - \frac{535}{12} - \frac{3}{4}(-1)^n.
\end{aligned}$$

We give only a proof sketch of Propositions 4.8 and 4.9.

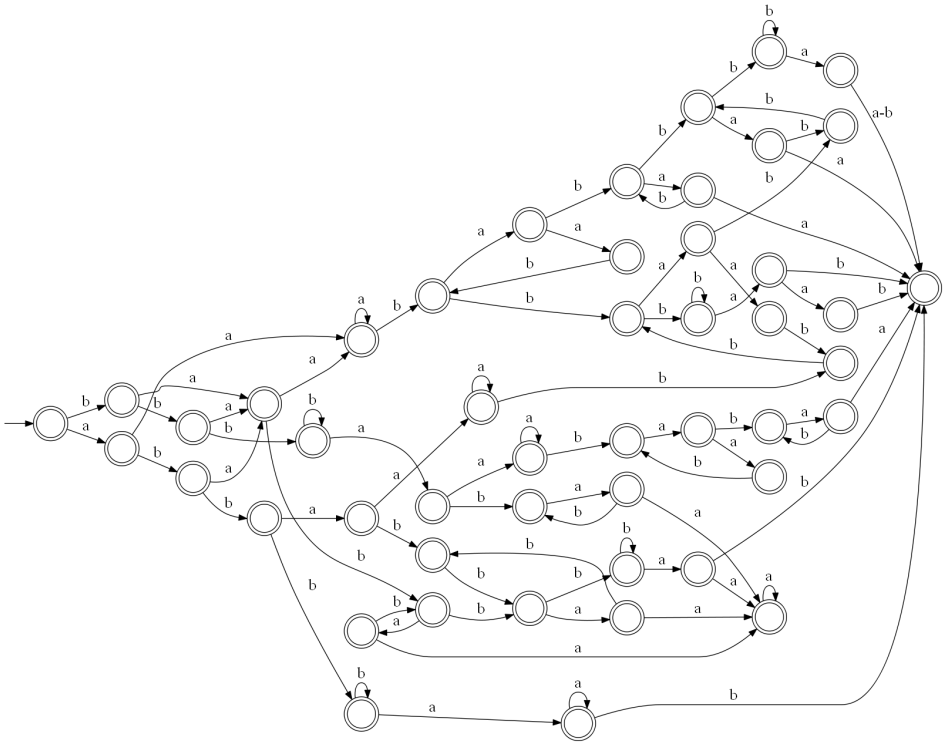

Figure 4. A DFA recognizing the minimal representatives of 3-abelian equivalence classes over $\Sigma = \{a, b\}$ with ordering $a \lhd b$.

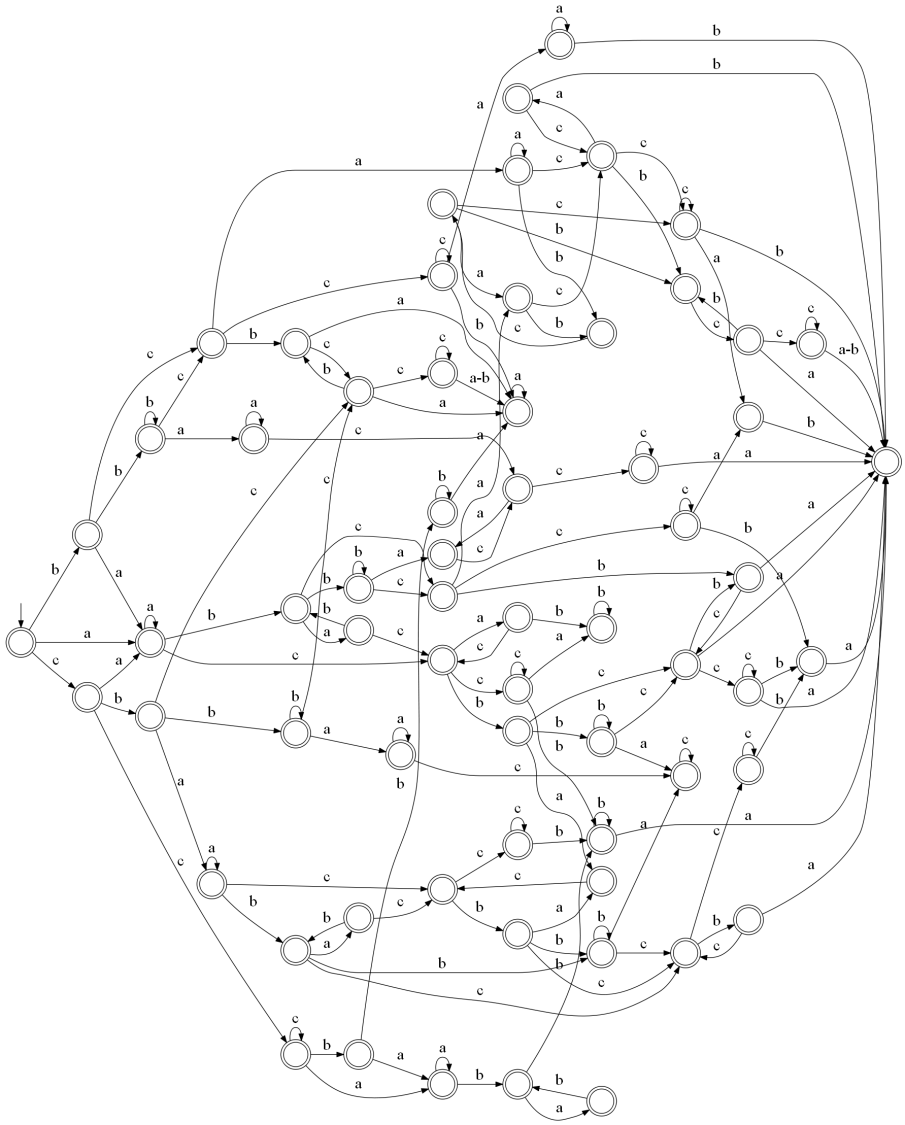

Figure 5. The minimal DFA recognizing $L_{\min}$ for $m = 3$ and $k = 2$ with ordering $a \lhd b \lhd c$.

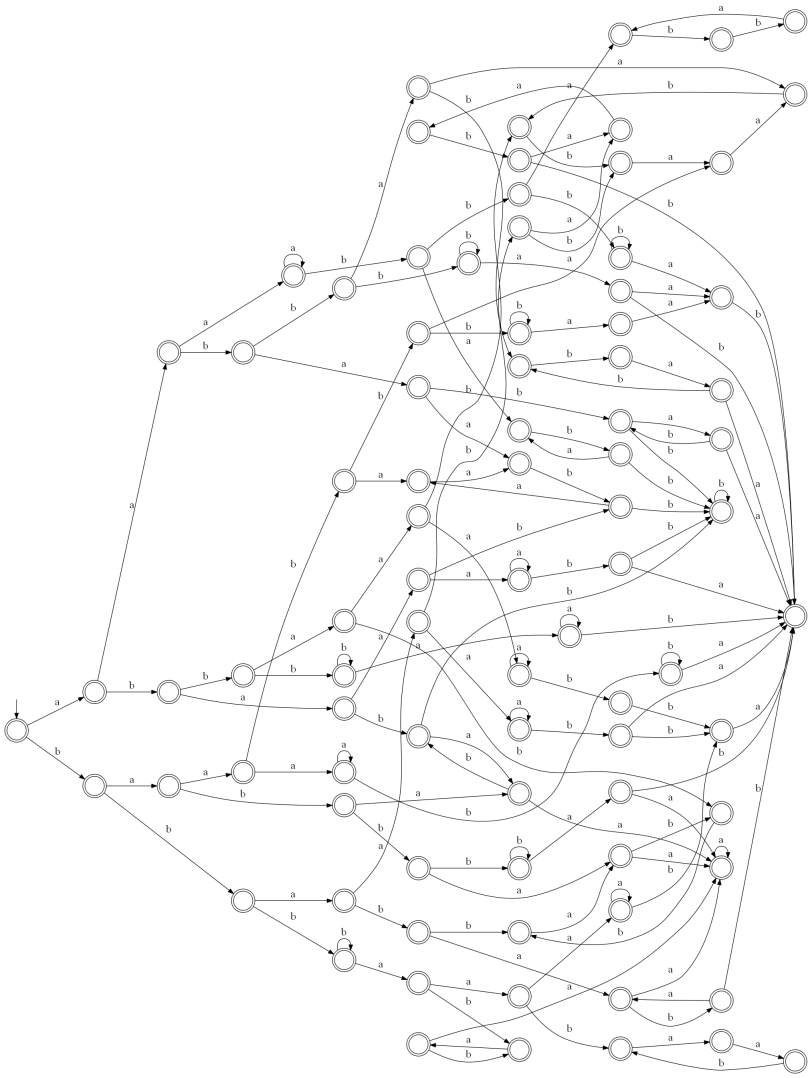

Figure 6. The minimal DFA recognizing 3-abelian singletons over $\{a, b\}$.

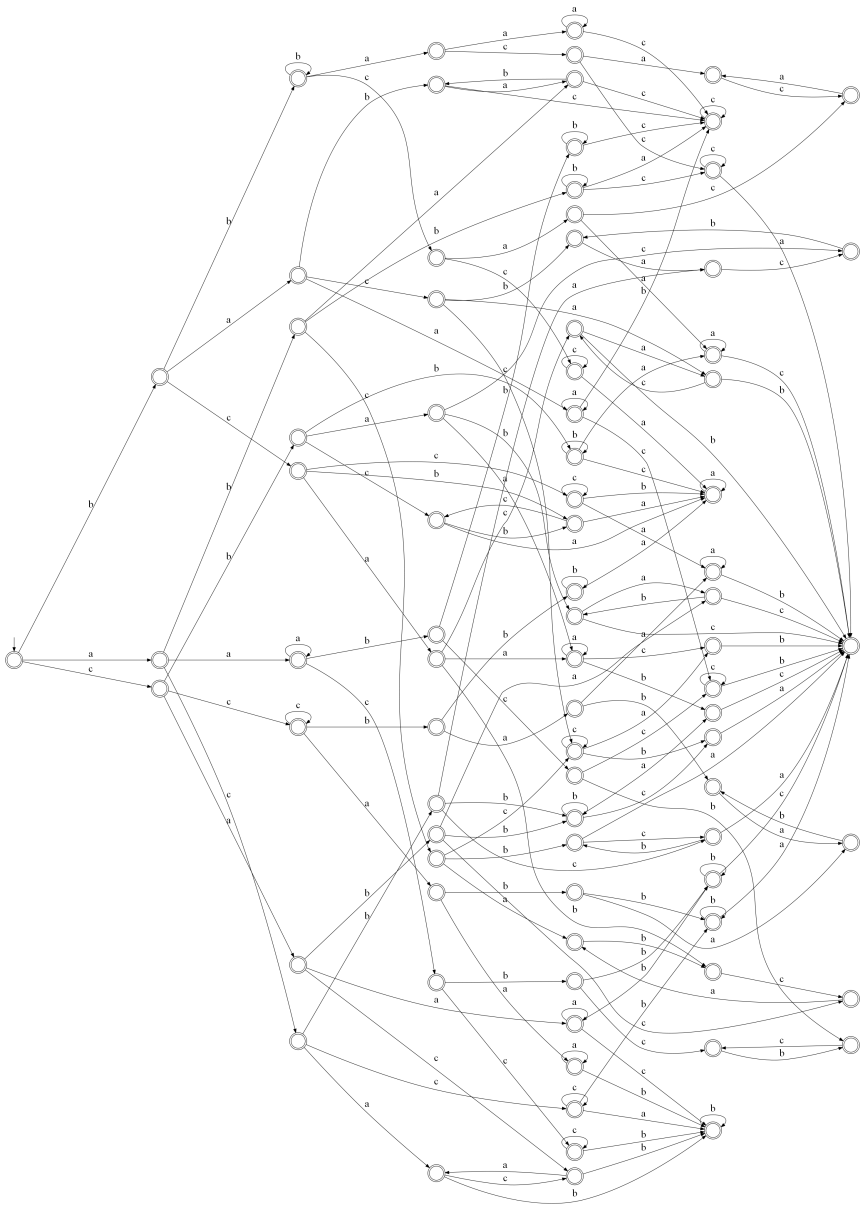

Figure 7. A DFA recognizing 2-abelian singletons over $\{a, b, c\}$.

Proof:
Using the idea of the proof of Theorem 4.3, we first construct deterministic automata for $L_{\min}$ and L_{sing} for small k and m. We then construct the the unary automaton $\mathcal{A}'$ and use these automata to compute the function ℓ: Let A be the adjacency matrix of $\mathcal{A}'$. It is known that, for all large enough n,

$$\ell_{\mathcal{A}'}(n) = \sum_{\lambda \in Eig(A)} p_\lambda(n)\lambda^n, \tag{4}$$

where the summation is taken over all distinct eigenvalues of A, and p_λ is a complex polynomial of degree at most $\mu_\lambda - 1$ for each eigenvalue λ. Here μ_λ is the multiplicity of λ as a root of the *minimal polynomial* of A. (See, for instance, [13, 15, 16].)

We give a brief sketch of the methods used to compute the closed forms of the complexities. For details see [16]. Let M be the adjacency matrix of the transformed unary NFA of the DFA of our language L. We identify the vector e_{init} corresponding to the initial state. We also construct the vector e_F corresponding to all accepting states. We then have that $\mathcal{P}_{k,m}(n) = e_{\text{init}} M^n e_F^T$.

To calculate $\mathcal{P}_{k,m}$ explicitly, the *Jordan decomposition* $M = SJS^{-1}$ of M is computed. A closed form for J^n, $n \geqslant 4$, can be easily computed, from which we obtain a closed form for $M^n = SJ^nS^{-1}$. It is then a simple task to compute $e_{\text{init}} M^n e_F^T$.

We also use the idea of curve fitting. This is done by first computing the eigenvalues λ, the minimal polynomial $m(x)$ of M, and the multiplicities μ_λ of the roots λ of $m(x)$. For each λ we compute the coefficients of the polynomials $p_\lambda(x)$ in (4) by fitting a curve to data points which are small values of the function $P_{k,m}(n)$. We note the data point corresponding to the smallest length n_0 should be large enough. The number n_0 is bounded by the largest *Jordan block* in the Jordan decomposition. This, in turn, is bounded by the maximal multiplicity among the roots of $m(x)$. □

The formulae for $\mathcal{P}_{2,2}$ and $\mathcal{S}_{2,2}$ have previously been proved, using different methods, in [17] and [10], respectively. The formulae obtained coincide with values previously computed by Eero Harmaala ($\mathcal{P}_{2,3}$ and $\mathcal{P}_{3,2}$ for $n = 2, \ldots, 18$ and $n = 4, \ldots, 21$, respectively) (private communication). We also computed the first few values of $S_{2,3}(n)$ and $S_{3,2}(n)$ and checked that they coincide with the formulae obtained. We note that the *On-Line Encyclopedia of Integer Sequences* (`http://oeis.org`, accessed June 10, 2016) doesn't contain any of the above sequences.

The methods used here are far from being practical for computing closed formulae for larger values of k and m, as is illustrated by the following example.

Example 4.10. Consider the binary alphabet: the minimal DFA recognizing $L_{k,\min}$ for $k = 2$ and $k = 3$ have 10 and 49 states in their minimal DFA, respectively. While these automata are easily handled, the task becomes a computationally challenging problem already for $k = 4$: $L_{4,\min}$ has a minimal DFA with 936 states. Here we give a candidate function for $\mathcal{P}_{4,2}$. We computed the values $\mathcal{P}_{4,2}(n)$ for $n = 4, \ldots, 49$ and then fit a curve f using these points to obtain the closed form below. We note that f agrees with $\mathcal{P}_{4,2}$ for the value $n = 50$ as well. We have not verified whether $f = \mathcal{P}_{4,2}$ identically, that is, whether $n_0 = 4$ is "large enough". In the following $\langle a_0, \ldots, a_{q-1} \rangle_n$ denotes the

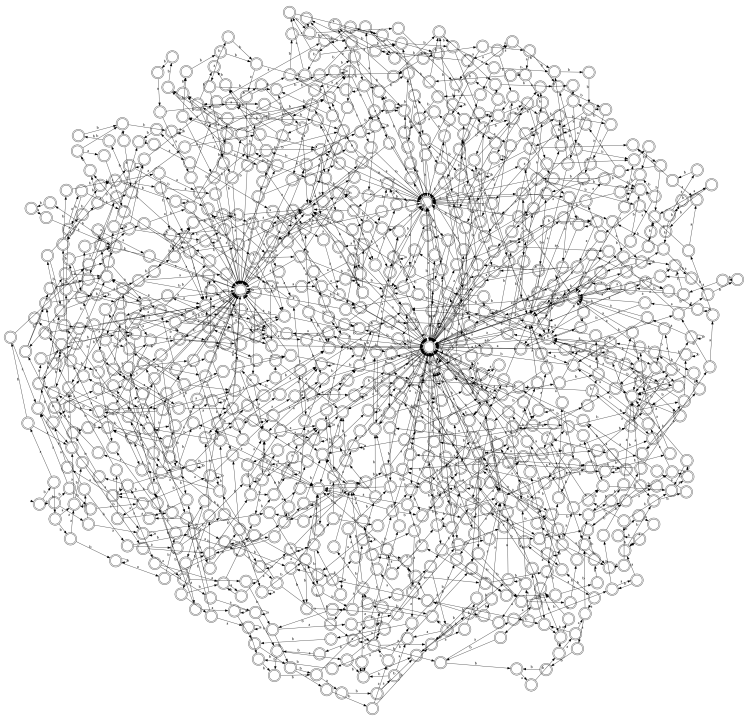

Figure 8. A DFA recognizing the minimal representatives of 4-abelian classes over $\{a, b\}$.

q-periodic function taking the value q_i when $n = i \pmod q$.

$$\begin{aligned}
P_{4,2}(n) = {} & \tfrac{283}{(512\cdot 243\cdot 25\cdot 49)}n^8 + \tfrac{223}{(32\cdot 243\cdot 25\cdot 49)}n^7 + \tfrac{2\,657}{(256\cdot 243\cdot 25\cdot 7)}n^6 \\
& - \tfrac{731}{(8\cdot 243\cdot 25\cdot 7)}n^5 + \tfrac{14\,111}{(32\cdot 243\cdot 25)}n^4 - \tfrac{1\,609}{(4\cdot 27\cdot 25)}n^3 \\
& + \tfrac{1\,850\,177\,503}{(512\cdot 729\cdot 25\cdot 49)}n^2 - \tfrac{3\,779\,893}{(64\cdot 729\cdot 7)}n + \tfrac{81\,883\,529\,107}{1\,024\cdot 729\cdot 125\cdot 49)} \\
& + \tfrac{1}{512}\langle 1,-1\rangle_n \cdot n^2 - \tfrac{5}{64}\langle 1,-1\rangle_n \cdot n + \tfrac{489}{1\,024}\langle 1,-1\rangle_n \\
& + \tfrac{1}{(2\cdot 729)}\langle -7,5,2\rangle_n \cdot n^2 + \tfrac{2}{729}\langle 38,-7,-31\rangle_n \cdot n + \tfrac{1}{(4\cdot 729)}\langle -1\,853,571,1\,282\rangle_n \\
& + \tfrac{1}{16}\langle -2,1,2,-1\rangle_n + \tfrac{2}{125}\langle 21,6,-4,-14,-9\rangle_n + \tfrac{1}{8}\langle -1,1,1,1,-1,-1\rangle_n \\
& + \tfrac{4}{49}\langle 2,1,-1,-4,-1,1,2\rangle_n.
\end{aligned}$$

Remark 4.11. The exponential blow-up of the computation time is due to complementation and nondeterminism of the automata obtained from the regular expressions (3). Also, by Theorem 4.1, the automaton obtained from (3) has to grow necessarily exponentially with respect to k when the alphabet is fixed; some of the polynomials p_λ in (4) have degree $m^{k-1}(m-1)$. For the case of k-abelian singletons, Theorem 4.2 does not give a large blow-up immediately, though in [10] it is conjectured that $\mathcal{S}_{k,m}(n) = \Theta(n^{N_m(k-1)-1})$. If true, a large blow-up in the number of states would be guaranteed.

5. On the Asymptotics of $\mathcal{P}_{k,m}(n)$

In Propositions 4.8 and 4.9 we notice that the functions $\mathcal{P}_{k,m}$ and $\mathcal{S}_{k,m}$ are asymptotic to certain polynomials, at least for small values for k and m. In other words, the constants mentioned in Theorem 4.1 are equal. In the case of $\mathcal{S}_{k,m}$, a closer inspection of the proof of Theorem 4.2 in [10] reveals that $\mathcal{S}_{k,m}(n) = \Theta(n^r)$ for some $r \in \mathbb{N}$, $r \leqslant N_m(k-1) - 1$ (recall Theorem 4.2 for notation). This section is devoted to proving the following result:

Theorem 5.1. The sequence $\mathcal{P}_{k,m}(n)$ is asymptotic to $Cn^{m^{k-1}(m-1)}$ for some rational constant C depending on k and m.

We consider the values $k \geqslant 1$ and $m \geqslant 2$ fixed constants for the remainder of this section. Further, we let $L_{\min} = L_{k,m,\min}$ unless otherwise stated.

We set some terminology and notation for directed graphs. For a directed graph $G = (V, E)$, the set V of vertices is denoted by $V(G)$, while the set E of its edges is denoted by $E(G)$. Here we allow multiple edges from one vertex to another, but in this case the edges should be labeled. A sequence $(x_i)_{i=0}^t$ of vertices $x_i \in V(G)$ such that $(x_i, x_{i+1}) \in E(G)$ for each $i \in [0, t-1]$ is called a *walk* (in G). If $x_i \neq x_j$ for each $i \neq j$, we say that W is a path. If the walk $(x_i)_{i=0}^{t-1}$ is a path and $x_t = x_0$, we call the walk $(x_i)_{i=0}^t$ a cycle. Note that in the case of multigraphs (here we consider underlying graphs of finite automata), a walk $W = (x_i)_{i=0}^t$ is uniquely defined by the sequence of edges used, not only the sequence of vertices.

In addition to underlying graphs of finite automata, we shall be considering *de Bruijn graphs*. For any $k \geqslant 1$ and alphabet Σ, the *de Bruijn graph* $dB_\Sigma(k)$ *of order* k *over* Σ is defined as a directed graph for which $V(dB_\Sigma(k)) = \Sigma^{k-1}$. There is an edge $(x, y) \in E(dB_\Sigma(k))$ if there exists a letter $a \in \Sigma$ such that $xa \in \Sigma^k$ ends with y. In this case (x, y) is denoted by (x, a). We shall omit Σ from the subscript when there is no danger of confusion. We note that any word $u \in \Sigma^*$ of length at least $k-1$ defines a walk W_u in $dB(k)$ and vice versa. Thus a (long enough) word $u \in \Sigma^*$ should be considered as a walk in $dB(k)$ and vice versa.

For a given walk $W = (x_i)_{i=0}^t$ we define $V(W)$ as the set of vertices along W and $E(W) = \{(x_i, x_{i+1}) \mid 0 \leqslant i < t\}$. We then call the (connected) graph $(V(W), E(W))$ the *restriction of* G *with respect to* W. We say that a cycle $C = (y_j)_{j=0}^s$ *occurs along* W if, for some $i \in [0, t]$, $x_{i+r} = y_r$ for all $r \in [0, s]$. The walk W is then said to *enter* C *at position* i if $x_{i+r} = y_r$ for all $r \in [0, s]$ and either $i = 0$ or $x_{i-1} \neq y_{s-1}$. Similarly, W is said to *leave* C *at position* i if W enters C at some position j, $x_{j+r} = y_{r \pmod s}$ for all $r \leq i - j$, and $x_{i+1} \neq y_{i-j+1 \pmod s}$.

Definition 5.2. Let G be a graph and W a walk in G. We say that W is *cycle-deterministic* if, for each cycle C occurring along W, W does not enter C at two distinct positions. The set of cycles along a cycle-deterministic walk W is denoted by $C(W)$.

Let then $B \subseteq V(G)$ be fixed and let W a cycle-deterministic walk in G starting from some $v \in B$. We call W B*-saturated* if $|C(W)|$ is maximal among all such walks. We omit the prefix B and simply call W *saturated* whenever B is clear from context.

Note that for a cycle-deterministic walk W and a cycle $C \in C(W)$, some of the vertices and the edges occurring in C may be traversed by W after leaving C. We are going to use the notion of cycle-

deterministic walks in two ways. In what follows, we use the notion to describe certain sub-automata of deterministic finite automata. In the cases we discuss, such walks have an even stronger property (see the next subsection). Later on, we are going to map walks in deterministic automata to walks in de Bruijn graphs and vice versa.

Example 5.3. Consider the de Bruijn graph $dB(3)$. The walk $W = (x_i)_{i=0}^{12}$ defined by $u = aaaabaabaabaa$ is cycle-deterministic. Indeed, u enters the cycle (or loop) (aa, aa) at position 0 and leaves the cycle at position 2. It does not enter the loop (aa, aa) after that. Further, the cycle (aa, ab, ba, aa) is entered at postion 2 and W does not leave this cycle. On the other hand, the walk W' defined by the word ua is not cycle-deterministic, as W' enters the cycle (aa, aa) at position 0 and position 11. The cycle (aa, ab, ba, aa) is now left at position 11.

Any walk W in the underlying graph G of a deterministic finite automaton $\mathcal{A}$ defines a sub-automaton of $\mathcal{A}$: the initial and final states being the state W starts from and the state W ends in, respectively. We denote this automaton by $\mathcal{A}_W$. Note that distinct walks may define the same automaton. We shall only be interested in walks starting from the unique initial state $\mathcal{A}$. In what follows, for an automaton $\mathcal{A}$ and initial state set I, we call the automaton $\mathcal{A}_W$ *saturated* if the walk W is I-saturated. The set of all saturated automata $\mathcal{A}_W$ of $\mathcal{A}$ is denoted by $\mathcal{W}(\mathcal{A})$.

5.1. On the Asymptotic Complexity of Regular Languages

For a language $L \subseteq \Sigma^*$, the complexity function $\mathcal{C}_L : \mathbb{N} \to \mathbb{N}$ of L is defined as $\mathcal{C}_L(n) = |\Sigma^n \cap L|$. Let us now turn to the theory of regular languages L having polynomially bounded complexity, that is, $\mathcal{C}_L(n) = \mathcal{O}(n^k)$ for some $k \in \mathbb{N}$. By Theorems 4.1 and 4.2, our languages $L_{\min}$ and L_{sing} fall into this category. We recall the following result from [18] (see also [16]):

Theorem 5.4. For a regular language L, we have $\mathcal{C}_L(n) = \mathcal{O}(n^k)$ for some $k \geqslant 0$ if and only if L can be represented as a finite union of regular expressions of the form $z_0 y_1^* z_1 \cdots y_t^* z_t$ with a non-negative integer $t \leqslant k + 1$, where $z_0, y_i, z_i \in \Sigma^*$ for all $i = 1, \ldots, t$.

This fact can be seen from a DFA $\mathcal{A}$ accepting such a language. Indeed, all walks in the underlying graph of $\mathcal{A}$ are necessarily cycle-deterministic and, furthermore, once a cycle C has been exited, none of the vertices $V(C)$ are visited later on. To get an idea of the notion, see, e.g., the automata in Figures 3 – 7.

Consider then a sub-automaton $\mathcal{A}_W$ such that the defining walk W starts from the initial state of $\mathcal{A}$ and ends in an accepting state of $\mathcal{A}$. Then $\mathcal{A}_W$ recognizes the language $z_0 y_1^* z_1 \cdots y_t^* z_t$, where $z_i \in \Sigma^*$ (resp., $y_i \in \Sigma^+$) are the labels of the paths connecting the initial vertices of the cycles along W (resp., the labels of the cycles along W). Moreover, by the determinism of $\mathcal{A}$, the longest common prefix p_i of y_i and z_i is *proper*, that is, $|p_i| < |y_i|$ and $|p_i| < |z_i|$ for each $i = 1, \ldots, t-1$. In particular, $z_i \in \Sigma^+$ for each $i = 1, \ldots, t-1$. It is worth noting that the majority of the elements of $L(\mathcal{A})$ is recognized by the saturated sub-automata $\mathcal{W}(\mathcal{A})$.

We now turn to the *generating functions* of the automata described above. For a general treatment on the topic of generating functions, see [19]. We shall briefly recall results concerning formal languages. To this end, let $L \subseteq \Sigma^*$ be a language. The *(ordinary) generating function* G_L of L is defined

as the formal power series

$$G_L(x) = \sum_{k=0}^{\infty} a_k x^k,$$

where $a_k = \mathcal{C}_L(k)$ for each $k \in \mathbb{N}$. To avoid cluttering the text we shall often omit the summation bounds. For two generating functions $G_1(x) = \sum a_k x^k$ and $G_2(x) = \sum b_k x^k$, the product $G_1(x) \cdot G_2(x)$ is defined as

$$G_1(x) \cdot G_2(x) = \sum_{k=0}^{\infty} \Big(\sum_{i+j=k} a_k b_{n-k} \Big) x^k.$$

Let then $L, K \subseteq \Sigma^*$ be languages such that for each $v \in L \cdot K = \{uw \mid u \in L, w \in K\}$, there is a unique decomposition $v = v_1 v_2$ such that $v_1 \in L$ and $v_2 \in K$. Then $G_L(x) \cdot G_K(x) = G_{LK}(x)$ as can be readily verified.

Example 5.5. Let $L = y^*$ for some $y \in \Sigma^+$. It is readily verified that

$$G_L(x) = \sum_{k=0}^{\infty} x^{|y|k} = \tfrac{1}{1-x^{|y|}}.$$

Let then L have $C_L(n) = \Theta(n)$. By Theorem 5.4 and the discussion following it, L can be represented as a union of regular expressions of the form $z_0 y_1^* z_1 y_2^* z_2$, where the longest common prefix of y_1 and z_1 is shorter than either of the words. Consider the languages L and K defined by the regular expressions $z_0 y_1^*$ and $z_1 y_2^* z_2$, respectively. It is readily verified that

$$G_L(x) = \sum_{k=0}^{\infty} x^{k|y_1|+|z_0|} = x^{|z_0|} \sum_{k=0}^{\infty} x^{k|y_1|} = \frac{x^{|z_0|}}{1 - x^{|y_1|}}$$

and, similarly

$$G_K(x) = \frac{x^{|z_1|+|z_2|}}{1 - x^{|y_2|}}.$$

Now the language $L \cdot K$ has the property that each of its elements u has a unique factorization of form $u = v_1 v_2$, where $v_1 \in L$ and $v_2 \in K$. Indeed, this follows easily from the property of the longest common prefix of y_1 and z_1 (see the proof of the following lemma). Thus

$$G_{L \cdot K}(x) = G_L(x) \cdot G_K(x) = \frac{x^{|z_0 z_1 z_2|}}{(1 - x^{|y_1|})(1 - x^{|y_2|})}.$$

We shall now generalize the above example.

Lemma 5.6. Let L be a regular language defined by the regular expression $z_0 y_1^* z_1 \cdots y_t^* z_t$, where the longest common prefix of y_i and z_i is shorter than either of y_i and z_i for each $i = 1, \ldots, t-1$. Then

$$G_L(x) = \frac{p_L(x)}{q_L(x)} = x^z \prod_{i=1}^{t} \frac{1}{1 - x^{|y_i|}}, \text{ where } z = \sum_{i=0}^{t} |z_i|. \tag{5}$$

Proof:
We prove the claim by induction on t, the case of $t = 1$ was handled in the above example. Suppose the claim is true for all expressions with the parameter value t and consider the case of $t + 1$. Let L be the language defined by the expression $z_0 y_1^* z_1 \cdots y_t^*$ and K the language defined by the expression $z_t y_{t+1} z_{t+1}$. Thus the generating function we are looking for is $L \cdot K$. We claim that

$$G_{L\cdot K} = G_L(x) \cdot G_K(x),$$

that is, each element of $L \cdot K$ has a unique factorization into a word of L concatenated with a word of K. Suppose the contrary:

$$z_0 y_1^{i_1} z_1 \cdots y_{t+1}^{i_{t+1}} z_{t+1} = z_0 y_1^{j_1} z_1 \cdots y_{t+1}^{j_{t+1}} z_{t+1}$$

for some $i_r, j_r \in \mathbb{N}$, $r = 1, \ldots, t + 1$. Let l be the minimum index where $i_l \neq j_l$. We may assume that $i_l > j_l$, from which it follows that

$$y_l^{i_l - j_l} z_l \cdots y_{t+1}^{i_{t+1}} z_{t+1} = z_l \cdots y_{t+1}^{j_{t+1}} z_{t+1},$$

which is impossible. The claim then follows by the induction hypothesis, since

$$G_{L\cdot K} = G_L(x) \cdot G_K(x) = x^z \prod_{i=1}^{t} \frac{1}{1 - x^{|y_i|}} \cdot x^{|z_t|+|z_{t+1}|} \frac{1}{1 - x^{|y_{t+1}|}}.$$

□

Let L be a regular language with generating function $G_L(x) = \sum a_k x^k$ which can be expressed as in (5). We shall analyze the asymptotic behaviour of the coefficients a_k by performing certain manpipulations to the rational expression of the generating function.

We recall the following property of rational expressions $r(x) = \frac{p(x)}{q(x)}$, where p and q are some polynomials over $\mathbb{C}$ (or any algebraically closed field). Let $q(x)$ have d distinct roots: Let $q(x)$ have the decomposition $q(x) = \prod_{i=1}^{d} (x - \lambda_i)^{m_i}$, where $\lambda_i \neq \lambda_j$ for $i \neq j$, for some $d \geq 1$ and $m_i \geq 1$, $i = 1, \ldots, d$. Then $r(x)$ may be expressed as

$$r(x) = C_0 + \sum_{i=1}^{d} \sum_{j=1}^{m_i} \frac{C_{ij}}{(\lambda_i - x)^j}, \tag{6}$$

where C_0, C_{ij} are constants for each $i = 1, \ldots d$, $j = 1, \ldots, m_i$. This is called the *partial fraction decomposition* or the *partial fraction expansion* of $r(x)$. We may now express the generating function $G_L(x)$ as a sum of generating functions using the partial fraction decomposition:

$$G_L(x) = \sum_{k=0}^{\infty} a_k x^k = \sum_{i=1}^{d} \sum_{j=1}^{m_i} C_{ij} \sum_{k=0}^{\infty} \binom{k+j-1}{j-1} \lambda_i^{-k} x^k \tag{7}$$

In the following, we call λ a *dominating root* of a polynomial $q(x)$ if the multiplicity of λ as a root of $q(x)$ is maximal.

Lemma 5.7. Let $G_L(x)$ have an expression as in (5). Then 1 is a dominating root of $q_L(x)$ and has multiplicity t. Furthermore, for $G_L(x) = \sum a_k x^k$, we have that

$$a_k = \sum_{i=1}^{d} \tfrac{C_{it}}{(t-1)!}\lambda_i^{-k} k^{t-1} + \mathcal{O}(k^{t-2}),$$

where $C_{it} = 0$ if λ_i is not dominating.

Proof:
First of all, 1 has multiplicity t. Furthermore, each of the polynomials $1 - x^{|y_i|}$ have $|y_i|$ distinct roots. It follows that the the maximum possible multiplicity of a root of $q_L(x)$ is t, whence 1 is a dominating root.

All the roots of $q_L(x)$ are roots of unity. Therefore the values λ_i^k, $k \in \mathbb{N}$, are uniformly bounded for each i. We then note that

$$\binom{k+j-1}{j-1} = \tfrac{1}{(j-1)!}(k+1)(k+2)\cdots(k+j-1) = \tfrac{1}{(j-1)!}k^{j-1} + \mathcal{O}(k^{j-2}).$$

It follows that the coefficient a_k is of the claimed order. □

Lemma 5.8. Let $G_L(x)$ be as in (5), and let C_{1t} be the coefficient of the term $\frac{1}{(1-x)^t}$ of the partial fraction decomposition of $G_L(x)$. Then $C_{1t} = \prod_{i=1}^{t} 1/|y_i|$.

Proof:
Let $C = C_{1t}$ for short. We may express $G_L(x) = \frac{x^z}{q(x)}$ as

$$G_L(x) = \tfrac{C}{(1-x)^t} + \tfrac{P(x)}{(1-x)^{t-1}R(x)},$$

where $R(x) = \frac{q(x)}{(1-x)^t} = \prod_{i=1}^{t}(\sum_{i=0}^{|y_i|-1} x^i)$ and $P(x)$ is some polynomial. Combining the terms yields

$$G_L(x) = \frac{C}{(1-x)^t} + \frac{P(x)}{(1-x)^{t-1}R(x)} = \frac{CR(x) + (1-x)P(x)}{q(x)}$$

implying that $CR(x) + (1-x)P(x) = x^z$. Evaluating both sides at $x = 1$ we obtain $C = 1/R(1)$. The claim follows. □

Proposition 5.9. Let $G_L(x)$ be as in (5). If $\gcd(|y_1|, |y_2|, \ldots, |y_t|) = 1$ then 1 is the unique dominating root of $q_L(x)$. In particular, $G_L(x) = \sum a_k x^k$, where $a_k = \frac{C_{1t}}{(t-1)!}k^{t-1} + \mathcal{O}(k^{t-2})$, where C_{1t} is as in the above lemma.

Proof:
Let $\lambda \neq 1$ be a root of $q_L(x)$. Assume to the contrary that λ is a root of each of the polynomials $1 - x^{|y_i|}$, $i = 1, \ldots, t$. Let m be the order of λ as a root of unity (note that $m \geqslant 2$). If λ is a root of the polynomial $1 - x^r$ for some r, then necessarily $m \mid r$. Since we are assuming that λ is a root of $1 - x^{|y_i|}$ for i, we have $\gcd(|y_1|, |y_2|, \ldots, |y_t|) \geq m > 1$, a contradiction. □

Corollary 5.10. Let L be a regular language with complexity $\mathcal{C}_L(n) = \Theta(n^k)$ for some k. Let $\mathcal{A}$ be a DFA recognizing L. If each automaton $\mathcal{A}_W \in \mathcal{W}(\mathcal{A})$ has 1 as a unique dominating root in the corresponding generating function, then $\mathcal{C}_L(n) \sim Dn^k$ where

$$D = \frac{1}{k!} \sum_{W \in W(\mathcal{A})} \prod_{C \in C(W)} \frac{1}{|C|}.$$

We shall be using the above corollary in our future considerations. Before moving towards the analysis of $L_{\min}$ we give a clarifying example concerning the notions discussed above.

Example 5.11. Let us consider the automaton $\mathcal{A}$ recognizing $L_{3,2,\min}$ in Figure 4. The number of cycles in $C(W)$ for a saturated walk $W \in W(\mathcal{A})$ is 5. There are several such walks but, considering cycles and the order they appear in, there are only two. For example, the walk defined by the computation of $\mathcal{A}$ on $abaaabaababababbabbba$ is considered as an automaton $\mathcal{A}_1$ recognizing the language $abaa(a^*)b(aab)^*b(ab)^*b(abb)^*bb^*a$. Since $\mathcal{A}$ is deterministic, automata obtained from two distinct saturated walks (in the sense they don't define the same automaton) define disjoint languages. In our example, taking the union of all these languages yields the expression

$$\{\varepsilon, b, ab, bb\}aaa^*b(aab)^*\big(b(aabb)^*ab + ab(ab)^*\big)b(abb)^*bb^*\{\varepsilon, a, aa, ab\}.$$

We obtain the following generating function for the union.

$$\begin{aligned}(1+x+2x^2)x^2 \cdot \tfrac{1}{1-x} \cdot x \cdot \tfrac{1}{1-x^3}\big(x\tfrac{1}{1-x^4}x^2 + x^2\tfrac{1}{1-x^2}\big) \cdot x\tfrac{1}{1-x^3} \cdot x\tfrac{1}{1-x} \cdot (1+x+2x^2)^2 \\ = \tfrac{x^7(1+x+2x^2)^2(1+x+x^2)}{(1-x)^5(1+x)(1+x^2)(1+x+x^2)^2} = \tfrac{4}{3}\tfrac{1}{(1-x)^5} - \tfrac{12x^{10}+\mathcal{O}(x^9)}{3(1-x)^4(1+x)(1+x^2)(1+x+x^2)}.\end{aligned}$$

Considering the power series $\sum_{n=0}^{\infty} a_n x^n$ obtained, the dominating term of the coefficient a_n comes from the corresponding coefficient in the series $\frac{1}{(1-x)^5} = \frac{1}{4!}\sum_{n=0}^{\infty}(n+4)(n+3)(n+2)(n+1)x^n$. Thus $a_n = \frac{4}{3}(\frac{n^4}{4!} + \mathcal{O}(n^3)) = \frac{1}{18}n^4 + \mathcal{O}(n^3)$. This is the correct coefficient by Proposition 4.8.

5.2. The Asymptotic Complexity of $L_{\min}$

We aim to show that, for $L_{\min}$ recognized by automaton $\mathcal{A}$, the root 1 is the unique dominating root of the generating function for each $\mathcal{A}_W \in \mathcal{W}(\mathcal{A})$. Theorem 5.1 then follows from Corollary 5.10. In order to accomplish this, we need a few definitions and a series of lemmata. From this point onwards, we reserve the symbol $\mathcal{A}_{\min}$ for the minimal DFA recognizing $L_{\min}$.

Our aim is to characterize the saturated automata of $\mathcal{A}_{\min}$ in terms of certain $V(dB(k))$-saturated walks of $dB(k)$. To this end we observe the following.

Lemma 5.12. Let $u \in L_{\min}$ with $|u| \geqslant k-1$. Then the walk W_u in $dB(k)$ is cycle-deterministic.

Proof:
Suppose the contrary, W_u returns to a cycle C. There exists vertices x and y such that W_u exits C via x and re-enters C via y: let (x, a) be the edge of C and let (x, b) be the edge used by W to

exit C. Note that the path from x to y along C is traversed through twice by definition. There is also walk from x to y using different edges. In other words we obtain six indices $i_1, \ldots, i_6$ with $i_1 < i_2 \leqslant i_3 < i_4 \leqslant i_5 < i_6$ such that xa occurs at positions i_1 and i_5, xb at position i_3 and y at positions i_2, i_4, and i_6. If $a \rhd b$ (resp., $a \lhd b$) then $S_{k,u}(i_1, i_2, i_3, i_4)$ (resp., $S_{k,u}(i_3, i_4, i_5, i_6)$) gives a lexicographically smaller word, which contradicts the minimality of u. □

Definition 5.13. Let $u \in \Sigma^*$ with $|u| \geqslant k-1$ and $W_u = (x_i)_{i=0}^t$ be the corresponding walk in $dB(k)$, that is, $x_i \in \Sigma^{k-1}$ for each $i = 0, \ldots, t$. For each $i \in [0, t-1]$ we define the *extension history* $\Delta_u^i \subseteq \Sigma^{k-1} \times \Sigma$ *of* u *at position* i recursively as follows. For $i = 0$ we have, for each $x \in \Sigma^{k-1}$, $(x, a) \in \Delta_u^0$ if and only if the first occurrence of x in u is followed by a, where $a \in \Sigma$.

Let then $i \geq 1$ and suppose that Δ_u^{i-1} is defined. Now x_i occurs (by definition) at position i of u, and suppose it is followed by b. Suppose further $(x_i, a) \in \Delta_u^{i-1}$. If $b = a$ we let $\Delta_u^i = \Delta_u^{i-1}$, otherwise we set $\Delta_u^i = (\Delta_u^{i-1} \setminus (x_i, a)) \cup (x_i, b)$. In the case of $a \neq b$ we call (x_i, b) an *update* and we say that the position i *defines* the update (x_i, b).

A sequence of extension histories $(\Delta_u^i)_{i=0}^t$ is called *increasing* if it satisfies the following property: for each $x \in \Sigma^{k-1}$, if $(x, a) \in \Delta_u^i$, $(x, b) \in \Delta_u^j$, and x occurs at position l for some indices i, j, l, where $i < j < l$, then it follows that $a \trianglelefteq b$. Otherwise it is called nonincreasing.

Example 5.14. Let $u = aababba$ and $k = 2$. The extension history thus consists of two elements at each time. For $i = 0$, the first occurrence of a is followed by a, the first occurrence of b by a, so that $\Delta_u^0 = \{(a, a), (b, a)\}$. At position 1 we have a followed by b, so we get an update (a, b); $\Delta_u^1 = \{(a, b), (b, a)\}$. At position 2 we have b followed by a, so that $\Delta_u^2 = \Delta_u^1$. At position 3 we have a followed by b, whence $\Delta_u^3 = \Delta_u^1$. At position 4 we have b followed by b, so an update occurs: $\Delta_u^4 = \{(a, b), (b, b)\}$. Finally, at position 5 we have b followed by a, so that $\Delta_u^5 = \{(a, b), (b, a)\}$.

The sequence of extension histories is increasing, since, even though position 5 defines the update (b, a) where the extension of b decreases, b does not occur afterwards.

We obtain a characterization of words $u \in L_{\min}$ using extension histories.

Lemma 5.15. Let $u \in \Sigma^*$ with $|u| \geqslant k-1$ and let $W_u = (x_i)_{i=0}^t$ be the corresponding walk in $dB(k)$. Then $u \in L_{\min}$ if and only if $(\Delta_u^i)_{i=0}^t$ is increasing.

Proof:
We note that, if $(x, a) \in \Delta_u^i$ and $(x, b) \in \Delta_u^j$ for some $i < j$ and $a \neq b$, then there exist indices $i' \leqslant i < j' \leqslant j$ such that xa occurs at position i' and xb occurs at position j'. If x still occurs at position l with $l > j$, then a possibility for a switching arises. If $u \in L_{\min}$ then necessarily $a \trianglelefteq b$. If, on the other hand, $a \trianglelefteq b$ whenever this kind of a situation happens, then u avoids switchings that increase lexicographically. Thus $u \in L_{\min}$ by the proof of Lemma 4.5. □

We make a further observation of the cycles along a walk defined by $u \in L_{\min}$.

Lemma 5.16. Let $W = (x_i)_{i=0}^t$ be a walk in $dB(k)$ defined by $u \in L_{\min}$ and let $C = (y_j)_{j=0}^s$ be a cycle along W such that W traverses C at least twice, that is, W enters C at some position g and leaves C at some position h with $h \geq g + 2s$. Then, for any $p \in \mathbb{N}$, the word corresponding to the

walk $(x_i)_{i=0}^{g-1} \cdot (y_{j \pmod s})_{j=0}^{ps} \cdot (x_j)_{j=g}^{j+(h \pmod s)} \cdot (x_i)_{i=h+1}^{t}$ is in $L_{\min}$. In other words, the cycle C may be repeated arbitrarily many times, and the corresponding words are in $L_{\min}$.

Proof:
Suppose this is not the case, for some p the corresponding word u' is not in $L_{\min}$. It follows that the sequence of extension histories of u' is nonincreasing. There thus exist a word $x \in \Sigma^{k-1}$, letters $a, b \in \Sigma$, and indices i, j, l, where $a \trianglelefteq b$ and $i < j < l$, such that the update (x, b) occurs at position i, the update (x, a) at position j, and x occurs in u at position l. By removing repetitions of C no new updates or new occurrences of x are created. We thus assume that repetitions of C are added.

Now one of these indices must occur in the newly added part, as otherwise $u \notin L_{\min}$. By adding repetitions of the cycle C to the original walk W, no new updates are created. Thus these updates must have occurred in u already, and hence the occurrence of x at position l must be created. Let l be the minimal index where such an occurrence is created. Since the cycle C is traversed at least twice in W, x occurs at index $l - s$ as well. This occurrence is after the previous update, which is a contradiction with the minimality of l. □

Note that the elements of the extension histories can be seen as edges in $dB(k)$. We make the following observation.

Lemma 5.17. Let then $W_u = (x_i)_{i=0}^{t}$ be a walk in $dB(k)$ and let $j \in [1, t]$ be fixed. Consider the graph G consisting of the vertices x_i of W_u and the edges of Δ_u^{j-1}. Then, for any $i \leqslant j$, there is a unique path from x_i to x_j in G.

Proof:
We prove this by induction starting from $i = j$ for which the trivial path (x_j) is admitted. Suppose the claim is true for all $l \in [i, j]$ for some $i \in [1, j]$. Consider the vertex x_{i-1}. If $x_{i-1} = x_l$ for some $l \geqslant i$, then there is nothing to prove. Otherwise, since we are dealing with the last occurrence of x_{i-1} in the walk $(x_r)_{r=1}^{j}$, by definition we have $(x_{i-1}, a) \in \Delta_u^{i-1} \cap \Delta_u^{j-1}$ unique for some $a \in \Sigma$. Thus, there is a unique simple path from x_{i-1} to x_i in G. By the induction hypothesis, we may extend this path uniquely all the way to x_j. □

Proposition 5.18. Let $u \in L_{\min}$ and let $W_u = (x_i)_{i=0}^{t}$ be the corresponding walk in $dB(k)$. Then $|C(W)| \leq 1 + \left|\cup_{i=1}^{t-1} \Delta_u^i \setminus \Delta_u^{i-1}\right|$. Here $\cup_{i=1}^{t-1} \Delta_u^i \setminus \Delta_u^{i-1}$ is the set of distinct updates in the sequence of extension histories of u.

Proof:
First of all, we observe that between two consecutive cycles along W there must occur an update to the extension histories when W exits the first cycle to reach the second.

Suppose then that the indices j and l with $j < l$ define the same update (x, b). Note that there exists a position r, $j < r < l$, defining the update (x, a) for some $a \neq b$. By Lemma 5.15 we have $a \rhd b$. We claim that there cannot occur a cycle after position l before another update occurs. Note that the endpoint (x, b) is the vertex x_{j+1}. By Lemma 5.17, there is a unique simple path using the edges of Δ_u^l that starts from x_{j+1} and ends in x. By Lemma 5.15, $(\Delta_u^i)_i$ is increasing, so an update must occur before completing the cycle by reaching x. The claim now follows. □

We are now ready to complete the proof of Theorem 5.1.

Proof:
Let $\mathcal{A}_W$ be a sub-automaton of $\mathcal{A}_{\min}$ defined by the walk W. For any word $u \in L(\mathcal{A}_W)$ the walk W_u in $dB(k)$ contains some cycles that correspond to cycles along W (if the cycles are iterated sufficiently many times), and some of the walks contain all such cycles. The maximum number of such cycles is $m^{k-1}(m-1)+1$. Indeed, by Theorem 4.1 we have $\mathcal{C}_{L_{\min}}(n) = \Theta\left(n^{m^{k-1}(m-1)}\right)$ and by Theorem 5.4 the number of cycles in a saturated sub-automaton $\mathcal{A}_W$ equals $m^{k-1}(m-1)+1$.

Let then u be a word such that the walk W_u in $dB(k)$ has all maximum possible number of cycles along it. By Proposition 5.18, the number of distinct updates in $(\Delta_u^i)_i$, for a word u for which the walk W_u in $dB(k)$ has maximal number of cycles, is at least $m^{k-1}(m-1)$. But this value is maximal: each word $x \in \Sigma^{k-1}$ can have at most $(m-1)$ updates. This sums up to $m^{k-1}(m-1)$ possible updates.

This implies that the saturated automaton $\mathcal{A}_W$ corresponding to the walk W_u in $dB(k)$ has the cycle (or more precisely, the loop) (a^{k-1}, a) along it. This is because the edge (a^{k-1}, a) must be used at some point. By Lemma 5.18, we must then enter a cycle before updating the extension history again in order to obtain the maximal number of cycles. But the extension history tells us that the edge (a^{k-1}, a) is repeated for some number of times until an update occurs. By Lemma **??**, the cycle can be repeated arbitrarily many times, each word corresponding to a word in $L_{\min}$. This cycle has length 1 so that 1 is a dominating root of the generating function of $\mathcal{A}_W$ by Proposition 5.9. Since W was arbitrary, the claim follows by Corollary 5.10. □

6. On the Structure of Fixed Sized Equivalence Classes

The regularity of the languages $L_{\min}$ and L_{sing} raises questions for the structure of larger equivalence classes. We are thus interested in the k-abelian equivalence classes of fixed cardinality. We employ the result of Theorem 3.1 to obtain a first step in this direction. In the following, we say that $y \in [x]_k$ is *extremal* if $y \in L_{\text{ext}} = L_{\min} \cup L_{\max}$.

Theorem 6.1. The language $L_2 = \{w \in \Sigma^* \mid |[w]_k| = 2\}$ is a regular language.

Proof:
Consider the regular language $L = \Sigma^* \setminus L_{\text{ext}}$: we have

$$L = \{w \in \Sigma^* \mid |[w]_k| \geqslant 3 \text{ and } w \text{ is not extremal}\},$$

since all classes containing at most two elements are removed. We shall then use the language operation R_k defined previously. Now $L' = R_k(L) \cup L = \{w \in \Sigma^* \mid |[w]_k| \geqslant 3\}$. Note that one operation of R_k is sufficient to fill the equivalence classes: by Lemma 4.4, each word $x \in L$ admits at least two switchings: one decreasing and the other increasing in lexicographic order. By Lemma 2.2 L' is regular. Finally, the complement of L' is the language $\{w \in \Sigma^* \mid |[w]_k| \leqslant 2\}$. We thus have that $L_2 = \overline{L'} \setminus L_{\text{sing}}$ is a regular language. □

For values larger than 2, the regularity of the union of equivalence classes of a fixed size is left open. We only note that the above approach does not extend at least immediately. The example below gives some evidence that already for the value 3 the problem becomes involved.

Note that the language operation R_k can be modified, e.g., to the operation $R_{k,\neq}$ defined as $R_{k,\neq}(L) = \{u \in \Sigma^* \mid \exists v \in L : u \in R_k(\{v\}) \setminus \{v\}\}$. This operation performs switchings that actually give another word. A straightforward modification of the proof of Theorem 3.1 shows that regular languages are closed under this operation as well.

Example 6.2. We show that the language

$$K = \{x \in \Sigma^* \mid R_{k,\neq}([x]_k \setminus L_{\text{ext}}) \subseteq L_{\text{ext}}\}$$

is regular. This is the language of words x for which any $y \in [x]_k$, y not extremal, admits exactly two switchings: the other giving the least element of $[x]_k$, the other giving the maximal element of $[x]_k$. Indeed, let again $L = \Sigma^* \setminus L_{\text{ext}} = \{w \in \Sigma^* \mid w \text{ not extremal}\}$. We may then perform

$$R_{k,\neq}(L) \setminus L_{\text{ext}} = \{w \in \Sigma^* \mid \exists x \in R_{k,\neq}(\{w\}) \setminus L_{\text{ext}}\}.$$

Taking the complement of this language gives our claim.

Note that the language $\{w \in \Sigma^* \mid |[w]_k| = 3\}$ is included in K, but that there exist other classes in K also (for example $u = a^k b a^{k-1} c^k d c^{k-1}$ for which $[u]_k = 4$). Further, Remark 3.2 hints that minimal elements with respect to a given regular language (instead of Σ^*) is more difficult to characterize. On the other hand, an old result gives a positive answer to this question when $k = 0$:

Theorem 6.3. (Theorem 4.1. in [20])
For every regular language L, the language $\min(L) = \{w \in L \mid w \trianglelefteq u \text{ for every } u \in L, |u| = |w|\}$ is regular, and a regular grammar for it can be effectively constructed.

7. Open Problems and Future Research

The topic of this paper opens up new aspects of k-abelian equivalence, and presents a series of questions. Though explicit formulas for the functions $\mathcal{P}_{k,m}$ and $\mathcal{S}_{k,m}$ were obtained for small values of k and m, it remains to compute the corresponding generating functions (which, by our results, are rational functions).

To conclude, we suggest the following open problem.

- Is the language of words w having $|[w]_k| = l$, where l is a fixed constant, a regular language? For $l = 2$, this is settled in the positive by Theorem 6.1.

Acknowledgments

The automata used to calculate the functions in Proposition 4.8 and Proposition 4.9 were constructed using the java package `dk.brics.automaton` [21]. The automata in Figure 3 were created using the software Graphviz [22].

The authors are grateful to Arseny Shur for valuable discussions. We would also like to thank the anonymous referees of the conference version for valuable comments which helped to improve the presentation.

References

[1] Cassaigne J, Karhumäki J, Saarela A. On Growth and Fluctuation of k-Abelian Complexity. In: Computer Science - Theory and Applications - 10th International Computer Science Symposium in Russia, CSR 2015, Listvyanka, Russia, July 13-17, 2015, Proceedings. 2015 pp. 109–122. doi:10.1007/978-3-319-20297-6_8. URL http://dx.doi.org/10.1007/978-3-319-20297-6_8.

[2] Ehlers T, Manea F, Mercas R, Nowotka D. k-Abelian pattern matching. *Journal of Discrete Algorithms*, 2015. **34**:37–48. doi:10.1016/j.jda.2015.05.004. URL http://dx.doi.org/10.1016/j.jda.2015.05.004.

[3] Karhumäki J, Puzynina S. On k-Abelian Palindromic Rich and Poor Words. In: Developments in Language Theory - 18th International Conference, DLT 2014, Ekaterinburg, Russia, August 26-29, 2014. Proceedings. 2014 pp. 191–202. doi:10.1007/978-3-319-09698-8_17. URL http://dx.doi.org/10.1007/978-3-319-09698-8_17.

[4] Karhumäki J, Puzynina S, Saarela A. Fine and Wilf's Theorem for k-Abelian Periods. *International Journal of Foundations of Computer Science*, 2013. **24**(7):1135–1152. doi:10.1142/S0129054113400352. URL http://dx.doi.org/10.1142/S0129054113400352.

[5] Karhumäki J, Saarela A, Zamboni LQ. Variations of the Morse-Hedlund Theorem for k-Abelian Equivalence. In: Developments in Language Theory - 18th International Conference, DLT 2014, Ekaterinburg, Russia, August 26-29, 2014. Proceedings. 2014 pp. 203–214. doi:10.1007/978-3-319-09698-8_18. URL http://dx.doi.org/10.1007/978-3-319-09698-8_18.

[6] Rao M, Rosenfeld M. Avoidability of long k-abelian repetitions. *Mathematics of Computation*, Published electronically: February 18, 2016. doi:10.1090/mcom/3085. URL http://dx.doi.org/10.1090/mcom/3085.

[7] Karhumäki J. Generalized Parikh Mappings and Homomorphisms. *Information and Control*, 1980. **47**(3):155–165. doi:10.1016/S0019-9958(80)90493-3. URL http://dx.doi.org/10.1016/S0019-9958(80)90493-3.

[8] Huova M, Saarela A. Strongly k-Abelian Repetitions. In: Combinatorics on Words - 9th International Conference, WORDS 2013, Turku, Finland, September 16-20. Proceedings. 2013 pp. 161–168. doi:10.1007/978-3-642-40579-2_18. URL http://dx.doi.org/10.1007/978-3-642-40579-2_18.

[9] Karhumäki J, Saarela A, Zamboni LQ. On a generalization of Abelian equivalence and complexity of infinite words. *Journal of Combinatorial Theory, Series A*, 2013. **120**(8):2189–2206. doi:10.1016/j.jcta.2013.08.008. URL http://dx.doi.org/10.1016/j.jcta.2013.08.008.

[10] Karhumäki J, Puzynina S, Rao M, Whiteland MA. On cardinalities of k-abelian equivalence classes. *Theoretical Computer Science*, 2017. **658, Part A**:190 – 204. doi:https://doi.org/10.1016/j.tcs.2016.06.010. Formal Languages and Automata: Models, Methods and Application In honour of the 70th birthday of Antonio Restivo, URL http://www.sciencedirect.com/science/article/pii/S0304397516302468.

[11] Cassaigne J, Karhumäki J, Puzynina S, Whiteland MA. k-Abelian Equivalence and Rationality, pp. 77–88. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-662-53132-7, 2016. doi: 10.1007/978-3-662-53132-7_7. URL http://dx.doi.org/10.1007/978-3-662-53132-7_7.

[12] Lothaire M (ed.). Combinatorics on Words. Cambridge University Press, second edition, 1997. ISBN 9780511566097. Cambridge Books Online, URL http://dx.doi.org/10.1017/CBO9780511566097.

[13] Eilenberg S. Automata, Languages, and Machines, volume A. Academic Press, Inc., New York, New York, USA, 1974. ISBN 0122340019.

[14] Salomaa A, Soittola M. Automata-Theoretic Aspects of Formal Power Series. Texts and Monographs in Computer Science. Springer, 1978. ISBN 978-0-387-90282-1. doi:10.1007/978-1-4612-6264-0. URL http://dx.doi.org/10.1007/978-1-4612-6264-0.

[15] Weintraub SH. Jordan Canonical Form: Theory and Practice. Synthesis Lectures on Mathematics & Statistics. Morgan & Claypool Publishers, 2009. doi:10.2200/S00218ED1V01Y200908MAS006. URL http://dx.doi.org/10.2200/S00218ED1V01Y200908MAS006.

[16] Gawrychowski P, Krieger D, Rampersad N, Shallit J. Finding the Growth Rate of a Regular of Context-Free Language in Polynomial Time, pp. 339–358. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-540-85780-8, 2008. doi:10.1007/978-3-540-85780-8_27. URL http://dx.doi.org/10.1007/978-3-540-85780-8_27.

[17] Huova M, Karhumäki J, Saarela A, Saari K. Local Squares, Periodicity and Finite Automata. In: Rainbow of Computer Science - Dedicated to Hermann Maurer on the Occasion of His 70th Birthday. 2011 pp. 90–101. doi:10.1007/978-3-642-19391-0_7. URL http://dx.doi.org/10.1007/978-3-642-19391-0_7.

[18] Szilard A, Yu S, Zhang K, Shallit J. Characterizing regular languages with polynomial densities, pp. 494–503. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-540-47291-9, 1992. doi:10.1007/3-540-55808-X_48. URL http://dx.doi.org/10.1007/3-540-55808-X_48.

[19] Flajolet P, Sedgewick R. Analytic Combinatorics. Cambridge University Press, New York, NY, USA, 1 edition, 2009. ISBN 0521898064, 9780521898065.

[20] Andrașiu M, Păun G, Dassow J, Salomaa A. Language-theoretic problems arising from Richelieu cryptosystems. *Theoretical Computer Science*, 1993. **116**(2):339 – 357. doi:http://dx.doi.org/10.1016/0304-3975(93)90327-P. URL http://www.sciencedirect.com/science/article/pii/030439759390327P.

[21] Møller A. dk.brics.automaton – Finite-State Automata and Regular Expressions for Java, 2010. URL http://www.brics.dk/automaton/.

[22] Gansner ER, North SC. An open graph visualization system and its applications to software engineering. *SOFTWARE - PRACTICE AND EXPERIENCE*, 2000. **30**(11):1203–1233. URL http://www.graphviz.org.