SEMI-FLOWER AUTOMATA

SHUBH NARAYAN SINGH

DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI
GUWAHATI - 781039, INDIA

AUGUST 2012

Semi-Flower Automata

By

Shubh Narayan Singh

Department of Mathematics

Submitted in fulfillment of the requirements of the degree of Doctor of Philosophy

to the

Indian Institute of Technology Guwahati
Guwahati - 781039, India

August 2012

To

My Parents

Certificate

This is to certify that the thesis entitled *Semi-Flower Automata* submitted by *Mr. Shubh Narayan Singh* to the Indian Institute of Technology Guwahati, for the award of the Degree of Doctor of Philosophy, is a record of the original bona fide research work carried out by him under my supervision and guidance. The thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree or diploma.

Guwahati
August 2012

Dr. K. V. Krishna
Supervisor

Acknowledgements

This section is compulsory in any thesis, this is due purely for politeness. However, in my case, writing this section is almost as important as writing the rest of the contents. The main reason for this is that, from my point of view, a thesis is not only a collection of research results that you have obtained by working really hard along the years, but also a source that has allowed you to acquire a really big personal enrichment. Thus, I feel the necessity to thank some of the people who have contributed into making this thesis possible.

In the first place I would like to express my deepest gratitude to my Ph.D Supervisor, Dr. K. V. Krishna, for his extreme support, meticulous supervision and keeping immense patience throughout my research work. His tireless working capacity, devotion towards his work as well as his clarity of presentation have strongly motivated me. I will be always indebted to him for his unflinching encouragement and support in various ways. Thank you sir for your extraordinary support and inspiration, and for all those interesting discussions that have helped me look at things, both mathematically and personally, in a new way.

I want to convey my sincere thanks to the doctoral committee of members Dr. Anupam Saikia, Dr. Purandar Bhaduri, Dr. Vinay Wagh for reviewing my research work and giving valuable suggestions for the improvements of my research work. I sincerely acknowledge Indian Institute of Technology Guwahati for providing me with the various facilities necessary to carry out my research. I am most grateful to Ministry of Human and Resource Development, Government of India, for providing me financial assistance for the completion of my thesis work.

I would like to thank my friends with whom I spent the magic moments during the last five years at IIT Guwahati. I am also thankful to all my research scholar friends of the Department of Mathematics, IIT Guwahati for their love and company during my stay in the IIT campus. Especially, I want to thank those my friends who have kept me away from the mathematics and helped to make my life enjoyable.

I do not have enough room to adequately thank my parents for everything they have done to enable me to be as ambitious as I wanted. Thanks for being proud of me even without knowing very well in what I have been working on during these years. At the same time, I have to thank all my relatives for their support and for respecting that this is the work that I love and tolerating the long periods of being neglected because of my thesis.

Finally, I want to use this opportunity to thank the almighty God for his strange favour in my life.

IIT Guwahati *Shubh Narayan Singh*

Abstract

The thesis aims at a further study on semi-flower automata – the concept introduced by Giambruno and Restivo to study finitely generated submonoids of free monoids. The material of the thesis is an interplay between automata and algebraic structures. In fact, using semi-flower automata we study the intersection problem of submonoids of free monoids. Further, we study certain structural properties of semi-flower automata using algebraic structures.

Contributing to the intersection problem of submonoids of free monoids, in this thesis, we obtain a sufficient condition for the Hanna Neumann property of the submonoids which are generated by finite prefix sets of words. In this connection, we obtain a general rank formula for the submonoids accepted by SFA. In order to study structural properties of semi-flower automata, we choose to study holonomy decomposition and on syntactic complexity. We ascertain holonomy decomposition and syntactic complexity of certain subclasses of circular semi-flower automata. Further, we count the number of primitive and generalized primitive words in the submonoids accepted by semi-flower automata.

Contents

Introduction

The present thesis is in the area of algebraic automata theory. The thesis concentrates on using automata to study certain properties of algebraic structures and vice versa. In fact, using automata-theoretic techniques we study the intersection problem of submonoids of free monoids. Further, we study certain structural properties of automata in algebraic approach.

The study of free monoids plays an important role in the combinatorics on words and formal language theory. Automata-theoretic tools have been deeply utilized in the literature for studying free monoids. There exists a lot of literature concerning automata accepting submonoids (cf. [Berstel and Perrin, 1985; Berstel et al., 2010]). Recently, Giambruno and Restivo [2008] introduced the concept called semi-flower automata to study finitely generated submonoids of a free monoid. A *semi-flower automaton* (in short, SFA) is a trim automaton with a unique initial state that is also a unique final state such that all the cycles in the automaton pass through the initial-final state. An SFA accepts a finitely generated submonoid of the free monoid over the underlying alphabet, and vice versa. Moreover, if an SFA is deterministic, it accepts the submonoid generated by a finite prefix set. Conversely, the submonoid generated by a finite prefix set is accepted by a deterministic SFA with at most one 'branch point going in' (in short, bpi). The notion of SFA has been an important tool in studying intersection problem for finitely generated submonoids of a free monoid.

In fact, Giambruno and Restivo have obtained the Hanna Neumann property for a class of submonoids generated by finite prefix sets of words.

The present thesis aims at further investigations on semi-flower automata. In this connection, we obtain a general rank formula pertaining to an SFA. Using the rank formula, we study the Hanna Neumann property. In order to study certain structural properties of SFA, we choose to investigate holonomy decomposition and syntactic complexity of certain types of SFA. Further, we count the primitive and generalized primitive words in the submonoids accepted by SFA.

After presenting the necessary fundamentals in Chapter 1, the main work of the thesis has been organized into four chapters as described below:

Chapter 2: The Hanna Neumann property

Chapter 3: Holonomy Decomposition

Chapter 4: Syntactic Complexity

Chapter 5: L-Primitive Words

Chapter 2. Howson [1954] proved that the intersection of two finitely generated subgroups of a free group is finitely generated. In 1956, Hanna Neumann improved the result that if H and K are finite rank subgroups of a free group, then

$$\widetilde{rk}(H \cap K) \leq 2\widetilde{rk}(H)\widetilde{rk}(K),$$

where $\widetilde{rk}(N) = \max(0, rk(N) - 1)$ for a subgroup N of rank $rk(N)$. Further, Neumann conjectured that

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K), \qquad (\star)$$

which is known as Hanna Neumann conjecture [Neumann, 1956]. In 1990, Walter Neumann proposed a stronger form of the conjecture called the strengthened Hanna Neumann conjecture (SHNC) [Neumann, 1990]. Meakin and Weil [2002] proved SHNC for the class of positively generated subgroups of a free group. The conjecture has recently been settled by Mineyev (cf. Mineyev [2011, 2012]) and announced

independently by Friedman (cf. Friedman [2011a,b]).

In contrast, it is not always true that the intersection of two finitely generated submonoids of a free monoid is finitely generated. It appears that the intersection problem for submonoids of free monoids is much more complex than the analogous problem for subgroups of free groups. In particular, the Hanna Neumann property for submonoids of a free monoid is of special interest. Two finitely generated submonoids H and K of a free monoid are said to satisfy the *Hanna Neumann property* (in short, HNP), if H and K satisfy the inequality $(\star)$. There are several contributions in the literature that study the intersection of two submonoids of a free monoid.

Tilson [1972] proved that the intersection of free submonoids of the free monoid over a finite alphabet is free. In connection with the HNP, Karhumäki obtained a result for submonoids of rank two of the free monoid over a finite alphabet. In fact, Karhumäki [1984] proved that the intersection of two submonoids of rank two is generated by either a set of at most two words or a regular language of a special form. Using SFA, Giambruno and Restivo [2008] have initiated the investigations on the intersection of two submonoids. They have obtained the HNP for a special class of submonoids generated by finite prefix sets of words.

We continue the work of Giambruno and Restivo in Chapter 2. Here, we present a sufficient condition for the HNP of the entire class of submonoids generated by finite prefix sets. In this connection, we also obtain a general rank formula for the submonoids which are accepted by SFA.

Chapter 3. The primary decomposition theorem due to Krohn and Rhodes [1965] is one of the fundamental results in the theory of automata and monoids. Eilenberg's holonomy decomposition theorem for transformation monoids is a sophisticated version of the Krohn-Rhodes decomposition theorem [Eilenberg, 1976]. The holonomy

decomposition theorem ascertains that every finite transformation monoid is covered by a wreath product of holonomy permutation-reset transformation monoids. The holonomy method appears to be relatively efficient and has been implemented computationally by Egri-Nagy and Nehaniv [2010]. The holonomy decomposition of the monoid of an automaton looks for groups induced by the elements of the monoid which permute certain subsets of the state set. These groups are called holonomy groups, which are building blocks for the components of the decomposition. Egri-Nagy and Nehaniv [2005] have proved that an automaton is algebraically cycle-free if and only if its holonomy groups are trivial. As circular automaton is algebraically cycle-free, it is a nontrivial problem to estimate the holonomy groups for a circular automaton.

In the direction of understanding some structural properties of SFA, we investigate the holonomy decomposition of SFA in Chapter 3. Here, we consider the decompositions of circular semi-flower automata (CSFA) classified by the number of bpis. We prove that the decomposition of a CSFA with at most one bpi has only one component determined by a cyclic group. Further, we obtain the holonomy decomposition of CSFA with two bpis.

Chapter 4. The syntactic complexity of a recognizable language is the cardinality of its syntactic monoid. Further, the syntactic complexity of a subclass of the class of recognizable languages is the maximal syntactic complexity of languages in that subclass, taken as a function of the state complexity of these languages. The syntactic complexity of a recognizable language provides a measure for the complexity of the recognizable language. The syntactic complexity of recognizable languages has received more attention in recent years (cf. [Beaudry and Holzer, 2011; Brzozowski and Li, 2012; Brzozowski and Liu, 2012; Brzozowski et al., 2012; Brzozowski and Ye, 2011]).

Holzer and König [2004] have studied the syntactic complexity of recognizable languages, in general. For instance, they showed that the syntactic complexity of unary recognizable languages is linear. If the size of the alphabet is at least three, then they also proved that the syntactic complexity is reached to the maximal size n^n. It turns out that the most crucial case is to determine the syntactic complexity of recognizable languages over a binary alphabet. In the binary alphabet case, Holzer and König have investigated the syntactic complexity of the monoids generated by two transformations, where one is a permutation with a single cycle and the other is not a permutation.

In Chapter 4 of the thesis, we focus on the syntactic complexity of the monoids generated by two transformations in which one is a circular permutation and the other is a special type of non-permutation. In fact, we investigate the syntactic complexity of the submonoids accepted by CSFA. We pursue this with respect to the number of bpis of CSFA. Here, we show that the syntactic complexity of the submonoids accepted by CSFA with at most one bpi is linear. Further, we prove that the syntactic complexity of the submonoids accepted by CSFA with two bpis over a binary alphabet is $2n(n+1)$.

Chapter 5. A nonempty word which is not a power of any other word is called a primitive word. It is well known that every word can be uniquely expressed as a power of primitive word [Lyndon and Schützenberger, 1962]. The concept of primitive words plays an important role in the algebraic theory of languages. Ito et al. [1988] have investigated the number of primitive words in the languages accepted by automata. Shyr and Tseng [1984] have proved that any noncommutative submonoid of a free monoid contains infinitely many primitive words. In the literature, there are various types of generalizations/extensions of the classic definition of primitive words [Czeizler et al., 2010; Domaratzki, 2004; Hsiao et al., 2002; Kari and Thierrin,

1998]. Given a language L, L-primitive words is yet another generalization of primitive words introduced by Krishna [2011]. A nonempty word which is not a proper power of any word in L is an L-primitive word.

In this chapter, we focus on the number of primitive words and L-primitive words in the languages of SFA. Indeed, we count the number in the submonoids of a free monoid. We observe that the number of primitive words in the submonoids is either at most one or infinite. Further, given a finite language L, we prove that the number of L-primitive words in submonoids is also either at most one or infinite.

Epilogue. In the present thesis, we study semi-flower automata in various aspects. In all the aspects under consideration, the thesis shows a lot of scope for further studies in the present topic. Some details on the further study have been provided with a concluding section at the end of each contributory chapter.

1

Automata and Monoids

In this chapter, we present certain fundamentals on automata and monoids concerning the thesis. Other than the material presented in section 1.3, rest of the material of the chapter is very standard in the present thesis area – algebraic automata theory. However, in order to fix the notation for the thesis, we state the necessary material. In this context, we present the notions of languages, automata, and their respective monoids, in sections 1.1, 1.2, and 1.4, respectively. One may refer to [Berstel and Perrin, 1985; Lawson, 2004] for more details of the material presented in these sections. Section 1.3 is devoted to the concept of semi-flower automata introduced by Giambruno and Restivo [2008], and review their relation with submonoids of free monoids.

1.1 Free monoids

In this section, we present the notions of languages and free monoids. We review the result concerning generators of submonoids of free monoids. Accordingly, we provide the notion of the rank of submonoids of free monoids.

Definition 1.1.1. Let A be a nonempty finite set called an *alphabet* and its elements are called *letters/symbols*. A *word* over A is a finite sequence of letters written by juxtaposing them. The set of all words over A forms a monoid with respect to concatenation of words, called the *free monoid* over A and it is denoted by A^*. The identity of A^* is the empty word (the empty sequence of letters), which is denoted by ε. The set of all nonempty words over A is denoted by A^+.

In what follows, A always denotes an alphabet.

Definition 1.1.2. A *language* over A is a subset of the free monoid A^*.

Definition 1.1.3. Let w be a word over A. For $a \in A$, the number of occurrences of a in w is denoted by $|w|_a$. Further, the *length of* w, denoted by $|w|$, is defined as $|w| = \sum_{a \in A} |w|_a$. Clearly, the length of the empty word is zero.

Definition 1.1.4. Let u, w be words over A. The word u is called a *prefix* (or *suffix*) of w, if there exists a word v such that $w = uv$ (or $w = vu$, respectively). The prefix (or the suffix) u is said to be *proper prefix* (or *proper suffix*, respectively) of w, if v is nonempty. A subset X of A^* is called a *prefix set* if no word of X is a proper prefix of another word of X.

Remark 1.1.5. If X is a prefix set and $\varepsilon \in X$, then $X = \{\varepsilon\}$.

Notation 1.1.6. For a subset X of A^*, we denote by X^* the submonoid generated by X, i.e.

$$X^* = \{x_1 x_2 \cdots x_k \mid x_i \in X \text{ and } k \geq 0\}.$$

Theorem 1.1.7. *Any submonoid H of A^* has a unique minimal set of generators given by*

$$(H \setminus \{\varepsilon\}) \setminus (H \setminus \{\varepsilon\})^2.$$

Here, $X \setminus Y = \{x \in X \mid x \notin Y\}$, the set difference.

Definition 1.1.8. Let H be a submonoid of A^*. The *rank* of H, denoted by $rk(H)$, is defined as the cardinality of the minimal set of generators X of H, i.e. $rk(H) = |X|$. If X is finite, then we say that H is a *finitely generated submonoid*. Further, the *reduced rank* of H, denoted by $\widetilde{rk}(H)$, is defined as

$$\widetilde{rk}(H) = \max(0, rk(H) - 1).$$

Remark 1.1.9 ([Karhumäki, 1984])**.** The intersection of two finitely generated submonoids of A^* is not necessarily finitely generated. For instance, consider the finitely generated submonoids $H = \{aab, aba\}^*$ and $K = \{a, baaba\}^*$ over the alphabet $A = \{a, b\}$. The intersection $H \cap K = \{a(abaaba)^n baaba \mid n \geq 0\}^*$ is not finitely generated submonoid of A^*.

1.2 Automata

In this section, we present the notion of an automaton and its digraph representation. Using the digraph representation, we discuss various notions related to automata. Further, we provide certain types and constructions in automata. Though the material of this section can be found in any standard book on automata, we would particularly refer to [Berstel and Perrin, 1985].

Definition 1.2.1. An *automaton* is a quintuple $\mathcal{A} = (Q, A, I, T, \mathcal{F})$, where Q is a nonempty finite set called the set of *states*, A is an alphabet called the *input alphabet*, I and T are subsets of Q, called the set of *initial states* and the set of *final states*, respectively, and $\mathcal{F} \subseteq Q \times A \times Q$ called the set of *transitions*. Clearly, by denoting

the states as vertices/nodes and the transitions as labeled arcs, an automaton can be represented by a directed graph (digraph) in which initial and final states shall be distinguished appropriately.

Definition 1.2.2. Let $\mathcal{A} = (Q, A, I, T, \mathcal{F})$ be an automaton and $q \in Q$.

(i) The state q is said to be a *branch point going in*, in short *bpi*, if the number of transitions coming into q (i.e. the indegree of q – the number of arcs coming into q – in the digraph of $\mathcal{A}$) is at least two. The set of all bpis of $\mathcal{A}$ is denoted by $BPI(\mathcal{A})$.

(ii) The state q is said to be a *branch point going out*, in short *bpo*, if the number of transitions going out from q (i.e. the outdegree of q – the number of arcs going out from q – in the digraph of $\mathcal{A}$) is at least two. The set of all bpos of $\mathcal{A}$ is denoted by $BPO(\mathcal{A})$.

Definition 1.2.3. Let $\mathcal{A} = (Q, A, I, T, \mathcal{F})$ be an automaton. A *path* in $\mathcal{A}$ is a finite sequence of consecutive arcs in its digraph. For $p_i \in Q$ $(0 \leq i \leq k)$ and $a_j \in A$ $(1 \leq j \leq k)$, let

$$p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{k-1}} p_{k-1} \xrightarrow{a_k} p_k$$

be a path, say P, in $\mathcal{A}$ that is starting at p_0 and ending at p_k. In this case, we write $\mathfrak{s}(P) = p_0$ and $\mathfrak{e}(P) = p_k$. The word $a_1 \cdots a_k \in A^*$ is the *label* of P, and the integer k is the *length* of P. A *null path* is a path from a state to itself labeled by ε, which is of length zero.

Definition 1.2.4. Let $\mathcal{A}$ be an automaton.

(i) A path in $\mathcal{A}$ is called *simple* if all the states on the path are distinct.

(ii) A path in $\mathcal{A}$ that starts and ends at the same state is called a *cycle* in $\mathcal{A}$, if it is not a null path.

(iii) A cycle in $\mathcal{A}$ that passes through the state q is called *simple in* q if q appears for exactly once in the cycle.

(iv) A cycle in $\mathcal{A}$ with all its states are distinct is called a *simple cycle*. That is, a simple cycle is simple in all its states.

Definition 1.2.5. Let $\mathcal{A}$ be an automaton and P a simple path in $\mathcal{A}$.

(i) A *subpath* of P is a subsequence of consecutive arcs in P.

(ii) A subpath of P is called a *prefix path* of P if both start at the same state.

(ii) A subpath of P is called a *suffix path* of P if both end at the same state.

Definition 1.2.6. Let $\mathcal{A}$ be an automaton. The *language accepted/recognized by* $\mathcal{A}$, denoted by $L(\mathcal{A})$, is the set of words that are the labels of the paths from an initial state to a final state. A language is called *recognizable* if it is accepted by an automaton.

Definition 1.2.7. Let $\mathcal{A} = (Q, A, I, T, \mathcal{F})$ be an automaton and $q \in Q$.

(i) The state q is said to be *accessible* if there is a path from an initial state to q.

(ii) The state q is said to be *coaccessible* if there is a path from q to a final state.

(iii) The *trim part* of $\mathcal{A}$, denoted by $\mathcal{A}^T$, is the automaton obtained from $\mathcal{A}$ by considering only the accessible and coaccessible states, and the respective transitions between them.

(iv) If $\mathcal{A} = \mathcal{A}^T$, then the automaton $\mathcal{A}$ is called a *trim automaton*.

Remark 1.2.8. If $\mathcal{A}$ is an automaton, then $L(\mathcal{A}) = L(\mathcal{A}^T)$.

Definition 1.2.9.

(i) An automaton is said to be *complete* if there is at least one transition defined for each pair of a state and a letter.

(ii) An automaton is said to be *deterministic* if it has a unique initial state and there is at most one transition defined for for each pair of a state and a letter.

Definition 1.2.10. Let $\mathcal{A}_1 = (Q_1, A, I_1, T_1, \mathcal{F}_1)$ and $\mathcal{A}_2 = (Q_2, A, I_2, T_2, \mathcal{F}_2)$ be two automata. The *product automaton* is the automaton

$$\mathcal{A}_1 \times \mathcal{A}_2 = (Q_1 \times Q_2, A, I_1 \times I_2, T_1 \times T_2, \mathcal{F})$$

such that for all $p, q \in Q_1$, $p', q' \in Q_2$ and $a \in A$,

$$((p, p'), a, (q, q')) \in \mathcal{F} \iff (p, a, q) \in \mathcal{F}_1 \text{ and } (p', a, q') \in \mathcal{F}_2.$$

Proposition 1.2.11. *If $\mathcal{A}_1$ and $\mathcal{A}_2$ are two automata, then*

$$L(\mathcal{A}_1 \times \mathcal{A}_2) = L((\mathcal{A}_1 \times \mathcal{A}_2)^T) = L(\mathcal{A}_1) \cap L(\mathcal{A}_2).$$

1.3 Semi-flower automata

In this section, we formally present the notion of semi-flower automata introduced by Giambruno and Restivo [2008]. Further, we review certain fundamental properties of semi-flower automata from [Giambruno, 2007].

Definition 1.3.1. An automaton is called a *monoidal automaton* if it is a trim automaton with a unique initial state that is equal to a unique final state.

Notation 1.3.2. If $\mathcal{A} = (Q, A, I, T, \mathcal{F})$ is a monoidal automaton, we denote the initial-final state by q_0. In which case, we simply write $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$. Further, let us denote by $C_\mathcal{A}$ the set of cycles in $\mathcal{A}$ that are simple in q_0, and by $Y_\mathcal{A}$ the set of their labels. Moreover, the unique initial-final state of $\mathcal{A}$ is distinguished by double outerlines in the digraph of $\mathcal{A}$.

Theorem 1.3.3. *Let $\mathcal{A}$ be a monoidal automaton; then, $\mathcal{A}$ accepts the submonoid generated by $Y_\mathcal{A}$.*

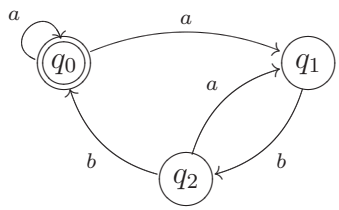

Figure 1.1: A monoidal automaton

Example 1.3.4. The automaton $\mathcal{A}$ given in Figure 1.1 is a monoidal automaton. Here, the set $Y_{\mathcal{A}} = \{a\} \cup \{a(ba)^n b \mid n \geq 0\}$ is infinite.

In the following propositions, we review some properties of monoidal automata with respect to their bpis.

Proposition 1.3.5. *If $\mathcal{A}$ is a monoidal automaton, then*

$$BPI(\mathcal{A}) = \varnothing \quad \text{if and only if} \quad BPO(\mathcal{A}) = \varnothing.$$

Proposition 1.3.6. *Let $\mathcal{A}$ be a monoidal automaton. If $BPI(\mathcal{A}) = \varnothing$, then the submonoid $L(\mathcal{A})$ is cyclic. In addition, if $\mathcal{A}$ is deterministic, then the converse is also true.*

The monoidal automata accepting finitely generated submonoids have special shape. Accordingly, for this purpose, the concept of semi-flower automata has been introduced.

Definition 1.3.7. A monoidal automaton $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ is called a *semi-flower automaton* (in short, SFA) if all the cycles in $\mathcal{A}$ visit the unique initial-final state q_0.

Remark 1.3.8. In an SFA, every cycle that is simple in q_0 is a simple cycle.

Theorem 1.3.9. *If $\mathcal{A}$ is an SFA, then the submonoid accepted by $\mathcal{A}$ is finitely generated. In fact, the generating set $Y_{\mathcal{A}}$ is finite.*

Remark 1.3.10. The set $Y_{\mathcal{A}}$ in an SFA is not necessarily a minimal set of generators of the submonoid $L(\mathcal{A})$. For instance, the automaton $\mathcal{A}$ given in FIGURE 1.2, is an SFA with $Y_{\mathcal{A}} = \{a, ba, aba\}$. Clearly, the word $aba \in Y_{\mathcal{A}}$ can be written as a concatenation of other two words.

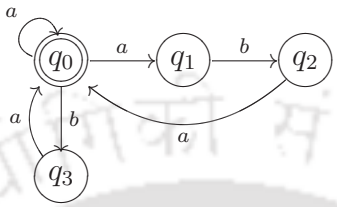

FIGURE 1.2: A semi-flower automaton

Theorem 1.3.11. *If $\mathcal{A}$ is a deterministic SFA, then $Y_{\mathcal{A}}$ is a prefix set and it is the minimal set of generators of the submonoid accepted by $\mathcal{A}$.*

Theorem 1.3.12. *Let X be a finite subset of A^* and H the submonoid generated by X. There exists an SFA with at most one bpi accepting H. Moreover, if X is a prefix set, then there exists a deterministic SFA with at most one bpi accepting H.*

In the following proposition, we present some properties of automata which are maintained by their product automaton.

Proposition 1.3.13. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two automata.*

(i) *If $\mathcal{A}_1$ and $\mathcal{A}_2$ are deterministic, then so is $\mathcal{A}_1 \times \mathcal{A}_2$.*

(ii) *If $\mathcal{A}_1$ and $\mathcal{A}_2$ are monoidal, then so is $(\mathcal{A}_1 \times \mathcal{A}_2)^T$.*

Remark 1.3.14. The product of two SFA is not necessarily an SFA. For instance, the automata $\mathcal{A}_H$ and $\mathcal{A}_K$ given in FIGURE 1.3 are deterministic SFA accepting the submonoids $H = \{aab, aba\}^*$ and $K = \{a, baaba\}^*$, respectively. Their product automaton $\mathcal{A}_H \times \mathcal{A}_K$ is given in FIGURE 1.4. Clearly, $\mathcal{A}_H \times \mathcal{A}_K$ has a cycle that is not passing through its initial-final state, so that $\mathcal{A}_H \times \mathcal{A}_K$ is not an SFA.

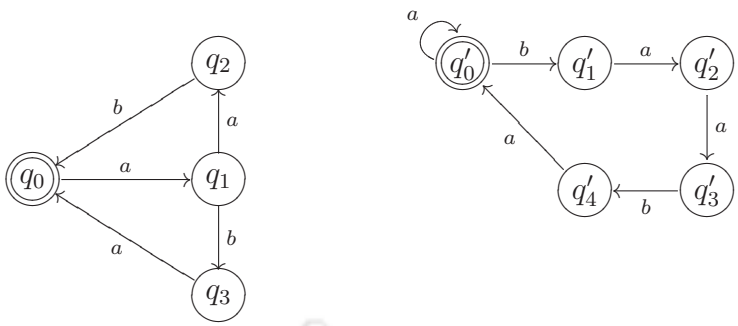

FIGURE 1.3: The deterministic SFA $\mathcal{A}_H$ (in the left) and $\mathcal{A}_K$ (in the right)

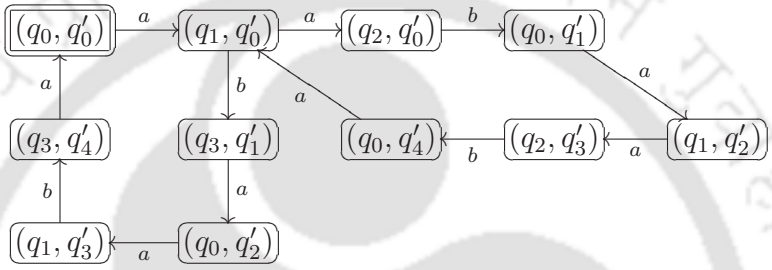

FIGURE 1.4: The automaton $(\mathcal{A}_H \times \mathcal{A}_K)^T$

1.4 Monoids of automata and languages

The fundamental notions in algebraic automata theory are monoids of automata and the syntactic monoids of languages. This section is devoted to introduce these notions and review a relation between them. For more details, one may refer to [Lawson, 2004; Pin, 1986].

We first fix the notation regarding functions. We write the argument of a function f on the left of f so that xf is the value of f at the argument x. The composition of functions is designated by concatenation, with the leftmost function understood to apply first so that $xfg = (xf)g$.

Let $\mathcal{A} = (Q, A, I, T, \mathcal{F})$ be a complete and deterministic automaton. As there is a unique transition defined over a state and a letter in $\mathcal{A}$, each $a \in A$ induces a function

$$\overline{a} : Q \longrightarrow Q$$

defined by $q\overline{a} = p$, where $(q, a, p) \in \mathcal{F}$. This phenomenon can naturally be extended to the words in A^*. For $x \in A^*$, the function induced by x, written $\overline{x} : Q \longrightarrow Q$, is defined inductively as follows. For $q \in Q$,

(i) if $x = \varepsilon$, the function $\overline{\varepsilon}$ is the identity function given by $q\overline{\varepsilon} = q$;

(ii) if $x = ya$, for some $a \in A$, then $q\overline{x} = (q\overline{y})\overline{a}$.

Definition 1.4.1. Let $\mathcal{A}$ be a complete and deterministic automaton. The set of functions $\{\overline{x} \mid x \in A^*\}$ forms a monoid under the composition of functions called the *monoid of the automaton* $\mathcal{A}$, and it is denoted by $M(\mathcal{A})$.

Remark 1.4.2. The monoid $M(\mathcal{A})$ is finite and generated by the functions induced by the letters of A.

Now, we recall two types of automata, viz. circular automata and permutation automata, which are studied in this thesis. These automata are defined relating to the features of functions induced by their input letters. We need the concept of circular permutation to define circular automata.

Definition 1.4.3. Let $X = \{p_1, \ldots, p_m\}$ be a nonempty finite set. A function, say f, on X is said to be a *circular permutation* on X if there is a cyclic ordering $p_{i_1}, \ldots, p_{i_m}$ on X such that, for $1 \leq j < m$,

$$p_{i_j} f = p_{i_{j+1}} \text{ and } p_{i_m} f = p_{i_1}.$$

Definition 1.4.4. Let $\mathcal{A}$ be a complete and deterministic automaton.

(i) The automaton $\mathcal{A}$ is called a *circular automaton* if there exists an input letter which induces a circular permutation on the state set.

(ii) The automaton $\mathcal{A}$ is called a *permutation automaton* if the function induced by each input letter is a permutation on the state set.

Remark 1.4.5.

(i) If $\mathcal{A}$ is a circular automaton, then $M(\mathcal{A})$ has a cyclic subgroup generated by the circular permutation.

(ii) $\mathcal{A}$ is permutation automaton if and only if $M(\mathcal{A})$ is a group.

Definition 1.4.6. Let $\mathcal{A}$ be a complete and deterministic automaton. $\mathcal{A}$ is said to be *minimal* if the number of states of $\mathcal{A}$ is less than or equal to the number of states of any other complete and deterministic automaton accepting $L(\mathcal{A})$.

Theorem 1.4.7. *Let $\mathcal{A}$ be a complete and deterministic automaton. $\mathcal{A}$ is minimal if and only if $\mathcal{A}$ is accessible and the equivalence relation $\sim_{\mathcal{A}}$ on the state set Q defined by*

$$p \sim_{\mathcal{A}} q \text{ if and only if } \forall x \in A^*(p\overline{x} \in T \iff q\overline{x} \in T)$$

is the diagonal relation.

Theorem 1.4.8. *A minimal automaton accepting a recognizable language is unique up to isomorphism.*

Definition 1.4.9. Let L be a language over A. The *syntactic congruence* of L, denoted by $\sim_L$, is the equivalence relation on A^* defined by

$$u \sim_L v \text{ if and only if } \forall x, y \in A^*(xuy \in L \iff xvy \in L).$$

The quotient monoid $A^*/_{\sim_L}$ is called the *syntactic monoid* of L.

Theorem 1.4.10. *Let L be a language over A. The language L is recognizable if and only if the syntactic monoid of L is finite.*

Theorem 1.4.11. *Let L be a recognizable language over A. The syntactic monoid of L is isomorphic to the monoid of the minimal automaton accepting L.*

2

Hanna Neumann Property

The intersection problem for submonoids of free monoids is much more complex than the analogous problem for subgroups of free groups. In particular, the Hanna Neumann property for submonoids of a free monoid is of special interest. Using an automata-theoretic approach, Giambruno and Restivo [2008] have obtained the Hanna Neumann property for a special class of submonoids generated by finite prefix sets of words. This chapter continues the work of Giambruno and Restivo and obtains a sufficient condition for the Hanna Neumann property for the entire class of submonoids generated by finite prefix sets of words. In this connection, a general rank formula for the submonoids which are accepted by semi-flower automata is also obtained.

2.1 State of the art

This section provides the state of the art on the Hanna Neumann property for submonoids of a free monoid.

Definition 2.1.1. Two finitely generated submonoids H and K of a free monoid are said to satisfy the *Hanna Neumann property* (in short, HNP), if H and K satisfy the inequality

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K).$$

There are several contributions in the literature to study the intersection of two submonoids of a free monoid. Tilson [1972] proved that the intersection of free submonoids of the free monoid over a finite alphabet is free. In connection to the HNP, Karhumäki obtained a result for submonoids of rank two of the free monoid over a finite alphabet. In fact, Karhumäki [1984] proved that the intersection of two submonoids of rank two is generated either by a set of at most two words or by a regular language of a special form. Using an automata-theoretic approach, Giambruno and Restivo [2008] have obtained the HNP for a special class of submonoids of a free monoid, as stated below.

Theorem 2.1.2 ([Giambruno and Restivo, 2008])**.** *Let $\mathcal{A}_H$ and $\mathcal{A}_K$ be deterministic SFA each with a unique bpi accepting submonoids H and K, respectively. If the automaton $(\mathcal{A}_H \times \mathcal{A}_K)^T$ is an SFA with at most one bpi, then*

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K).$$

Further, we observe the following theorem in the context of SFA with no bpis.

Theorem 2.1.3. *Let $\mathcal{A}_H$ and $\mathcal{A}_K$ be deterministic SFA accepting submonoids H and K, respectively. If $\mathcal{A}_H$ or $\mathcal{A}_K$ has no bpis, then*

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K).$$

Proof. We prove that the monoidal automaton $(\mathcal{A}_H \times \mathcal{A}_K)^T$ has no bpis. Consequently, by Proposition 1.3.6, the submonoid $H \cap K$ is cyclic so that the result follows. Without loss of generality, assume that $BPI(\mathcal{A}_H) = \varnothing$. By Proposition 1.3.5, we have $BPO(\mathcal{A}_H) = \varnothing$.

We claim that $BPO((\mathcal{A}_H \times \mathcal{A}_K)^T) = \varnothing$. Let $(p, q) \in BPO((\mathcal{A}_H \times \mathcal{A}_K)^T)$. Since $(\mathcal{A}_H \times \mathcal{A}_K)^T$ is deterministic, there exist two distinct input letters $a_1, a_2 \in A$ such that the transitions $\big((p, q), a_1, (p_1, q_1)\big)$ and $\big((p, q), a_2, (p_2, q_2)\big)$ are in $(\mathcal{A}_H \times \mathcal{A}_K)^T$, for some states (p_1, q_1) and (p_2, q_2). Thus, the transitions (p, a_1, p_1) and (p, a_2, p_2) are in $\mathcal{A}_H$. But, since $a_1 \neq a_2$, we have $BPO(\mathcal{A}_H) \neq \varnothing$; a contradiction. □

Now, we summarize the state of the art on the HNP of submonoids generated by finite prefix sets of words.

Let H and K be submonoids generated by finite prefix sets of words over A. In view of Theorem 1.3.12, suppose $\mathcal{A}_H$ and $\mathcal{A}_K$ are deterministic SFA over A with at most one bpi accepting H and K, respectively. Clearly, $(\mathcal{A}_H \times \mathcal{A}_K)^T$ is a deterministic monoidal automaton accepting $H \cap K$. In order to consider the case that $H \cap K$ is finitely generated, one could restrict $(\mathcal{A}_H \times \mathcal{A}_K)^T$ to be semi-flower. With this hypothesis, we discuss the HNP of H and K as follows.

Case 1. $\mathcal{A}_H$ or $\mathcal{A}_K$ has no bpis: In this case, by Theorem 2.1.3, the submonoids H and K satisfy the HNP.

Case 2. $\mathcal{A}_H$ and $\mathcal{A}_K$ have unique bpi: In this case, $(\mathcal{A}_H \times \mathcal{A}_K)^T$ can have arbitrary number of bpis. But, if $(\mathcal{A}_H \times \mathcal{A}_K)^T$ has at most one bpi, then by Theorem 2.1.2, the submonoids H and K satisfy the HNP.

In general, if $(\mathcal{A}_H \times \mathcal{A}_K)^T$ has more than one bpi, there are several examples of H and K which fail to satisfy the HNP (cf. [Giambruno, 2007; Giambruno and Restivo, 2008]). For instance, in Example 2.1.4, we give H and K which do not satisfy the HNP, where $(\mathcal{A}_H \times \mathcal{A}_K)^T$ has two bpis.

Example 2.1.4. Consider the submonoids $H = \{aa, aba, ba, bb\}^*$ and $K = \{a, bab\}^*$ of the free monoid $\{a, b\}^*$. We give the automata $\mathcal{A}_H$ and $\mathcal{A}_K$ which accept H and K, respectively, in FIGURE 2.1. Note that $\mathcal{A}_H$ and $\mathcal{A}_K$ are deterministic semi-flower automata, each with unique bpi. The (trim part of) product automaton $\mathcal{A}_H \times \mathcal{A}_K$ is shown in FIGURE 2.2. Clearly, $\mathcal{A}_H \times \mathcal{A}_K$ is semi-flower with two bpis, viz. (q_0, q_0') and (q_0, q_2'). One can observe that the $rk(H \cap K) = 5$; whereas, $rk(H) = 4$ and $rk(K) = 2$. Thus, H and K do not satisfy the HNP, i.e.

$$\widetilde{rk}(H \cap K) > \widetilde{rk}(H)\widetilde{rk}(K).$$

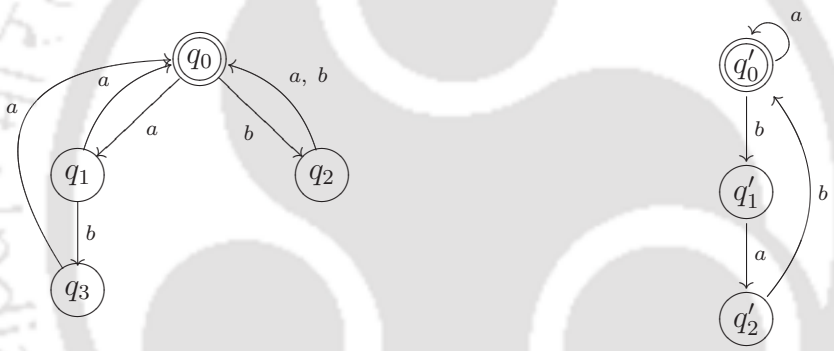

FIGURE 2.1: $\mathcal{A}_H$ (in the left) and $\mathcal{A}_K$ (in the right) of Example 2.1.4

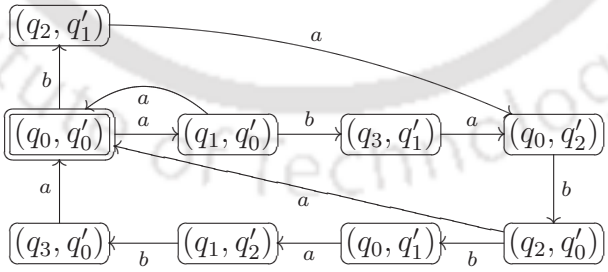

FIGURE 2.2: $\mathcal{A}_H \times \mathcal{A}_K$ of Example 2.1.4

Thus, if $(\mathcal{A}_H \times \mathcal{A}_K)^T$ has an arbitrary number of bpis, in this work, we would investigate certain conditions so that H and K satisfy the HNP. For that purpose,

we would require the following supplementary results from [Giambruno and Restivo, 2008]. For $i \geq 0$, we write

$$BPO_i(\mathcal{A}) = \{q \in Q \mid \text{ the number of transitions defined on } q \text{ is equal to } i\},$$

i.e. the set of states whose outdegree – the number of arcs going out of the state – in the digraph of $\mathcal{A}$ is i.

Proposition 2.1.5. *Let A be an alphabet of cardinality n. If $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ is a deterministic SFA, then*

$$|\mathcal{F}| - |Q| = \sum_{i=2}^{n} |BPO_i(\mathcal{A})|(i-1).$$

Proposition 2.1.6. *Let A be an alphabet of cardinality n and let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two deterministic automata over A. If $c_i = |BPO_i(\mathcal{A}_1)|$ and $d_i = |BPO_i(\mathcal{A}_2)|$, for each $i \in \{1, \ldots, n\}$, then*

$$|BPO_t(\mathcal{A}_1 \times \mathcal{A}_2)| \leq \sum_{t \leq r,s \leq n} c_r d_s.$$

Proposition 2.1.7. *Let $\langle c_1, \ldots, c_n \rangle$ and $\langle d_1, \ldots, d_n \rangle$ be two finite sequences of natural numbers; then*

$$\sum_{t=2}^{n}(t-1)\left(\sum_{t \leq r \leq n} c_r \sum_{t \leq s \leq n} d_s\right) \leq \left(\sum_{i=2}^{n}(i-1)c_i\right)\left(\sum_{j=2}^{n}(j-1)d_j\right).$$

2.2 BPR of semi-flower automata

In this section, we introduce a concise notation for an SFA in which only the initial-final state, bpis and the respective paths between them will be considered along with their labels. We call this as *bpi's and paths representation*, in short BPR, of the semi-flower automaton.

Definition 2.2.1. Let $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ be an SFA; the BPR of $\mathcal{A}$ is a quintuple $\mathcal{A}' = (Q', A, q_0, q_0, \mathcal{F}')$, where

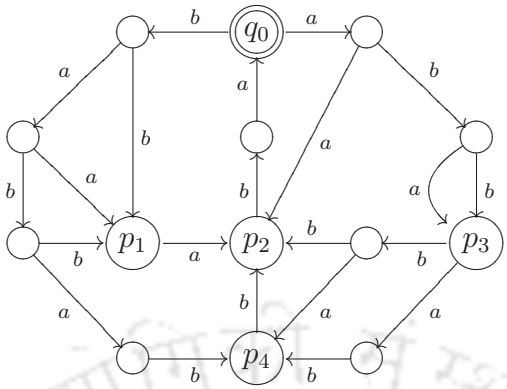

FIGURE 2.3: A semi-flower automaton

(i) $Q' = BPI(\mathcal{A}) \cup \{q_0\}$, and

(ii) $\mathcal{F}'$ is the finite subset of $Q' \times A^* \times Q'$ defined by $(p_0 = p, x, q = p_k) \in \mathcal{F}'$ if and only if there exist distinct $p_1, \ldots, p_{k-1} \in Q \setminus Q'$ and $x = a_1 \cdots a_k$, for $a_i \in A$, such that $(p_{i-1}, a_i, p_i) \in \mathcal{F}$ for all $1 \leq i \leq k$, i.e.

$$p = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{k-1}} p_{k-1} \xrightarrow{a_k} p_k = q$$

is a simple path from p to q (or simple cycle, when $p = q$) labeled by x in which the intermediate nodes, if any, are outside Q'.

Remark 2.2.2. By adopting the digraph representation of an automaton, we can draw a digraph for the BPR of an SFA. Here, the arcs are labeled by the labels (words) of respective simple paths (or simple cycles) of the SFA.

Example 2.2.3. The BPR of the SFA given in FIGURE 2.3 is shown in FIGURE 2.4.

Remark 2.2.4. Let $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ be an SFA and let $\mathcal{A}' = (Q', A, q_0, q_0, \mathcal{F}')$ be the BPR of $\mathcal{A}$.

(i) Every cycle in $\mathcal{A}'$ passes through the state q_0.

(ii) The number of simple cycles in $\mathcal{A}$ is equal to the number of simple cycles in $\mathcal{A}'$.

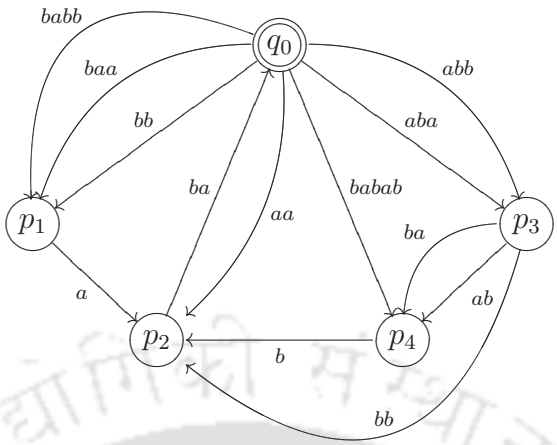

FIGURE 2.4: The BPR of the SFA given in FIGURE 2.3

(iii) For any $p, q \in Q'$, the number of simple paths from p to q in $\mathcal{A}$ is equal to the number of simple paths from p to q in $\mathcal{A}'$.

Proposition 2.2.5. *Let $\mathcal{A}$ be an SFA and let $\mathcal{A}'$ be the BPR of $\mathcal{A}$. There is a linear ordering $\preccurlyeq$ on the states of $\mathcal{A}'$ such that*

(i) *the initial-final state q_0 is the least element, and*

(ii) *for $j \neq 0$, if $q_j \preccurlyeq q_i$, then there is no arc from q_j to q_i in $\mathcal{A}'$.*

Proof. Construct the digraph $\mathcal{G}$ from the digraph of $\mathcal{A}'$ by removing the arcs which are leaving out of the initial-final state q_0. By Remark 2.2.4(i), $\mathcal{G}$ is a directed acyclic graph. Define a relation $\leq$ on the nodes of $\mathcal{G}$ by

$$p \leq q \text{ if and only if there is a simple path from } q \text{ to } p \text{ in } \mathcal{G}.$$

As the null path is a simple path from a node to itself in $\mathcal{G}$, clearly $\leq$ is reflexive. Since there are no cycles in $\mathcal{G}$, it can be observed that $\leq$ is anti-symmetric and transitive. Thus, the relation $\leq$ is a partial ordering on the nodes of $\mathcal{G}$. Since every state in $\mathcal{A}$ is coaccessible, the initial-final state q_0 is the least element with respect to $\leq$.

As every partial ordering can be extended to a linear ordering, consider a linear ordering $\preccurlyeq$ of the nodes of $\mathcal{G}$ which is an extension of $\leq$. Thus, the obtained linear ordering $\preccurlyeq$ is a desired one. □

Remark 2.2.6. By applying a topological sort algorithm (e.g. refer [Cormen et al., 2001]) on the directed acyclic graph $\mathcal{G}$, one can get a linear ordering as described in Proposition 2.2.5.

In what follows, by a *topological ordering* of the bpis of an SFA is meant a linear ordering on the states (possibly, except the initial-final state) of its BPR as in Proposition 2.2.5.

Example 2.2.7. Let $\mathcal{A}$ be the SFA given in FIGURE 2.3. A topological ordering of the bpis of $\mathcal{A}$ is

$$p_2, p_4, p_3, p_1.$$

Notice that there is no arc from p_3 or p_4 to p_1 in the BPR of $\mathcal{A}$ (cf. FIGURE 2.4). Hence, in a topological ordering of the bpis of $\mathcal{A}$, the bpi p_1 can come at any position after p_2. Thus, the possible other topological orderings are p_2, p_1, p_4, p_3 and p_2, p_4, p_1, p_3.

Notation 2.2.8. Let $\mathcal{A}$ be an SFA and let $\mathcal{A}'$ be the BPR of $\mathcal{A}$. In this chapter, if we say that $\mathcal{A}$ has m bpis, we always assume that $q_1, q_2, \ldots, q_m$ are the bpis of $\mathcal{A}$, which are considered in a topological ordering, i.e.

$$q_1 \preccurlyeq q_2 \preccurlyeq \cdots \preccurlyeq q_m.$$

As per the ordering, we also fix the following numbers in the BPR $\mathcal{A}'$.

(i) For $1 \leq i \leq m$, κ_i always refer to the number of arcs from the state q_0 to the bpi q_i in $\mathcal{A}'$.

(ii) For $1 \leq i, j \leq m$, κ_{ij} always refer to the number of arcs from the bpi q_i to the bpi q_j in $\mathcal{A}'$.

Remark 2.2.9. As per the topological ordering of the bpis of $\mathcal{A}$, we have $\kappa_{ij} = 0$ for all $j \geq i > 1$. If the initial-final state q_0 of $\mathcal{A}$ is a bpi, then clearly $q_1 = q_0$ so that, for $j \geq 1$, $\kappa_{1j} = \kappa_j$; otherwise, $\kappa_{1j} = 0$.

Remark 2.2.10. In the digraph of $\mathcal{A}$ (as well as in $\mathcal{A}'$), the indegree of a bpi q_j, $1 \leq j \leq m$, is given by the expression

$$\kappa_j + \sum_{i=j+1}^{m} \kappa_{ij}.$$

2.3 Rank of submonoids

In this section, we obtain the rank of the submonoid of a free monoid that is accepted by an SFA. The following lemma is useful in obtaining the rank of an SFA.

Lemma 2.3.1. *Let $\mathcal{A}$ be an SFA and let p be the first bpi in a topological ordering of the bpis of $\mathcal{A}$, i.e. $p \preccurlyeq q$, for all bpis q; then*

(i) *there is a unique simple path from p to the initial-final state q_0, and*

(ii) *every cycle in $\mathcal{A}$ visits p.*

Proof. If the initial-final state q_0 is a bpi, then clearly $p = q_0$. In which case, the null path is the unique simple path from p to q_0. And, since $\mathcal{A}$ is semi-flower, every cycle in $\mathcal{A}$ visits p. If q_0 is not a bpi, we proceed as follows.

(i) Since p is coaccessible, there is a path from p to the state q_0. Now suppose there are two different paths P_1 and P_2 with labels u and v, respectively, from p to the state q_0. Let w be the label of longest suffix path P' which is in common between the paths P_1 and P_2. As the state q_0 is not a bpi, $w \neq \varepsilon$. But then $\mathfrak{s}(P')$ will be a bpi different from p. This is a contradiction to the choice of p. Thus, there is a unique simple path from p to q_0.

(ii) Suppose there is a cycle that is not visiting p. Also, from above (i), there is a simple path from p to the state q_0. Let P be the longest common suffix path of these two paths ending at q_0. Clearly, $\mathfrak{s}(P)$ is a bpi and $\mathfrak{s}(P) \neq p$. Since there is a path from p to $\mathfrak{s}(P)$, we have $\mathfrak{s}(P) \preccurlyeq p$. This is a contradiction to the choice of p. Hence, every cycle in $\mathcal{A}$ visits p.

□

Corollary 2.3.2. *Let $\mathcal{A}$ be an SFA. If p is the first bpi in a topological ordering of the bpis of $\mathcal{A}$, then p is the first bpi in any topological ordering of the bpis of $\mathcal{A}$.*

Let $\mathcal{A}$ be an SFA. Now we are ready to present the result on the rank of the submonoid $L(\mathcal{A})$. The rank of $L(\mathcal{A})$ can be characterized using the bpis of $\mathcal{A}$. Note that, if there is no bpi in $\mathcal{A}$, then clearly the rank of $L(\mathcal{A})$ is either 0 or 1. If $\mathcal{A}$ has at least one bpi, we have the following theorem.

Theorem 2.3.3. *Let $\mathcal{A}$ be an SFA and $m \geq 1$. If $\mathcal{A}$ has m bpis, then*

$$rk(L(\mathcal{A})) \leq \sum_{i=1}^{m} \kappa_i \overline{\kappa_{i0}}, \qquad (\#)$$

where $\overline{\kappa_{i0}}$ is the number of simple paths from the bpi q_i to the initial-final state q_0. The number $\overline{\kappa_{i0}}$ can be given by the recursive formula

$$\overline{\kappa_{10}} = 1 \text{ and } \overline{\kappa_{i0}} = \sum_{j=1}^{i-1} \kappa_{ij} \overline{\kappa_{j0}}, \text{ for } i > 1.$$

Moreover, if $\mathcal{A}$ is deterministic, then the equality holds in $(\#)$.

Proof. Let $q_1 \preccurlyeq q_2 \preccurlyeq \cdots \preccurlyeq q_m$ be the bpis of $\mathcal{A}$. It is known from Theorem 1.3.9 that

$$rk(L(\mathcal{A})) \leq |Y_{\mathcal{A}}| \leq |C_{\mathcal{A}}|.$$

We prove the result by showing that $|C_{\mathcal{A}}|$, the number of simple cycles in $\mathcal{A}$ passing through the state q_0, is equal to the righthand side of (#), i.e. we show that

$$|C_{\mathcal{A}}| = \sum_{i=1}^{m} \kappa_i \overline{\kappa_{i0}}.$$

By Remark 2.2.4(ii), $|C_{\mathcal{A}}| = |C_{\mathcal{A}'}|$, where $C_{\mathcal{A}'}$ is the number of simple cycles in the BPR $\mathcal{A}'$ of $\mathcal{A}$.

For $1 \leq i \leq m$, let ν_i be the number of simple cycles in $\mathcal{A}'$ that are passing through the bpi q_i but not through any bpi q_j with $j > i$. Clearly,

$$|C_{\mathcal{A}'}| = \sum_{i=1}^{m} \nu_i.$$

We conclude the result by arguing that $\nu_i = \kappa_i \overline{\kappa_{i0}}$, for $1 \leq i \leq m$.

In case $i = 1$, ν_1 is the number of simple cycles in $\mathcal{A}'$ that are passing through the bpi q_1 but not through any other bpi. First note that, by Lemma 2.3.1 there is a unique simple path from q_1 to q_0 so that $\overline{\kappa_{10}} = 1$. Now, each simple cycle that is counted in ν_1 is merely an arc from q_0 to q_1 followed by the unique simple path from q_1 to q_0. Thus, the number of simple cycles counted in ν_1 is the number of arcs from q_0 to q_1, i.e. κ_1. Hence, we have

$$\nu_1 = \kappa_1 = \kappa_1 \overline{\kappa_{10}}.$$

For $i > 1$, as per the topological ordering, ν_i is clearly obtained by multiplying the number of arcs from q_0 to the bpi q_i and the number of simple paths from q_i to q_0. That is,

$$\nu_i = \kappa_i \overline{\kappa_{i0}}$$

as desired. Now, we obtain the recursive formula for $\overline{\kappa_{i0}}$. For $1 \leq t < i$, let μ_{it} be the number of simple paths in $\mathcal{A}'$ from the bpi q_i to q_0 that are passing through the bpi q_t but not through any other bpi q_j with $j > t$. Clearly, $\overline{\kappa_{i0}} = \sum_{t=1}^{i-1} \mu_{it}$. But, for $1 \leq t < i$, the number μ_{it} is nothing else but the product of the number of arcs from

q_i to q_t and the number of simple paths from q_t to q_0, i.e. $\mu_{it} = \kappa_{it}\overline{\kappa_{t0}}$. Hence, we have the recursive formula

$$\overline{\kappa_{i0}} = \sum_{t=1}^{i-1} \kappa_{it}\overline{\kappa_{t0}}.$$

If $\mathcal{A}$ is deterministic, then by Theorem 1.3.11, we have

$$rk(L(\mathcal{A})) = |C_{\mathcal{A}}| = \sum_{i=1}^{m} \kappa_i\overline{\kappa_{i0}}.$$

□

Now, Theorem 2.10 of [Giambruno and Restivo, 2008] is an immediate corollary as stated below. We will use this corollary in one of our main results.

Corollary 2.3.4. *If $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ is an SFA with a unique bpi, then*

$$rk(L(\mathcal{A})) \leq \kappa_1 = |\mathcal{F}| - |Q| + 1.$$

Moreover, if $\mathcal{A}$ is deterministic, then the equality holds.

Example 2.3.5. Let us consider the topological ordering

$$p_2 \preccurlyeq p_4 \preccurlyeq p_3 \preccurlyeq p_1$$

of the bpis of the SFA $\mathcal{A}$ given in FIGURE 2.3. That is, $q_1 = p_2$, $q_2 = p_4$, $q_3 = p_3$ and $q_4 = p_1$. Accordingly, $\kappa_1 = 1, \kappa_2 = 1, \kappa_3 = 2$ and $\kappa_4 = 3$. Also, $\kappa_{41} = 1, \kappa_{42} = 0, \kappa_{43} = 0$, $\kappa_{31} = 1, \kappa_{32} = 2$ and $\kappa_{21} = 1$. Since $\mathcal{A}$ is deterministic, we have

$$\begin{aligned} rk(L(\mathcal{A})) &= \kappa_1 + \kappa_2(\kappa_{21}) + \kappa_3(\kappa_{31} + \kappa_{32}\kappa_{21}) + \\ &\quad \kappa_4(\kappa_{41} + \kappa_{42}\kappa_{21} + \kappa_{43}\kappa_{31} + \kappa_{43}\kappa_{32}\kappa_{21}) \\ &= 11. \end{aligned}$$

2.4 A sufficient condition

In this section, we obtain a sufficient condition for the HNP of two submonoids which are accepted by deterministic SFA with a unique bpi. The following lemma is useful in obtaining the proposed result.

Lemma 2.4.1. *Let $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ be an SFA and $m \geq 1$. If $\mathcal{A}$ has m bpis, then*

$$|\mathcal{F}| - |Q| + 1 \geq rk(L(\mathcal{A})) - \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij}(\kappa_i \overline{\kappa_{j0}} - 1) \right).$$

Moreover, if $\mathcal{A}$ is deterministic, then the equality holds.

Proof. Since the number of transitions $|\mathcal{F}|$ of $\mathcal{A}$ is the total indegree (i.e. the sum of indegrees of all the states) of the digraph of $\mathcal{A}$, by Remark 2.2.10, we have

$$|\mathcal{F}| = |Q| - m + \sum_{j=1}^{m} \left(\kappa_j + \sum_{i=j+1}^{m} \kappa_{ij} \right).$$

Consequently,

$$\begin{aligned} |\mathcal{F}| - |Q| + 1 &= \kappa_1 + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} \\ &= \kappa_1 + \sum_{i=2}^{m} \kappa_i \overline{\kappa_{i0}} + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \overline{\kappa_{i0}}. \end{aligned}$$

Now, by Theorem 2.3.3 and simple algebraic manipulations, we have

$$\begin{aligned} &|\mathcal{F}| - |Q| + 1 \\ &\geq rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \overline{\kappa_{i0}} \\ &= rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \left(\sum_{j=1}^{i-1} \kappa_{ij} \overline{\kappa_{j0}} \right) \\ &= rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \left(\kappa_{i1} + \sum_{j=2}^{i-1} \kappa_{ij} \overline{\kappa_{j0}} \right) \\ &= rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{j=1}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \kappa_{i1} - \sum_{i=2}^{m} \sum_{j=2}^{i-1} \kappa_i \kappa_{ij} \overline{\kappa_{j0}} \\ &= rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) + \sum_{i=2}^{m} \kappa_{i1} + \sum_{j=2}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \kappa_i \kappa_{i1} - \sum_{i=2}^{m} \sum_{j=2}^{i-1} \kappa_i \kappa_{ij} \overline{\kappa_{j0}} \\ &= rk(L(\mathcal{A})) + \sum_{j=2}^{m} (\kappa_j - 1) - \sum_{i=2}^{m} \kappa_{i1}(\kappa_i - 1) + \sum_{j=2}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \sum_{j=2}^{i-1} \kappa_i \kappa_{ij} \overline{\kappa_{j0}} \end{aligned}$$

$$
\begin{aligned}
&= rk(L(\mathcal{A})) - \sum_{i=2}^{m} (\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{m} \sum_{i=j+1}^{m} \kappa_{ij} - \sum_{i=2}^{m} \sum_{j=2}^{i-1} \kappa_i \kappa_{ij} \overline{\kappa_{j0}} \\
&= rk(L(\mathcal{A})) - \sum_{i=2}^{m} (\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{m} \sum_{i=2}^{m} \kappa_{ij} - \sum_{i=2}^{m} \sum_{j=2}^{m} \kappa_i \kappa_{ij} \overline{\kappa_{j0}}, \\
&\quad \text{as } \kappa_{ij} = 0 \text{ for all } j \geq i > 1 \\
&= rk(L(\mathcal{A})) - \sum_{i=2}^{m} (\kappa_i - 1)(\kappa_{i1} - 1) - \sum_{i=2}^{m} \sum_{j=2}^{m} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \\
&= rk(L(\mathcal{A})) - \sum_{i=2}^{m} (\kappa_i - 1)(\kappa_{i1} - 1) - \sum_{i=2}^{m} \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \\
&\quad \text{as } \kappa_{ij} = 0 \text{ for all } j \geq i > 1.
\end{aligned}
$$

Thus,

$$
|\mathcal{F}| - |Q| + 1 \geq rk(L(\mathcal{A})) - \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right).
$$

□

Now, by Proposition 2.1.5, we have the following corollary.

Corollary 2.4.2. *Let A be an alphabet of cardinality n and let $\mathcal{A}$ be a deterministic SFA over A. For $m \geq 1$, if $\mathcal{A}$ has m bpis, then*

$$
\begin{aligned}
rk(L(\mathcal{A})) &= \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) \\
&\quad + \sum_{t=2}^{n} |BPO_t(\mathcal{A})|(t-1) + 1.
\end{aligned}
$$

Theorem 2.4.3. *Let $\mathcal{A}_H$ and $\mathcal{A}_K$ be deterministic SFA over A each with a unique bpi accepting submonoids H and K, respectively. For $m \geq 1$, if the automaton $(\mathcal{A}_H \times \mathcal{A}_K)^T$ is an SFA with m bpis, say $q_1, q_2, \ldots, q_m$ considered in a topological ordering, then*

$$
\widetilde{rk}(H \cap K) \leq \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) + \widetilde{rk}(H) \widetilde{rk}(K),
$$

where κ_i is the number of arcs from the initial-final state to q_i and κ_{ij} is the number of arcs from q_i to q_j in the BPR of $(\mathcal{A}_H \times \mathcal{A}_K)^T$.

Proof. Let A be an alphabet of cardinality n. For $m \geq 1$, note that

$$\begin{aligned}
\widetilde{rk}(H \cap K) &= rk(L(\mathcal{A}_H \times \mathcal{A}_K)) - 1 \\
&= \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) \\
&\quad + \sum_{t=2}^{n} |BPO_t(\mathcal{A}_H \times \mathcal{A}_K)|(t-1) \text{ by Corollary 2.4.2} \\
&\leq \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) \\
&\quad + \sum_{t=2}^{n} (t-1) \left(\sum_{t \leq r,s \leq n} c_r d_s \right) \text{ by Proposition 2.1.6,}
\end{aligned}$$

where $c_r = |BPO_r(\mathcal{A}_H)|$ and $d_s = |BPO_s(\mathcal{A}_K)|$. Further, by Proposition 2.1.7, we have

$$\begin{aligned}
\widetilde{rk}(H \cap K) &\leq \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) \\
&\quad + \left(\sum_{i=2}^{n} (i-1) c_i \right) \left(\sum_{j=2}^{n} (j-1) d_j \right) \\
&= \sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) + \widetilde{rk}(H)\widetilde{rk}(K) \\
&\quad \text{by Corollary 2.3.4 and Proposition 2.1.5.}
\end{aligned}$$

Hence the result. □

We now state a sufficient condition for the Hanna Neumann property of the submonoids under consideration.

Corollary 2.4.4. *In addition to the hypothesis of Theorem 2.4.3, if there is no path between any two bpis q_i and q_j, for $i, j > 1$, except those are passing through q_1 and there is a unique simple path from each bpi to q_1 in the automaton $(\mathcal{A}_H \times \mathcal{A}_K)^T$, then*

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K).$$

Proof. For $i, j > 1$, if there is no path between the bpis q_i and q_j, except those are passing through q_1, then $\kappa_{ij} = 0$. Further, for $i \geq 2$, if there is a unique simple path from each bpi q_i to q_1, then the path cannot pass through any other bpi. Thus, we have $\kappa_{i1} = 1$ so that

$$\sum_{i=2}^{m} \left((\kappa_i - 1)(\kappa_{i1} - 1) + \sum_{j=2}^{i-1} \kappa_{ij} (\kappa_i \overline{\kappa_{j0}} - 1) \right) = 0.$$

Hence, by Theorem 2.4.3,

$$\widetilde{rk}(H \cap K) \leq \widetilde{rk}(H)\widetilde{rk}(K).$$

□

2.5 Some Examples

In this section, we first illustrate the sufficient condition given in Corollary 2.4.4 with Example 2.5.1. Further, through Example 2.5.2, we observe that the condition is not necessary for the HNP. At last, in Example 2.5.3, we also provide some nondeterministic SFA whose languages (submonoids) satisfy HNP.

Example 2.5.1. Consider the deterministic SFA $\mathcal{A}_{H_1}$ and $\mathcal{A}_{K_1}$, each with unique bpi, given in FIGURE 2.5. Note that $\mathcal{A}_{H_1}$ and $\mathcal{A}_{K_1}$, respectively, accept the submonoids $H_1 = \{ab, bc, acc, baa\}^*$ and $K_1 = \{c, ac, abb, bca, bcb\}^*$ over the free monoid $\{a, b, c\}^*$. Clearly, $rk(H_1) = 4$ and $rk(K_1) = 5$. The (trim part of) product automaton $(\mathcal{A}_{H_1} \times \mathcal{A}_{K_1})^T$ is shown in FIGURE 2.6. Clearly, $(\mathcal{A}_{H_1} \times \mathcal{A}_{K_1})^T$ is a deterministic SFA with three bpi's, viz. $(p_0, p_0'), (p_2, p_0')$ and (p_3, p_0'). Since the initial-final state (p_0, p_0') of $(\mathcal{A}_{H_1} \times \mathcal{A}_{K_1})^T$ is a bpi, it is the first bpi in any topological ordering of the bpi's of $(\mathcal{A}_{H_1} \times \mathcal{A}_{K_1})^T$. Observe that there is a unique simple path from each of the other two bpi's to (p_0, p_0') and there is no path between the bpi's (p_2, p_0') and (p_3, p_0'), except those are passing through (p_0, p_0'). Further, note that $rk(H_1 \cap K_1) = 4$ and

the submonoids H_1 and K_1 satisfy HNP, i.e.

$$\widetilde{rk}(H_1 \cap K_1) \leq \widetilde{rk}(H_1)\widetilde{rk}(K_1).$$

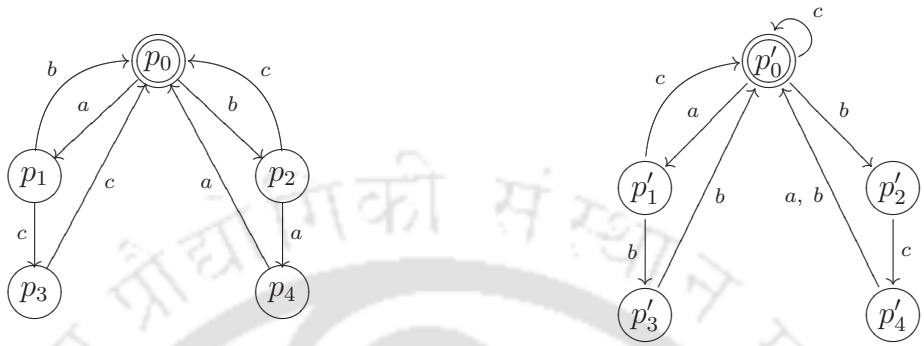

Figure 2.5: $\mathcal{A}_{H_1}$ (in the left) and $\mathcal{A}_{K_1}$ (in the right) of Example 2.5.1

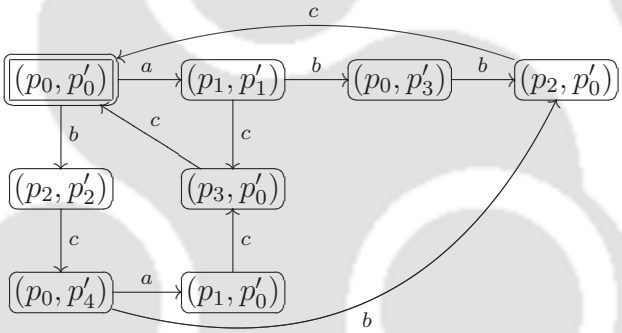

Figure 2.6: $(\mathcal{A}_{H_1} \times \mathcal{A}_{K_1})^T$ of Example 2.5.1

Example 2.5.2. Consider the deterministic SFA $\mathcal{A}_{H_2}$ and $\mathcal{A}_{K_2}$, each with unique bpi, given in Figure 2.7. Note that $\mathcal{A}_{H_2}$ and $\mathcal{A}_{K_2}$, respectively, accept the submonoids $H_2 = \{a, bb, bab\}^*$ and $K_2 = \{b, ab, aaa\}^*$ over the free monoid $\{a, b\}^*$. Clearly, $rk(H_2) = 3 = rk(K_2)$. The (trim part of) product automaton $(\mathcal{A}_{H_2} \times \mathcal{A}_{K_2})^T$ is shown in Figure 2.8. Clearly, $(\mathcal{A}_{H_2} \times \mathcal{A}_{K_2})^T$ is a deterministic SFA with two bpi's, viz. (p_0, p'_0) and (p_1, p'_0). Note that $rk(H_2 \cap K_2) = 5$ so that H_2 and K_2 satisfy HNP. However, observe that there are two paths from (p_1, p'_0) to (p_0, p'_0). Hence, the condition given in Corollary 2.4.4 is not necessary for the HNP of two submonoids of a free monoid.

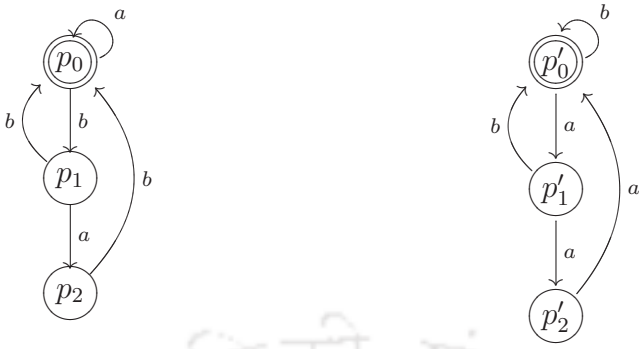

FIGURE 2.7: $\mathcal{A}_{H_2}$ (in the left) and $\mathcal{A}_{K_2}$ (in the right) of Example 2.5.2

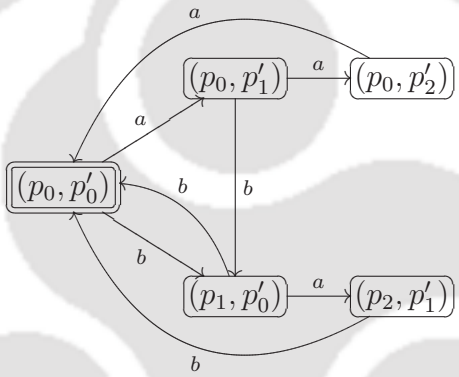

FIGURE 2.8: $(\mathcal{A}_{H_2} \times \mathcal{A}_{K_2})^T$ of Example 2.5.2

Example 2.5.3. Consider the nondeterministic SFA $\mathcal{A}_{H_3}$ and $\mathcal{A}_{K_3}$, each with unique bpi, given in FIGURE 2.9. Note that $\mathcal{A}_{H_3}$ and $\mathcal{A}_{K_3}$, respectively, accept the submonoids $H_3 = \{ab, aba, ba\}^*$ and $K_3 = \{a, bb, bab, bbb\}^*$ over the free monoid $\{a, b\}^*$. Clearly, $rk(H_3) = 3$ and $rk(K_3) = 4$. The (trim part of) product automaton $(\mathcal{A}_{H_3} \times \mathcal{A}_{K_3})^T$ is shown in FIGURE 2.10. Clearly, $(\mathcal{A}_{H_3} \times \mathcal{A}_{K_3})^T$ is a nondeterministic SFA with three bpi's, viz. $(p_0, p'_0), (p_0, p'_2)$ and (p_2, p'_0). Note that $rk(H_3 \cap K_3) = 4$ and the submonoids H_3 and K_3 satisfy HNP.

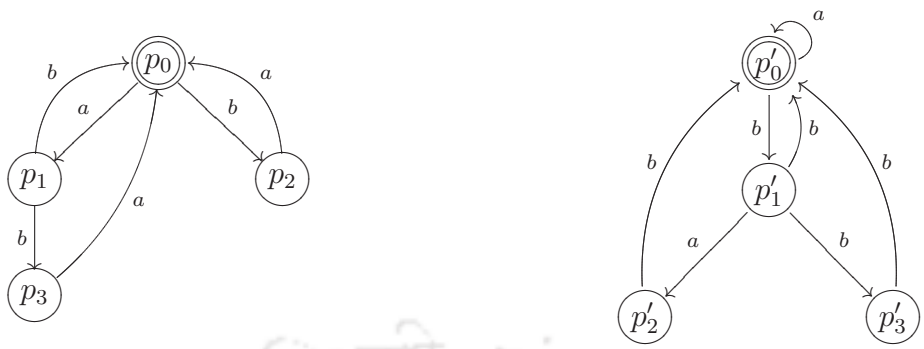

FIGURE 2.9: $\mathcal{A}_{H_3}$ (in the left) and $\mathcal{A}_{K_3}$ (in the right) of Example 2.5.3

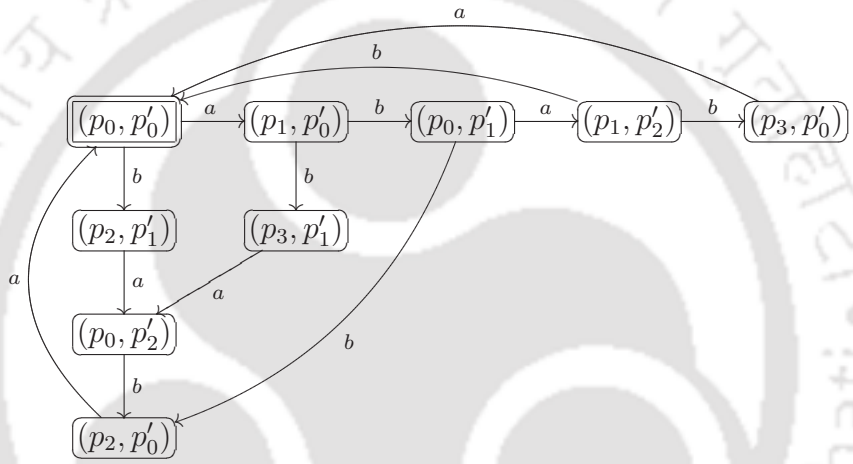

FIGURE 2.10: $(\mathcal{A}_{H_3} \times \mathcal{A}_{K_3})^T$ of Example 2.5.3

2.6 Conclusion

This work considers the intersection problem for two submonoids of a free monoid which are generated by finite prefix sets. In particular, this work has obtained a sufficient condition for the Hanna Neumann property for the class of submonoids generated by finite prefix sets. In that connection, a general rank formula for the submonoids which are accepted by semi-flower automata is also obtained. Thus, this work addresses one of the problems, viz. the prefix case, posed by Giambruno and Restivo in the conclusions of the paper [Giambruno and Restivo, 2008]. However, there is a lot more to investigate the general problem concerning the intersection of

two arbitrary submonoids of a free monoid. For instance, even in the prefix case, one could investigate the necessary and sufficient conditions for the Hanna Neumann property. On the other hand, the intersection problem for two submonoids generated by finite non-prefix sets of words is of particular interest. For this problem, the rank formula that is obtained (for nondeterministic automata) in this chapter may be useful.

3

Holonomy Decomposition

The primary decomposition theorem due to Krohn and Rhodes [1965] is one of the fundamental results in the theory of automata and monoids. Eilenberg's holonomy decomposition theorem for transformation monoids is a sophisticated version of the Krohn-Rhodes decomposition theorem [Eilenberg, 1976]. The holonomy decomposition method appears to be relatively efficient and has been implemented computationally by Egri-Nagy and Nehaniv [2010]. In order to understand some structural properties of SFA, we investigate their holonomy decomposition. We pursue the decomposition of SFA classified by their number of bpis. In this chapter, we obtain the holonomy decomposition of circular SFA with at most two bpis.

3.1 Transformation monoids

The purpose of this section is to present the holonomy decomposition theorem and its necessary background material. For more details, one may refer to [Dömösi and Nehaniv, 2005; Egri-Nagy, 2005; Eilenberg, 1976].

Definition 3.1.1. A *transformation monoid* is a pair (P, M), where P is a nonempty finite set and M is a submonoid of the monoid of all functions on P with respect to composition. Further, (P, M) is called *transformation group* if M is a group.

Notation 3.1.2. For $p \in P$, let $\widehat{p}$ be the constant function on P which takes the value p, i.e. $q\widehat{p} = p$, $\forall q \in P$.

In what follows, let (P, M) be a transformation monoid.

Definition 3.1.3. The *closure* of (P, M), denoted by $\widehat{(P, M)}$, is defined as $(P, \widehat{M})$, where $\widehat{M}$ is the monoid generated by $M \cup \bigcup_{p \in P} \{\widehat{p}\}$.

Definition 3.1.4. The *skeleton space* of (P, M) is a pair $(\mathscr{J}, \leq)$, where

$$\mathscr{J} = \Big\{ Pm \;\Big|\; m \in M \Big\} \cup \Big\{ \{p\} \;\Big|\; p \in P \Big\}$$

and the preorder $\leq$ is defined on $\mathscr{J}$ by

$$R \leq S \text{ if and only if } R \subseteq Sm, \text{ for some } m \in M.$$

For $R, S \in \mathscr{J}$, if $R \leq S$ and $R \neq S$, then we write $R < S$. Further, we define an equivalence relation $\sim$ on $\mathscr{J}$ by

$$R \sim S \text{ if and only if } R \leq S \text{ and } S \leq R.$$

Notation 3.1.5. For $i \geq 1$, we write $\mathscr{J}_i$ to denote the set of all elements of the skeleton space of cardinality i, i.e.

$$\mathscr{J}_i = \Big\{ S \in \mathscr{J} \;\Big|\; |S| = i \Big\}.$$

In what follows, let $\mathscr{J}$ be the skeleton space of (P, M).

Definition 3.1.6. For $S \in \mathscr{J}$, we define the set $K(S)$ to be the set of elements of M which behave as permutations on S, i.e.

$$K(S) = \{m \in M \mid Sm = S\}.$$

Definition 3.1.7. For $S \in \mathscr{J}$ with $|S| > 1$, we define *paving of* S, denoted by $B(S)$, to be the set of maximal elements (with respect to the set inclusion) of $\mathscr{J}$ that are contained in S, i.e.

$$B(S) = \{R \in \mathscr{J} \mid R \subsetneq S \text{ and if } T \in \mathscr{J} \text{ with } R \subseteq T \subseteq S \text{ then } R = T \text{ or } T = S\}.$$

Remark 3.1.8. Each $m \in K(S)$ acts as a permutation on $B(S)$.

Notation 3.1.9. For $m \in K(S)$, we write $\check{m}$ to denote the permutation on $B(S)$ induced by m.

Definition 3.1.10. For $S \in \mathscr{J}$ with $|S| > 1$, the set $G(S)$ of all permutations of $B(S)$ induced by the elements of $K(S)$ is called the *holonomy group* of S in (P, M).

Remark 3.1.11. The pair $\big(B(S), G(S)\big)$ is a transformation group.

Definition 3.1.12. A *height function* $h : \mathscr{J} \longrightarrow \mathbb{N}$ is defined inductively as follows. For $R \in \mathscr{J}$,

(i) If $|R| = 1$, then $Rh = 0$

(ii) If $|R| > 1$; let $R_0 < R_1 < R_2 < \cdots < R_n = R$ is a longest chain in $\mathscr{J}$ ending in R and $|R_0| = 1$. Then $Rh = n$.

The *height* of (P, M) is defined to be Ph.

Due to Eilenberg, every finite transformation monoid is covered by a wreath product of its holonomy transformation monoids, where covering relation and wreath product between transformation monoids are defined as follows.

Definition 3.1.13. Let (P_1, M_1) and (P_2, M_2) be transformation monoids. We say (P_1, M_1) is *covered* by (P_2, M_2), denoted by $(P_1, M_1) \prec (P_2, M_2)$, if there exists a surjective partial function $f : P_2 \longrightarrow P_1$ and, for any $m_1 \in M_1$, there exists an element $m_2 \in M_2$ such that

$$(p_2 f)m_1 = (p_2 m_2)f, \quad \forall p_2 \in P_2.$$

Definition 3.1.14. Let (P_1, M_1) and (P_2, M_2) be transformation monoids. The *wreath product* of (P_1, M_1) and (P_2, M_2), denoted by $(P_1, M_1) \circ (P_2, M_2)$, is the transformation monoid $(P_1 \times P_2, M_1^{P_2} \times M_2)$ with the following action:

$$(p_1, p_2)(f, m_2) = (p_1(p_2 f), p_2 m_2)$$

for all $(p_1, p_2) \in P_1 \times P_2$ and for all $(f, m_2) \in M_1^{P_2} \times M_2$, where $M_1^{P_2}$ is the set of all functions from P_2 to M_1.

Now, we state the *holonomy decomposition theorem for finite transformation monoids* in the following.

Theorem 3.1.15 ([Eilenberg, 1976])**.** *If (P, M) is a finite transformation monoid, then*

$$(P, M) \prec \widehat{\mathscr{H}_n} \circ \widehat{\mathscr{H}_{n-1}} \circ \ldots \circ \widehat{\mathscr{H}_1},$$

where $n = Ph$ and for $1 \leq i \leq n$,

$$\mathscr{H}_i = \left(\prod_{j=1}^{k_i} B(T_{ij}), \prod_{j=1}^{k_i} G(T_{ij}) \right),$$

where k_i is the number of equivalence classes at height i and $\{T_{ij} \mid 1 \leq j \leq k_i\}$ is the set of representatives of equivalence classes at height i.

The holonomy decomposition is also used to study the structural properties of certain algebraic structures [Holcombe, 1980; Krishna and Chatterjee, 2007]. The holonomy decomposition method is relatively efficient and has been implemented

computationally by Egri-Nagy and Nehaniv [2010]. One can use the computer algebra package, SgpDec developed by them to study the holonomy decomposition of transformation monoids.

In this chapter, we consider only complete and deterministic automata. Recall that an automaton is said to be a circular automaton if there exists an input letter which induces a circular permutation on the state set of the automaton. Circular automata have been studied in various contexts. The Černý conjecture is true for circular automata [Dubuc, 1998; Pin, 1978].

The holonomy decomposition looks for holonomy groups. These groups are the building blocks for the components of the decomposition. Egri-Nagy and Nehaniv [2005] proved that the monoid of an automaton is aperiodic (i.e. the subgroups in the monoid are trivial) if and only if the holonomy groups in the transformation monoid of the automaton are trivial. In view of Remark 1.4.5(i), it is nontrivial to investigate the holonomy groups in the transformation monoid of circular SFA. In this chapter, we pursue this task in circular SFA which are classified by their number of bpis.

3.2 Circular SFA

In this section, we establish some properties of circular semi-flower automata (CSFA). In what follows, $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ denotes a complete and deterministic automaton such that $|Q| = n$. Further, for $m \geq 1$, $\mathscr{C}_m$ denotes a transformation group (X, C_m), for some set X with $|X| = m$ and C_m is the cyclic group generated by a circular permutation on X.

The following lemma is useful in the sequel.

Lemma 3.2.1. *Let $\mathcal{A}$ be an SFA over A and $a, b \in A$.*

(i) *If $\overline{a}$ is a permutation on Q, then $\overline{a}$ is a circular permutation on Q.*

(ii) *If $\overline{a}$ and $\overline{b}$ are permutations on Q, then $\overline{a} = \overline{b}$.*

Proof.

(i) It is well known that any permutation on a nonempty finite set can be written as a composition of disjoint cycles [Dummit and Foote, 2004]. Let us assume that the permutation $\overline{a}$ is a product of more than one cycles. Then, a cycle which does not contain the initial-final state q_0 is clearly a cycle in $\mathcal{A}$ that does not pass through q_0. Since $\mathcal{A}$ is semi-flower, it is not possible. Thus, $\overline{a}$ has a single cycle on Q, so that $\overline{a}$ is a circular permutation on Q.

(ii) On the contrary, let us assume that $\overline{a} \neq \overline{b}$. From part (i), the permutations $\overline{a}$ and $\overline{b}$ are circular permutations on Q. Let cyclic orderings on Q with respect to $\overline{a}$ and $\overline{b}$ be as shown below.

$$\overline{a} : q_0, q_{i_1}, q_{i_2}, \ldots, q_{i_{n-1}}$$
$$\overline{b} : q_0, q_{j_1}, q_{j_2}, \ldots, q_{j_{n-1}}$$

Since $\overline{a} \neq \overline{b}$, let k be the least number such that $q_{i_k} \neq q_{j_k}$. Note that there exists $s > k$ such that $q_{i_k} = q_{j_s}$ and also there exists $r > k$ such that $q_{j_k} = q_{i_r}$. Now, the path as shown below

$$q_{i_k} \xrightarrow{a^{r-k}} q_{i_r} = q_{j_k} \xrightarrow{b^{s-k}} q_{j_s} = q_{i_k}$$

is a cycle labeled by $a^{r-k}b^{s-k}$. Clearly, this cycle does not pass through the initial-final state q_0. But, this is not possible in an SFA. Hence, $\overline{a} = \overline{b}$.

□

Corollary 3.2.2. *If $\mathcal{A}$ is a CSFA, then there is a unique circular permutation induced by the input symbols of $\mathcal{A}$.*

In this chapter, we present holonomy decomposition of CSFA classified by their number of bpis. In this context, we first prove certain properties pertaining to the bpis of an SFA.

Lemma 3.2.3. *Let $\mathcal{A}$ be an SFA over A; then,*

$$BPI(\mathcal{A}) = \varnothing \iff |A| = 1.$$

Proof. In an n-state complete and deterministic automaton,

the total indegree of all states = the total number of transitions = $n|A|$.

Since $\mathcal{A}$ is accessible, indegree of each state is at least one. Consequently,

$$BPI(\mathcal{A}) = \varnothing \iff \text{the total indegree of all states } = n \iff |A| = 1.$$

□

In what follows, let $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ be a CSFA. For the rest of the chapter, we fix the following regarding $\mathcal{A}$. Assume $a \in A$ induces a circular permutation $\overline{a}$ on the state set Q of $\mathcal{A}$. Accordingly,

$$\overline{a} : q_0, q_1, \ldots, q_{n-1}$$

is the cyclic ordering on Q with respect to $\overline{a}$.

Lemma 3.2.4. *If $\mathcal{A}$ has at least one bpi, then its initial-final state is always a bpi.*

Proof. Since $\mathcal{A}$ has at least one bpi, by Lemma 3.2.3, we have $|A| \geq 2$. We claim that $q_{n-1}\overline{b} = q_0$, for all $b \in A$, so that q_0 is a bpi. Let us assume the contrary, i.e. $q_{n-1}\overline{c} \neq q_0$, for some $c \in A$. Since $\mathcal{A}$ is complete and deterministic, $q_{n-1}\overline{c} = q_i$, for some i (with $1 \leq i < n$). Note that $q_i\overline{a^{n-i-1}c} = q_i$. Thus, we have a cycle in $\mathcal{A}$ from q_i to q_i labeled by $a^{n-i-1}c$ that does not visit q_0. This is a contradiction. Hence, $q_{n-1}\overline{b} = q_0$, for all $b \in A$. □

Definition 3.2.5. Let X be a nonempty finite set and f a function on X. The *rank* of f, denoted by rank(f), is the cardinality of the image set Xf.

Lemma 3.2.6. *For $1 \leq m < n$, if $|BPI(\mathcal{A})| = m$, then any non-permutation in $M(\mathcal{A})$ has rank at most m.*

Proof. Since $m \geq 1$, by Lemma 3.2.3, there is a $b \in A \setminus \{a\}$. It is clear that $\overline{a}$ contributes one to the indegree of each state of $\mathcal{A}$. For $b \in A \setminus \{a\}$, if $\overline{b}$ is a permutation, then $|Q\overline{b}| = n > m$. Therefore, we have $|BPI(\mathcal{A})| = n > m$; which is a contradiction. Thus, for all $b \in A \setminus \{a\}$, the function $\overline{b}$ is not a permutation; in fact, $|Q\overline{b}| \leq m$. Now, for $x \in A^*$, if $\overline{x}$ is a non-permutation, then x contains a symbol b from $A \setminus \{a\}$. Hence, the rank of $\overline{x}$ is at most m. □

In view of Lemma 3.2.4, we have the following corollary of Lemma 3.2.6.

Corollary 3.2.7. *If $\mathcal{A}$ has a unique bpi, then $Q\overline{b} = \{q_0\}$, for all $b \in A \setminus \{a\}$.*

3.3 CSFA with at most one bpi

In this section, we present holonomy decomposition of CSFA with at most one bpi. We first observe that the holonomy decomposition of SFA with no bpis follows from the general case of permutation SFA. Recall that an automaton is a permutation automaton if the function induced by each input symbol is a permutation on the state set. Clearly, an automaton is a permutation automaton if and only if its monoid is a group.

By Lemma 3.2.1, we have the following proposition which also provides the holonomy decomposition of a permutation SFA.

Proposition 3.3.1. *If $\mathcal{A}$ is a permutation SFA, then $M(\mathcal{A})$ is a cyclic group. Further,*

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n}.$$

Now, we investigate the holonomy decomposition of CSFA with no bpis. By Lemma 3.2.3, if $\mathcal{A}$ is an SFA with no bpis, then $|A| = 1$, say $A = \{a\}$. Note that the function $\overline{a}$ is a circular permutation on Q. Thus, $\mathcal{A}$ is a circular as well as permutation SFA. Hence, by Proposition 3.3.1, we have the following theorem.

Theorem 3.3.2. *Let $\mathcal{A}$ be an SFA with no bpis, then*

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n}.$$

Now, we present the holonomy decomposition of CSFA with a unique bpi in the following theorem.

Theorem 3.3.3. *If $\mathcal{A}$ is a CSFA with a unique bpi, then*

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n}.$$

Proof. By Corollary 3.2.7, we have $Q\overline{b} = \{q_0\}$, for all $b \in A \setminus \{a\}$. This implies that $\overline{b} = \overline{c}$, for all $b, c \in A \setminus \{a\}$. Thus, $M(\mathcal{A})$ is generated by the set $\{\overline{a}, \overline{b}\}$.

For $\overline{x} \in M(\mathcal{A})$, by Lemma 3.2.6, we have either $|Q\overline{x}| = n$ or $|Q\overline{x}| = 1$. Consequently, the skeleton space of $(Q, M(\mathcal{A}))$ is

$$\mathscr{J} = \{Q\} \cup \mathscr{J}_1.$$

Note that

$$K(Q) = \left\{ \overline{a^i} \;\middle|\; 1 \leq i \leq n \right\} \quad \text{and} \quad B(Q) = \mathscr{J}_1.$$

Clearly, $|B(Q)| = n$ and the holonomy group of Q is

$$G(Q) = \left\{ \check{\overline{a^i}} \;\middle|\; 1 \leq i \leq n \right\},$$

where each element $\check{\overline{a^i}}$ is the permutation on $B(Q)$ induced by the corresponding element $\overline{a^i} \in K(Q)$. For $1 \leq i \leq n$, since $\overline{a^i} = \overline{a}^i$, we have $\check{\overline{a^n}} = (\check{\overline{a}}^n) = \check{\varepsilon}$. This implies that the holonomy group $G(Q)$ is a cyclic group of order n generated by $\check{\overline{a}}$. Consequently, we have

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n}.$$

□

3.4 CSFA with two bpis

The present section investigate the holonomy decomposition of CSFA with two bpis. In this section, $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ denotes a CSFA with two bpis. By Lemma 3.2.4, the initial-final state q_0 of $\mathcal{A}$ is a bpi. Let q_m, where $1 \leq m < n$, be the other bpi of $\mathcal{A}$ so that $BPI(\mathcal{A}) = \{q_0, q_m\}$. Note that, by Lemma 3.2.3, we have $|A| \geq 2$.

Lemma 3.4.1.

(i) *For $b \in A$, if rank$(\bar{b}) = 2$, then $Q\bar{b} = BPI(\mathcal{A})$.*

(ii) *There exists a symbol $b \in A$ such that $Q\bar{b} = BPI(\mathcal{A})$.*

Proof. We first note that $\bar{a}$ contributes one to the indegree of each state in Q. Since $BPI(\mathcal{A}) = \{q_0, q_m\}$, we have $Q\bar{b} \subseteq \{q_0, q_m\}$, for all $b \in A \setminus \{a\}$.

(i) Straightforward from the above statement.

(ii) Let us assume that $Q\bar{b} \neq \{q_0, q_m\}$, for all $b \in A \setminus \{a\}$. Then, for $b \in A \setminus \{a\}$, either $Q\bar{b} = \{q_0\}$ or $Q\bar{b} = \{q_m\}$. For some $b \in A \setminus \{a\}$, if $Q\bar{b} = \{q_m\}$, then there is loop at q_m; which is not possible. Consequently, for all $b \in A \setminus \{a\}$, $Q\bar{b} = \{q_0\}$. This implies $BPI(\mathcal{A}) = \{q_0\}$; a contradiction. Hence, there exists $b \in A$ such that $Q\bar{b} = BPI(\mathcal{A})$.

□

The following lemma provides the skeleton space of the transformation monoid $(Q, M(\mathcal{A}))$.

Lemma 3.4.2. *The skeleton space of the transformation monoid $(Q, M(\mathcal{A}))$ is given by*

$$\mathscr{J} = \{Q\} \cup \mathscr{J}_2 \cup \mathscr{J}_1,$$

where

$$\mathscr{J}_2 = \left\{ \{q_0, q_m\}\overline{a^i} \;\middle|\; 1 \leq i \leq n \right\}.$$

Proof. In view of Lemma 3.2.6, other than Q and singletons, the skeleton space $\mathscr{J}$ can have some sets of size two. Thus, it is sufficient to determine $\mathscr{J}_2$.

By Lemma 3.4.1(ii), there exists an input symbol, say $b \in A \setminus \{a\}$, such that $Q\overline{b} = \{q_0, q_m\}$. Therefore, for all $1 \leq i \leq n$, the image set

$$Q\overline{ba^i} = \{q_0, q_m\}\overline{a^i} \in \mathscr{J}_2.$$

Thus, we have

$$\left\{\{q_0, q_m\}\overline{a^i} \;\middle|\; 1 \leq i \leq n\right\} \subseteq \mathscr{J}_2.$$

Let us assume that $Q\overline{w} \in \mathscr{J}_2$, for some $w \in A^*$. Then w is of the form

$$w = a^{i_1}b_1a^{i_2}b_2 \cdots a^{i_k}b_ka^{i_{k+1}},$$

for $i_j \geq 0$ $(1 \leq j \leq k+1)$ and $b_i \in A$ $(1 \leq i \leq k)$ such that the rank of each function $\overline{b_i}$ is two (cf. Lemma 3.2.6). Write $w = a^{i_1}b_1ub_ka^{i_{k+1}}$, where $u = a^{i_2}b_2 \cdots a^{i_k}$. Since $\text{rank}(\overline{b_1ub_k}) = \text{rank}(\overline{b_k}) = 2$, we have

$$Q\overline{b_1ub_k} = Q\overline{b}_k = \{q_0, q_m\},$$

by Lemma 3.4.1(i). Consequently,

$$Q\overline{w} = Q\overline{a^{i_1}b_1ub_ka^{i_{k+1}}} = \{q_0, q_m\}\overline{a^{i_{k+1}}}.$$

Hence,

$$\mathscr{J}_2 = \left\{\{q_0, q_m\}\overline{a^i} \;\middle|\; 1 \leq i \leq n\right\}.$$

□

Remark 3.4.3. The cardinality of $\mathscr{J}_2$ is not necessarily n as shown in Example 3.4.4.

Example 3.4.4. The automaton $\mathcal{A}$ given in FIGURE 3.1 is a CSFA with $BPI(\mathcal{A}) = \{1, 3\}$, and $|Q| = 4$. Here, $Q\overline{b} = \{1, 3\}$ and we observe that

$$\{1, 3\}\overline{a} = \{2, 4\}, \{1, 3\}\overline{a^2} = \{1, 3\}, \quad \text{so that} \quad |\mathscr{J}_2| = 2.$$

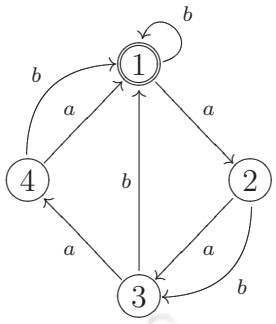

FIGURE 3.1: A CSFA $\mathcal{A}$ with two bpis

Lemma 3.4.5. *There exists $x \in A^*$ such that $q_0\overline{x} = q_m$ and $q_m\overline{x} = q_0$.*

Proof. If there exists $b \in A \setminus \{a\}$ such that $q_0\overline{b} \neq q_0$, then clearly $x = b$ will serve the purpose. Otherwise, we have $q_0\overline{b} = q_0$, for all $b \in A \setminus \{a\}$. However, by Lemma 3.4.1, there exist a symbol $c \in A$ such that $Q\overline{c} = \{q_0, q_m\}$. If $c = a$, then $Q = \{q_0, q_m\}$ and the result is straightforward.

Let us assume that $c \neq a$. Clearly, $q_0\overline{c} = q_0$ and there exists a state q_i (with $1 \leq i < k$) such that $q_i\overline{c} = q_m$. Let t (with $1 \leq t < m$) be the least number such that $q_t\overline{c} = q_m$. Choose $x = a^t c$ and observe that $q_0\overline{x} = q_m$. We claim that $q_m\overline{x} = q_0$.

On the contrary, assume $q_m\overline{x} \neq q_0$. Then, $q_m\overline{x} = q_m$ so that there is a cycle from q_m to q_m labeled x. Thus, the cycle should pass through q_0. Since $q_0\overline{c} = q_0$, there exist t_1 and t_2 $(1 \leq t_1, t_2 < t)$ with $t_1 + t_2 = t$ such that

$$q_m\overline{a^{t_1}} = q_0 \text{ and } q_0\overline{a^{t_2}c} = q_m.$$

Note that $q_0\overline{a^{t_2}c} = q_{t_2}\overline{c} = q_m$. This contradicts the choice of t, as $t_2 < t$. Thus, $q_m\overline{x} = q_0$. □

Theorem 3.4.6. *If $\mathcal{A}$ is a CSFA with $BPI(\mathcal{A}) = \{q_0, q_m\}$, then*

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_r} \circ \widehat{\mathscr{C}_2},$$

where r (with $1 \leq r \leq n$) is the smallest number such that $\{q_0, q_m\}\overline{a^r} = \{q_0, q_m\}$.

Proof. From Lemma 3.4.2, the skeleton space of $(Q, M(\mathcal{A}))$ is

$$\mathscr{J} = \{Q\} \cup \mathscr{J}_2 \cup \mathscr{J}_1$$

in which all the elements of $\mathscr{J}_2$ are equivalent to each other.

For $1 \leq i \leq n$, note that $\overline{a^i}$ permutes the elements of Q and, for $x \in A^*$, if $\overline{x} \neq \overline{a^i}$, then $\overline{x}$ is not a permutation on Q (cf. Lemma 3.2.1). Consequently,

$$K(Q) = \left\{ \overline{a^i} \;\middle|\; 1 \leq i \leq n \right\}.$$

Since all the elements of $\mathscr{J}_2$ are maximal in Q, we have $B(Q) = \mathscr{J}_2$. Let r (with $1 \leq r \leq n$) be the smallest integer such that $\{q_0, q_m\}\overline{a^r} = \{q_0, q_m\}$ so that $|B(Q)| = r$. Consequently, the holonomy group

$$G(Q) = \left\{ \breve{\overline{a^i}} \;\middle|\; 1 \leq i \leq r \right\},$$

where each function $\breve{\overline{a^i}}$ is a permutation on $B(Q)$ induced by the corresponding function $\overline{a^i} \in K(Q)$. Since $\breve{\overline{a^i}} = \breve{\overline{a}}^i$, the holonomy group $G(Q)$ is a cyclic group of order r generated by $\breve{\overline{a}}$. Thus,

$$(B(Q), G(Q)) = \mathscr{C}_r.$$

Let $P = \{q_0, q_m\}$ be a representative in $\mathscr{J}_2$. Clearly,

$$B(P) = \{\{q_0\}, \{q_m\}\}.$$

By Lemma 3.4.5, there exist $x \in A^*$ such that $q_0\overline{x} = q_m$ and $q_m\overline{x} = q_0$, so that $K(P) = \{\overline{x}, \overline{\varepsilon}\}$. Consequently, the holonomy group $G(P) = C_2$ and hence,

$$(B(P), G(P)) = \mathscr{C}_2.$$

Thus, the holonomy decomposition of $\mathcal{A}$ is given by

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_r} \circ \widehat{\mathscr{C}_2}.$$

□

Corollary 3.4.7. *Let n be an odd number; if $\mathcal{A}$ is a CSFA with two bpis such that $|Q| = n$, then*

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n} \circ \widehat{\mathscr{C}_2}.$$

Proof. From Theorem 3.4.6, we have

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_r} \circ \widehat{\mathscr{C}_2},$$

where r (with $1 \leq r \leq n$) is the smallest number such that $\{q_0, q_m\}\overline{a^r} = \{q_0, q_m\}$. We claim that $r = n$. If $r < n$, since $\{q_0, q_m\}\overline{a^r} = \{q_0, q_m\}$ and $\overline{a}$ is a circular permutation on Q, it follows that $q_0\overline{a^r} = q_m$, and $q_m\overline{a^r} = q_0$. This implies that $q_0\overline{a^{2r}} = q_0$ with $1 < 2r < 2n$. Therefore, $2r = n$; a contradiction. Hence,

$$(Q, M(\mathcal{A})) \prec \widehat{\mathscr{C}_n} \circ \widehat{\mathscr{C}_2}.$$

□

3.5 Conclusion

In this work, we have initiated the investigations on the holonomy decomposition of CSFA, classified by their number of bpis. In fact, first we have ascertained the holonomy decomposition of CSFA with at most one bpi. Further, we obtained the holonomy decomposition of CSFA with two bpis. One can target to address the holonomy decomposition of CSFA with arbitrary number of bpis. In general, one can look for holonomy decomposition of SFA.

4

Syntactic Complexity

The syntactic complexity of a class of recognizable languages provides a measure for the complexity of the class. Holzer and König [2004] observed that the syntactic complexity of the class of all recognizable languages over unary alphabet is linear; while it is maximum, if the size of alphabet is at least three. It turns out that the most crucial case is to determine the syntactic complexity for recognizable languages over a binary alphabet. In this chapter, we investigate the syntactic complexity of various classes of submonoids accepted by CSFA, classified by their number of bpis. We first give the syntactic complexity of the class of submonoids accepted by CSFA with at most one bpi. Then, we consider CSFA with two bpis over a binary alphabet and obtain the syntactic complexity of the respective submonoids.

We now formally present the notion of syntactic complexity. In view of Theorem 1.4.8 and Theorem 1.4.10, we have the following definitions.

Definition 4.0.1. Let L be a recognizable language.

(i) The number of states in the minimal automaton accepting L is called the *state complexity* of L.

(ii) The size of the syntactic monoid of L is called the *syntactic complexity* of L.

Definition 4.0.2. The *syntactic complexity of a class* of recognizable languages is the maximal syntactic complexity of languages in that class, taken as a function of the state complexity of these languages.

Recall that the syntactic monoid of a recognizable language is isomorphic to the monoid of the minimal automaton accepting the language (cf. Theorem 1.4.11). Therefore, in order to calculate the syntactic complexity of a recognizable language, it is convenient to consider the monoid of the minimal automaton accepting the language.

4.1 The monoid of CSFA

In this section, we prove some of the basic properties and facts about CSFA which are useful in this chapter. We consider only complete and deterministic automata throughout the chapter. In what follows, let $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ be a complete and deterministic CSFA with $|Q| = n$ and $M(\mathcal{A})$ the monoid of $\mathcal{A}$. We follow the same notation introduced in Chapter 3, regarding CSFA. We denote by G the submonoid generated by $\overline{a}$ in $M(\mathcal{A})$, where $a \in A$ induces a circular permutation on Q.

We first prove some results which are useful in the sequel.

Proposition 4.1.1. *The submonoid G is a cyclic subgroup of order n in $M(\mathcal{A})$. Further, G contains all the permutations of $M(\mathcal{A})$.*

Proof. Since G is the submonoid generated by circular permutation $\overline{a}$ on Q, we have G is a cyclic group of order n. Now, let $\overline{x} \in M(\mathcal{A})$ be a permutation on Q for $x = a_1a_2\cdots a_m$ with $a_i \in A$ $(1 \leq i \leq m)$. Then

$$\overline{x} = \overline{a_1a_2\cdots a_m} = \overline{a}_1\overline{a}_2\cdots\overline{a}_m.$$

Clearly, each function $\overline{a_i}$ is a permutation on Q. By Lemma 3.2.1(ii), we have $\overline{a} = \overline{a_i}$, for all i $(1 \leq i \leq m)$. This implies that $\overline{x} = \overline{a^m}$ and consequently $\overline{x} \in G$. □

Remark 4.1.2. For $p, q \in Q$, there exists $\overline{x} \in G$ such that $p\overline{x} = q$. Indeed, if $p = q_i$ and $q = q_j$, for some i, j (with $1 \leq i \leq j \leq n$), then $\overline{x} = \overline{a^{j-i}}$ will serve the purpose.

Proposition 4.1.3. *$\mathcal{A}$ is a minimal automaton.*

Proof. Since $\mathcal{A}$ is accessible, in view of Theorem 1.4.7, it is sufficient to prove that the relation $\sim_{\mathcal{A}}$ is diagonal. Let p and q be two distinct states. By Remark 4.1.2, there exists $\overline{x} \in G$ such that $p\overline{x} = q_0$.

Now, we claim that $q\overline{x} \neq q_0$. For, if $q\overline{x} = q_0$, then $p\overline{x} = q\overline{x}$. Since $\overline{x} \in G$, we have $p = q$; a contradiction. Hence, the relation $\sim_{\mathcal{A}}$ is diagonal and consequently, $\mathcal{A}$ is minimal. □

We now recall the notion of group actions and its related concepts which are useful in the present context. For more details, one may refer to any book on basic abstract algebra (cf. [Dummit and Foote, 2004]).

Definition 4.1.4. Let $(H, \circ)$ be a group with identity e and X a nonempty set. A *group action* of H on X is a function $\cdot : X \times H \longrightarrow X$ satisfying the following axioms. For $x \in X$ and $h, h' \in H$,

$$x \cdot e = x \ \text{ and } \ x \cdot (h \circ h') = (x \cdot h) \cdot h'.$$

For $x \in X$, the *orbit of* x, denoted by $\mathcal{O}(x)$, is the equivalence class of x with respect to the equivalence relation $\sim$ on X defined by

$$x \sim y \iff x \cdot h = y \ \text{ for some } \ h \in H.$$

Clearly, $\mathcal{O}(x) = \{x \cdot h \mid h \in H\}$. Further, for $x \in X$, the *stabilizer of* x, denoted by H_x, is the subgroup of H defined by

$$H_x = \{h \in H \mid x \cdot h = x\}.$$

Remark 4.1.5. For $x \in X$, $|\mathcal{O}(x)| = [H, H_x]$, the index of the stabilizer H_x in H.

Let us consider the group action of G on $M(\mathcal{A})$ with respect to the monoid operation, the composition of functions. Note that

$$M(\mathcal{A}) = \bigcup_{x \in A^*} \mathcal{O}(\overline{x}).$$

Proposition 4.1.6. *For $x \in A^*$, we have $|\mathcal{O}(\overline{x})| = n$.*

Proof. For $x \in A^*$, we have $|\mathcal{O}(\overline{x})| = [G, G_{\overline{x}}]$. Since $|G| = n$, it is sufficient to prove that $G_{\overline{x}} = \{\overline{\varepsilon}\}$. Let $\overline{y} \in G_{\overline{x}}$, we have $\overline{x}\ \overline{y} = \overline{x}$. This implies that, for $q \in Q$, $q(\overline{x}\ \overline{y}) = q\overline{x}$, i.e. $(q\overline{x})\overline{y} = q\overline{x}$. Write $q\overline{x} = q'$, then $q'\overline{y} = q'$.

We claim that $\overline{y} = \overline{\varepsilon}$. Let $p \in Q$ be an arbitrary state. By Remark 4.1.2, there exists $\overline{z} \in G$ such that $p = q'\overline{z}$. Consider

$$p\overline{y} = (q'\overline{z})\overline{y} = (q'\overline{y})\overline{z} = q'\overline{z} = p.$$

Hence, $\overline{y} = \overline{\varepsilon}$ and consequently $G_{\overline{x}} = \{\overline{\varepsilon}\}$. □

Thus, to compute the size of $M(\mathcal{A})$, it is sufficient to count the number of distinct orbits with respect to the group action of G on $M(\mathcal{A})$. In the following sections, we investigate the size of $M(\mathcal{A})$ classified by the number of bpis of $\mathcal{A}$.

4.2 CSFA with at most one bpi

In this section, we investigate the syntactic complexity of the submonoids accepted by CSFA with at most one bpi. We first observe that the syntactic complexity of the submonoids accepted by SFA with no bpis follows from the general case

of permutation SFA. By Lemma 3.2.1(i) and Proposition 4.1.3, any permutation SFA is a minimal automaton. Now, by Lemma 3.2.1(ii), we have the following proposition which also provides the syntactic complexity of the submonoids accepted by permutation SFA.

Proposition 4.2.1. *If $\mathcal{A}$ is a permutation SFA, then $M(\mathcal{A})$ is a cyclic group generated by $\overline{a}$. Further, the syntactic complexity of the submonoids accepted by permutation SFA is n.*

Let $\mathcal{A}$ be an SFA with no bpis, then by Lemma 3.2.3, we have $|A| = 1$, say $A = \{a\}$. Note that the function $\overline{a}$ is a circular permutation on Q. Thus, $\mathcal{A}$ is a circular as well as permutation SFA. Hence, by Proposition 4.2.1, we have the following theorem.

Theorem 4.2.2. *The syntactic complexity of the submonoids accepted by SFA with no bpis is n.*

Now, we determine the syntactic complexity of the submonoids accepted by CSFA with a unique bpi in the following theorem.

Theorem 4.2.3. *The syntactic complexity of the submonoids accepted by CSFA with a unique bpi is $2n$.*

Proof. Let $\mathcal{A}$ be a CSFA with a unique bpi. By Corollary 3.2.7, we have $Q\overline{b} = \{q_0\}$, for all $b \in A \setminus \{a\}$. This implies that for $b, c \in A \setminus \{a\}$, we have $\overline{b} = \overline{c}$. Now, we take a symbol $b \in A \setminus \{a\}$. The orbit of $\overline{b}$ is

$$\mathcal{O}(\overline{b}) = \{\overline{ba^i} \mid 1 \leq i \leq n\}.$$

Let $\overline{x}$ be a non-permutation in $M(\mathcal{A})$. By Lemma 3.2.6, the non-permutation $\overline{x}$ is a constant function. This implies that $Q\overline{x} = \{q_k\}$, for some k (with $0 \leq k < n$). Note that $Q\overline{ba^k} = \{q_k\}$. Therefore, $\overline{x} = \overline{ba^k} \in \mathcal{O}(\overline{b})$ and consequently the orbit $\mathcal{O}(\overline{b})$ contains all non-permutations in $M(\mathcal{A})$.

Thus, there are exactly two distinct orbits, one with all permutations (i.e. G) and other with all non-permutations. By Proposition 4.1.6, we have $|M(\mathcal{A})| = 2n$. Since $\mathcal{A}$ is arbitrary, we have the syntactic complexity of the submonoids accepted by CSFA with a unique bpi is $2n$. □

4.3 CSFA with two bpis

In this section, we investigate the syntactic complexity of CSFA with two bpis. In the previous section, we have observed that the syntactic complexity of CSFA with at most one bpi is independent of the size of the input alphabet. In contrast, the syntactic complexity of CSFA with two bpis varies with respect to the size of input alphabet. It is evident from the following example.

Example 4.3.1. Consider the CSFA $\mathcal{A}_1 = (\{q_0, q_1, q_2, q_3, q_4\}, \{a, b\}, q_0, q_0, \mathcal{F}_1)$, where the set of transitions $\mathcal{F}_1$ is given by the following table.

$\mathcal{F}_1$	q_0	q_1	q_2	q_3	q_4
a	q_1	q_2	q_3	q_4	q_0
b	q_3	q_3	q_3	q_0	q_0

Also, consider the CSFA $\mathcal{A}_2 = (\{q_0, q_1, q_2, q_3, q_4\}, \{a, b, c\}, q_0, q_0, \mathcal{F}_2)$, where the set of transitions $\mathcal{F}_2$ is given by the following table.

$\mathcal{F}_2$	q_0	q_1	q_2	q_3	q_4
a	q_1	q_2	q_3	q_4	q_0
b	q_3	q_3	q_3	q_0	q_0
c	q_3	q_0	q_3	q_0	q_0

Note that $\mathcal{A}_1$ and $\mathcal{A}_2$ are 5-state CSFA with the same set of bpis $\{q_0, q_3\}$. The automaton $\mathcal{A}_1$ is defined over the binary alphabet $\{a, b\}$, while $\mathcal{A}_2$ is defined over the ternary alphabet $\{a, b, c\}$. We observed that the syntactic complexity of the submonoid accepted by $\mathcal{A}_1$ is 60, whereas it is 110 for the submonoid accepted by

$\mathcal{A}_2$. One can observe these numbers through the computer algebra system GAP [2012].

In view of Example 4.3.1, in this section, we restrict ourselves to investigate the syntactic complexity of the class of the submonoids accepted by CSFA with two bpis over a binary alphabet. In fact, in this section, we prove the following main theorem.

Theorem 4.3.2. *The syntactic complexity of the class of the submonoids accepted by CSFA with two bpis over a binary alphabet is* $2n(n+1)$.

We fix the following notation for rest of the section. Let $A = \{a, b\}$ be the binary alphabet and $\mathcal{A} = (Q, A, q_0, q_0, \mathcal{F})$ a CSFA with two bpis. As earlier, $\overline{a}$ is the circular permutation. Let $\overline{b}$ be the non-permutation. Note that, $Q\overline{b} = BPI(\mathcal{A})$. By Lemma 3.2.4, the initial-final state q_0 is a bpi. Let q_m, for some m (with $1 \leq m < n$), be the other bpi of $\mathcal{A}$ so that $BPI(\mathcal{A}) = \{q_0, q_m\}$. We need to establish some results for proving Theorem 4.3.2. In the following, these results are presented in various subsections.

4.3.1 Idempotents

In this subsection, we obtain the idempotents of $M(\mathcal{A})$ which will be useful to give a representation of the elements of $M(\mathcal{A})$. In view of Lemma 3.2.6, for $x \in A^*$, we have $\text{rank}(\overline{x}) \in \{1, 2, n\}$. Clearly, the identity element $\overline{\varepsilon}$ in $M(\mathcal{A})$ is only idempotent of rank n. All the elements of rank one in $M(\mathcal{A})$ are idempotent, provided that they exist. We now estimate idempotents of rank two in $M(\mathcal{A})$. For that, we first prove the following results.

Proposition 4.3.3. *For* $1 \leq i \leq n$ *and* $x \in A^*$, *if* $\overline{x}$ *is an idempotent in* $M(\mathcal{A})$, *then* $(\overline{a^i x a^{n-i}})$ *is also an idempotent in* $M(\mathcal{A})$.

Proof. Given that $\overline{x}^2 = \overline{x}$. For $1 \leq i \leq n$, consider

$$\overline{(a^i x a^{n-i})}^2 = (\overline{a^i x a^{n-i}})(\overline{a^i x a^{n-i}}) = (\overline{a^i x^2 a^{n-i}}) = (\overline{a^i x a^{n-i}}).$$

Hence, $(\overline{a^i x a^{n-i}})$ is an idempotent in $M(\mathcal{A})$. □

Remark 4.3.4. Let $\overline{x}$ be an element in $M(\mathcal{A})$ such that $Q\overline{x} = \{q_0, q_m\}$. If $q_0\overline{x} = q_0$ and $q_m\overline{x} = q_m$, then $\overline{x}$ is an idempotent.

Proposition 4.3.5. *Let t be a natural number such that $t < m < n$; there exists a natural number k such that $m \leq t + k(n-m) < n$.*

Proof. Since $n-m > 0$, note that the sequence $\{t+i(n-m)\}_{i=0,1,2,\ldots}$ is an increasing sequence. Let k be the least number such that $m \leq t + k(n-m)$. We prove that $t + k(n-m) < n$. Since k is least, we have $t + (k-1)(n-m) < m$. This implies that

$$t + (k-1)n - km < 0.$$

Now, we have $t + k(n-m) = t + (k-1)n - km + n < n$. □

Lemma 4.3.6. *There exists a natural number r (with $1 \leq r < n$) such that the function $\overline{a^r b}$ is an idempotent of rank two in $M(\mathcal{A})$.*

Proof. Since q_m is the bpi of $\mathcal{A}$, there exists j (with $0 \leq j < m$) such that $q_j\overline{b} = q_m$. Let t (with $0 \leq t < m$) be the least number such that $q_t\overline{b} = q_m$ so that $q_0\overline{a^t b} = q_m$. Consequently, as $q_m\overline{a^{n-m}} = q_0$, we have

$$q_m\overline{a^{n-m+t}b} = q_m.$$

If $q_0\overline{a^{n-m+t}b} = q_0$, then choose $r = n - m + t$ and by Remark 4.3.4, the function $\overline{a^r b}$ is an idempotent of rank two in $M(\mathcal{A})$. Otherwise, since the letter b is suffix of word $a^{n-m+t}b$, we have $q_0\overline{a^{n-m+t}b} = q_m$. Then

$$q_m\overline{a^{2(n-m)+t}b} = q_m.$$

If $q_0\overline{a^{2(n-m)+t}b} = q_0$, then choose $r = 2(n-m)+t$ and again by Remark 4.3.4, the function $\overline{a^r b}$ is an idempotent of rank two in $M(\mathcal{A})$. Otherwise, since the letter b is suffix of word $a^{2(n-m)+t}b$, we have $q_0\overline{a^{2(n-m)+t}b} = q_m$. Then

$$q_m\overline{a^{3(n-m)+t}b} = q_m.$$

As long as we continue this process, in each i^{th} step, we have $q_m\overline{a^{i(n-m)+t}b} = q_m$. Note that, by Proposition 4.3.5, there exists a natural number k such that $m \leq k(n-m)+t < n$. If the above process terminates with a number r before k^{th} step, then we are through. Otherwise, in the k^{th} step, we have $q_m\overline{a^{k(n-m)+t}b} = q_m$. Moreover, since $m \leq k(n-m)+t < n$,

$$q_0(\overline{a^{t+k(n-m)}b}) = q_{k(n-m)+t}\overline{b} = q_0.$$

Thus, choose $r = k(n-m)+t$, and hence by Remark 4.3.4, the function $\overline{a^r b}$ is an idempotent of rank two in $M(\mathcal{A})$. □

Notation 4.3.7. In this chapter, κ denotes the number obtained in Lemma 4.3.6. That is, $\overline{a^\kappa b}$ is an idempotent of rank two in $M(\mathcal{A})$.

Lemma 4.3.8.

(i) *If $q_0\overline{b} \neq q_0$, then $\overline{b}^2$ is an idempotent of rank two in $M(\mathcal{A})$.*

(ii) *If $q_0\overline{b} = q_0$, then there exists t (with $1 \leq t < m$) such that the function $(\overline{a^t b})^2$ is an idempotent of rank two in $M(\mathcal{A})$.*

Proof. We know that $q_m\overline{b} = q_0$.

(i) Since $q_0\overline{b} \neq q_0$, we have $q_0\overline{b} = q_m$. Consider $Q\overline{b}^2 = (Q\overline{b})\overline{b} = \{q_0, q_m\}\overline{b} = \{q_0, q_m\}$. Also, since $q_0\overline{b}^2 = q_0$ and $q_m\overline{b}^2 = q_m$, by Remark 4.3.4, the function $\overline{b}^2$ is an idempotent of rank two in $M(\mathcal{A})$.

(ii) Since $q_0\overline{b} = q_0$, the state q_1 is not a bpi. Therefore, $1 < m < n$. Further, there exists j (with $0 < j < m$) such that $q_j\overline{b} = q_m$. Let t (with $1 \leq t < m$) be the least number such that $q_t\overline{b} = q_m$ so that $q_0\overline{a^t b} = q_m$. We claim that $q_m\overline{a^t b} = q_0$.

On the contrary, assume that $q_m\overline{a^tb} \neq q_0$. Then, $q_m\overline{a^tb} = q_m$ so that there is a cycle from q_m to q_m labeled by a^tb. Since $\mathcal{A}$ is an SFA, the cycle should pass through q_0. Since $q_0\overline{b} = q_0$, there exist t_1 and t_2 $(1 \leq t_1, t_2 < t)$ with $t_1 + t_2 = t$ such that

$$q_m\overline{a^{t_1}} = q_0 \text{ and } q_0\overline{a^{t_2}b} = q_m.$$

Note that $q_0\overline{a^{t_2}b} = q_{t_2}\overline{b} = q_m$. This contradicts the choice of t, as $t_2 < t$. Thus, $q_m\overline{a^tb} = q_0$.

Now, observe that $Q(\overline{a^tb})^2 = (Q\overline{a^tb})\overline{a^tb} = \{q_0, q_m\}\overline{a^tb} = \{q_0, q_m\}$. Further, $q_0(\overline{a^tb})^2 = q_0$ and $q_m(\overline{a^tb})^2 = q_m$. By Remark 4.3.4, the function $(\overline{a^tb})^2$ is an idempotent of rank two in $M(\mathcal{A})$. □

Notation 4.3.9. In this chapter, τ denotes the number obtained in Lemma 4.3.8(ii). That is, if $q_0\overline{b} = q_0$, then $(\overline{a^\tau b})^2$ is an idempotent of rank two in $M(\mathcal{A})$.

In view of Proposition 4.3.3, we have the following corollary of Lemma 4.3.6 and Lemma 4.3.8.

Corollary 4.3.10. *For $1 \leq i \leq n$,*

(i) $\overline{a^i(a^\kappa b)a^{n-i}}$ *is an idempotent of rank two in $M(\mathcal{A})$.*

(ii) *If $q_0\overline{b} \neq q_0$, then $\overline{a^ib^2a^{n-i}}$ is an idempotent of rank two in $M(\mathcal{A})$.*

(iii) *If $q_0\overline{b} = q_0$, then $\overline{a^i(a^\tau b)^2a^{n-i}}$ is an idempotent of rank two in $M(\mathcal{A})$.*

Definition 4.3.11. We call the following list of $2n + 2$ idempotents, if they exist, in $M(\mathcal{A})$ as the *basic idempotents*. The set of all the basic idempotents in $M(\mathcal{A})$ is denoted by $\mathfrak{B}$.

(i) The idempotent $\overline{\varepsilon}$.

(ii) The idempotent whose image set is $\{q_0\}$, denoted by $\overline{\nu}$.

(iii) For $1 \leq i \leq n$, the idempotent $\overline{a^i(a^\kappa b)a^{n-i}}$.

(iv) For $1 \leq i \leq n$, if $q_0\bar{b} \neq q_0$, then the idempotent $\overline{a^i b^2 a^{n-i}}$; else, the idempotent $\overline{a^i(a^\tau b)^2 a^{n-i}}$.

Remark 4.3.12. Clearly, $|\mathfrak{B}| \leq 2(n+1)$.

The following example shows that the cardinality of the set of basic idempotents is not necessarily $2(n+1)$.

Example 4.3.13. The automaton given in FIGURE 4.1 is a CSFA, say $\mathcal{A}$. Note that $BPI(\mathcal{A}) = \{q_0, q_2\}$. By using GAP [2012], we observed that $M(\mathcal{A})$ has no constant element. Further, we observe that $\bar{b}^2 = \overline{a^\kappa b}$, where $\kappa = 2$. Hence, $|\mathfrak{B}| < 2(n+1)$

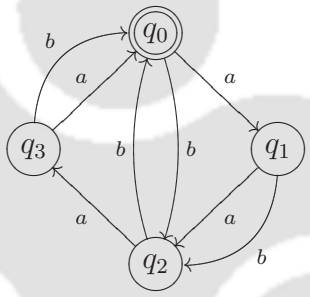

FIGURE 4.1: A semi-flower automaton with two bpis

4.3.2 Elements of rank two

In this subsection, we obtain a representation of the elements of rank two in $M(\mathcal{A})$. Here, we recall the definition of the complement of a function of rank two from [Krawetz et al., 2005].

Definition 4.3.14. Let X be a nonempty finite set and α a function on X such that $X\alpha = \{i, j\}$. The *complement* of α is the function $\alpha^\#$ defined by, for $k \in X$,

$$k\alpha^\# = \begin{cases} i & \text{if } k\alpha = j; \\ j & \text{if } k\alpha = i. \end{cases}$$

The following lemma is useful in the sequel.

Lemma 4.3.15.

(i) *If* $q_0\overline{b} \neq q_0$, *then* $\overline{b}^{\#} = \overline{b}^2$.

(ii) *If* $q_0\overline{b} = q_0$, *then* $\overline{b}^{\#} = \overline{ba^\tau b}$.

Proof. We recall that $q_m\overline{b} = q_0$ and $Q\overline{b} = \{q_0, q_m\}$. Note that, for $q \in Q$, either $q\overline{b} = q_0$ or $q\overline{b} = q_m$.

(i) Since $q_0\overline{b} \neq q_0$, we have $q_0\overline{b} = q_m$. Let q be an arbitrary element of Q. If $q\overline{b} = q_0$, then

$$q\overline{b}^2 = (q\overline{b})\overline{b} = q_0\overline{b} = q_m.$$

Else,

$$q\overline{b}^2 = (q\overline{b})\overline{b} = q_m\overline{b} = q_0.$$

Hence, $\overline{b}^{\#} = \overline{b}^2$.

(ii) Given $q_0\overline{b} = q_0$. Let q be an arbitrary element of Q. If $q\overline{b} = q_0$, then

$$q\overline{ba^\tau b} = (q\overline{b})\overline{a^\tau b} = q_0\overline{a^\tau b} = q_m$$

(cf. Lemma 4.3.8(ii)). Else,

$$q\overline{ba^\tau b} = (q\overline{b})\overline{a^\tau b} = q_m\overline{a^\tau b} = q_0.$$

Hence, $\overline{b}^{\#} = \overline{ba^\tau b}$.

□

Theorem 4.3.16. *Any element of rank two in* $M(\mathcal{A})$ *has one of the following forms.*

(β) $\overline{a^iba^j}$

(γ) $\overline{a^ib^2a^j}$

(δ) $\overline{a^i b a^\tau b a^j}$

Here, $i, j \in \{1, \ldots, n\}$.

Proof. Note that every element of rank two in $M(\mathcal{A})$ should have at least one b. Let $w = a^{i_1}ba^{i_2}b \ldots ba^{i_{k-1}}ba^{i_k} \in A^*$, for $i_t \geq 0$ ($t \in \{1, \ldots, k\}$), such that $\overline{w}$ be an arbitrary element of rank two in $M(\mathcal{A})$. Write $w = a^{i_1}buba^{i_k}$, where $u = a^{i_2}b \ldots ba^{i_{k-1}}$. Clearly, the function $\overline{bub}$ has rank two with the image set $\{q_0, q_m\}$.

Case-1 ($\overline{bub} = \overline{b}$)**:** Clearly, $\overline{w} = \overline{a^{i_1}buba^{i_k}} = \overline{a^{i_1}ba^{i_k}}$, which is in the form ($\beta$).

Case-2 ($\overline{bub} \neq \overline{b}$)**:** First we claim that $\overline{bub} = \overline{b}^{\#}$. Since $\overline{bub} \neq \overline{b}$, there exist $p \in Q$ such that $p\overline{bub} \neq p\overline{b}$. Now, we consider two subcases according to the state $p\overline{b}$.

Subcase-1 ($p\overline{b} = q_0$)**:** Since $\overline{bub} \neq \overline{b}$, we have $p\overline{bub} = q_m$. Consequently,

$$q_0\overline{ub} = q_m.$$

Let $q \in Q$ be an arbitrary element. Then, either $q\overline{b} = q_0$ or $q\overline{b} = q_m$.

If $q\overline{b} = q_0$, then

$$q\overline{bub} = (q\overline{b})\overline{ub} = q_0\overline{ub} = q_m.$$

Else ($q\overline{b} = q_m$), $q\overline{bub} = (q\overline{b})\overline{ub} = q_m\overline{ub}$. To show the last term is equal to q_0, let us assume the contrary. That is, assume $q_m\overline{ub} \neq q_0$. Then, $q_m\overline{ub} = q_m$. Consequently,

$$Q\overline{bub} = (Q\overline{b})\overline{ub} = \{q_0, q_m\}\overline{ub} = \{q_m\}.$$

This is a contradiction to $\overline{bub}$ is of rank two. Thus, if $q\overline{b} = q_m$, then $q\overline{bub} = q_0$. Hence, $\overline{bub} = \overline{b}^{\#}$.

Subcase-2 ($p\overline{b} \neq q_0$)**:** One can proceed in the similar lines as in Subcase-1 and obtain that $\overline{bub} = \overline{b}^{\#}$.

If $q_0\overline{b} \neq q_0$, then by Lemma 4.3.15(i), we have $\overline{b}^{\#} = \overline{b}^2$. Consequently,

$$\overline{w} = \overline{a^{i_1}buba^{i_k}} = \overline{a^{i_1}b^2a^{i_k}},$$

which is in the form (γ).

If $q_0\overline{b} = q_0$, then by Lemma 4.3.15(ii), we have $\overline{b}^{\#} = \overline{ba^{\tau}b}$. Consequently,

$$\overline{w} = \overline{a^{i_1}buba^{i_k}} = \overline{a^{i_1}ba^{\tau}ba^{i_k}},$$

which is in the form (δ).

□

4.3.3 Representation of $M(\mathcal{A})$

In this subsection, we give a canonical representation of the elements of $M(\mathcal{A})$ in terms of basic idempotents and circular permutation.

Theorem 4.3.17. *Every element of $M(\mathcal{A})$ can be written as a composition of a basic idempotent and a permutation, i.e.*

$$M(\mathcal{A}) = \mathfrak{B}G = \left\{ \overline{e}\,\overline{g} \;\middle|\; \overline{e} \in \mathfrak{B} \text{ and } \overline{g} \in G \right\}.$$

Proof. For $x \in A^*$, by Lemma 3.2.6, we have $\text{rank}(\overline{x}) \in \{1, 2, n\}$. If $\text{rank}(\overline{x}) = 1$, then the function $\overline{x}$ is an idempotent (being a constant function). Therefore, there exists i (with $1 \leq i \leq n$) such that

$$\overline{x} = \overline{\nu}\,\overline{a^i} \in \mathfrak{B}G.$$

If $\text{rank}(\overline{x}) = n$, then the function $\overline{x}$ is a permutation of the form $\overline{x} = \overline{a^i}$, for some i (with $1 \leq i \leq n$). Clearly, $\overline{x} \in G$ so that

$$\overline{x} = \overline{\varepsilon}\,\overline{x} \in \mathfrak{B}G.$$

If $\text{rank}(\overline{x}) = 2$, then, by Theorem 4.3.16, $\overline{x} = \overline{a^iba^j}$ or $\overline{x} = \overline{a^ib^2a^j}$ or $\overline{x} = \overline{a^iba^{\tau}ba^j}$, for some $i, j \in \{1, \ldots, n\}$.

If $\overline{x} = \overline{a^i b a^j}$, then

$$\overline{x} = \overline{a^{i-\kappa}(a^\kappa b)a^j} = \overline{a^{i-\kappa}(a^\kappa b)a^{n-(i-\kappa)}}\ \ \overline{a^{j+(i-\kappa)}} = \overline{a^{i'}(a^\kappa b)a^{n-i'}}\ \ \overline{a^{j'}},$$

where i' and j' are, respectively, the residues of $(i-\kappa)$ and $(j+i-\kappa) \mod n$. Consequently, $\overline{x} \in \mathfrak{B}G$.

If $\overline{x} = \overline{a^i b^2 a^j}$, then

$$\overline{x} = \overline{a^i b^2 a^{n-i}}\ \ \overline{a^{j-(n-i)}} = \overline{a^i b^2 a^{n-i}}\ \ \overline{a^{j'}},$$

where j' is the residue of $(j+i-n) \mod n$. Consequently, $\overline{x} \in \mathfrak{B}G$.

If $\overline{x} = \overline{a^i b a^\tau b a^j}$, then

$$\overline{x} = \overline{a^{i-\tau}(a^\tau b)^2 a^j} = \overline{a^{i-\tau}(a^\tau b)^2 a^{n-(i-\tau)}}\ \ \overline{a^{j-n+(i-\tau)}} = \overline{a^{i'}(a^\tau b)^2 a^{n-i'}}\ \ \overline{a^{j'}},$$

where i' and j' are, respectively, the residues of $(i-\tau)$ and $(j+i-\tau-n) \mod n$. Consequently, $\overline{x} \in \mathfrak{B}G$.

Thus, in all the cases the function $\overline{x} \in M(\mathcal{A})$ can be written as a composition of a basic idempotent and a permutation in G. Hence, $M(\mathcal{A}) = \mathfrak{B}G$. □

4.3.4 An example

Consider the CSFA $\mathcal{A}' = (Q, A, 1, 1, \mathcal{F})$ with $Q = \{1, 2, \ldots, n\}$, $A = \{a, b\}$, and the transitions are given in the following table.

$\mathcal{F}$	1	2	3	$\cdots$	$n-1$	n
a	2	3	4	$\cdots$	n	1
b	2	1	1	$\cdots$	1	1

Clearly, the input letters a and b induces the functions $\overline{a}$ and $\overline{b}$ on Q, respectively, given as

$$\overline{a} = \begin{pmatrix} 1 & 2 & 3 & \cdots & n-1 & n \\ 2 & 3 & 4 & \cdots & n & 1 \end{pmatrix} \text{ and } \overline{b} = \begin{pmatrix} 1 & 2 & 3 & \cdots & n-1 & n \\ 2 & 1 & 1 & \cdots & 1 & 1 \end{pmatrix}.$$

One can observe that $Q\overline{bab} = \{1\}$. Therefore, the function $\overline{bab}$ is the constant function $\overline{\nu}$ in $M(\mathcal{A}')$. Further, we observe that $\kappa = n-1$ and the functions

$$\overline{b}^2 = \begin{pmatrix} 1 & 2 & 3 & \cdots & n-1 & n \\ 1 & 2 & 2 & \cdots & 2 & 2 \end{pmatrix} \text{ and } \overline{a^\kappa b} = \begin{pmatrix} 1 & 2 & 3 & \cdots & n-1 & n \\ 1 & 2 & 1 & \cdots & 1 & 1 \end{pmatrix}$$

are idempotents of rank two in $M(\mathcal{A}')$. By Proposition 4.3.3, the functions $\overline{a^i b^2 a^{n-i}}$ and $\overline{a^i(a^{n-1}b)a^{n-i}}$ are basic idempotents of rank two in $M(\mathcal{A}')$, where $i \in \{1, 2, \ldots, n\}$. Now, we pursue on the orbits of basic idempotents of rank two. In this connection, first note that, for $1 \leq r \leq n$,

$$\overline{a^r b} = \begin{pmatrix} 1 & 2 & \cdots & n-r & n-r+1 & n-r+2 & \cdots & n-1 & n \\ 1 & 1 & \cdots & 1 & 2 & 1 & \cdots & 1 & 1 \end{pmatrix},$$

$$\overline{a^r b^2} = \begin{pmatrix} 1 & 2 & \cdots & n-r & n-r+1 & n-r+2 & \cdots & n-1 & n \\ 2 & 2 & \cdots & 2 & 1 & 2 & \cdots & 2 & 2 \end{pmatrix}.$$

For $1 \leq j < i \leq n$, let us assume that $\mathcal{O}(\overline{a^i b^2 a^{n-i}}) \cap \mathcal{O}(\overline{a^j b^2 a^{n-j}}) \neq \varnothing$. Then, for some t (with $1 \leq t \leq n$),

$$\overline{a^i b^2 a^{n-i}} = \overline{a^j b^2 a^{n-j}}\ \ \overline{a^t} \Longrightarrow \overline{a^{i-j} b^2} = \overline{b^2 a^{i-j+t}}.$$

If $i - j + t \neq 0 (\text{mod } n)$, then $Q\overline{b^2 a^{i-j+t}} \neq \{1, 2\} = Q\overline{a^{i-j} b^2}$; a contradiction. Otherwise, we have $\overline{a^{i-j} b^2} = \overline{b^2}$. But, from the above shown $\overline{b}^2$ and $\overline{a^r b^2}$, we can observe that $\overline{a^{i-j} b^2} \neq \overline{b^2}$. Hence, for $1 \leq j < i \leq n$, we have

$$\mathcal{O}(\overline{a^i b^2 a^{n-i}}) \cap \mathcal{O}(\overline{a^j b^2 a^{n-j}}) = \varnothing.$$

Similarly, we can prove that, for $1 \leq j < i \leq n$, we have

$$\mathcal{O}(\overline{a^i(a^{n-1}b)a^{n-i}}) \cap \mathcal{O}(\overline{a^j(a^{n-1}b)a^{n-j}}) = \varnothing.$$

Note that $\overline{b}^2 \neq \overline{a^{n-1}b}$. Now, for $1 \leq j < i \leq n$, let us assume that

$$\mathcal{O}(\overline{a^i b^2 a^{n-i}}) \cap \mathcal{O}(\overline{a^j(a^{n-1}b)a^{n-j}}) \neq \varnothing.$$

Then, for some t (with $1 \leq t \leq n$), we have

$$\overline{a^ib^2a^{n-i}} = \overline{a^j(a^{n-1}b)a^{n-j}}\ \ \overline{a^t} \Longrightarrow \overline{a^{i-j}b^2} = \overline{(a^{n-1}b)a^{i-j+t}}.$$

If $i-j+t \neq 0(\mod\ n)$, then $Q\overline{b^2a^{i-j+t}} \neq \{1,2\} = Q\overline{a^{i-j}b^2} = \{1,2\}$; a contradiction. Otherwise, we have $\overline{a^{i-j}b^2} = \overline{a^{n-1}b}$. But, from the above shown $\overline{a^{n-1}b}$ and $\overline{a^rb^2}$, we can observe that $\overline{a^{i-j}b^2} \neq \overline{a^{n-1}b}$. Hence, for $1 \leq j < i \leq n$, we have

$$\mathcal{O}(\overline{a^ib^2a^{n-i}}) \cap \mathcal{O}(\overline{a^j(a^{n-1}b)a^{n-j}}) = \varnothing.$$

Thus, all the orbits of the basic idempotents of rank two are disjoint and so all the basic idempotents of rank two are distinct. Thus, $|\mathfrak{B}| = 2(n+1)$. Consequently, $M(\mathcal{A}') = \mathfrak{B}G = 2n(n+1)$. Hence, the syntactic complexity of the submonoid accepted by the CSFA $\mathcal{A}'$ is $2n(n+1)$.

4.3.5 Proof of Theorem 4.3.2

Now, we prove the main Theorem 4.3.2. We know that

$$\begin{aligned} M(\mathcal{A}) &= \bigcup_{\overline{x} \in M(\mathcal{A})} \mathcal{O}(\overline{x}) \\ &= \bigcup_{\overline{x} \in \mathfrak{B}G} \mathcal{O}(\overline{x}) \text{ by using Theorem 4.3.17} \\ &= \bigcup_{\overline{x} \in \mathfrak{B}} \mathcal{O}(\overline{x}). \end{aligned}$$

This implies that

$$\begin{aligned} |M(\mathcal{A})| &\leq |\mathfrak{B}||\mathcal{O}(\overline{x})| \\ &\leq 2n(n+1) \text{ by using Proposition 4.1.6 and Remark 4.3.12.} \end{aligned}$$

Thus, the sizes of syntactic monoids of the submonoids accepted by CSFA with two bpis over a binary alphabet is bounded by $2n(n+1)$, where n is the state complexity of the CSFA. For the class of automata displayed in Subsection 4.3.4, the syntactic monoid size is exactly $2n(n+1)$. Hence, the syntactic complexity of the class of submonoids accepted by CSFA with two bpis over a binary alphabet is $2n(n+1)$.

4.4 Conclusion

This work investigates the syntactic complexity of the various classes of the submonoids accepted by CSFA, classified by their number of bpis. In fact, we showed that the syntactic complexity of the submonoids accepted by CSFA with at most one bpi is linear. Further, we proved that the syntactic complexity of the class of the submonoids accepted by CSFA with two bpis over a binary alphabet is $2n(n+1)$. In that connection, we obtained a representation for the functions of rank two in the monoid of CSFA with two bpis over a binary alphabet. However, there is a lot more to investigate the syntactic complexity concerning the finitely generated submonoids of a free monoid. For instance, one can target to address the syntactic complexity of the class of the submonoids accepted by CSFA with two bpis over an arbitrary alphabet. In general, one can study the syntactic complexity of the class of the submonoids accepted by CSFA and SFA with more than two bpis.

5

L-Primitive Words

The concept of primitive words plays an important role in the algebraic theory of languages. Ito et al. [1988] have studied the primitive words in the languages of automata, in general. In this chapter, we focus on investigating the primitive words in the languages of semi-flower automata. In fact, we extend our study on the primitive words in the submonoids of free monoids. We could quickly ascertain that the number of primitive words in a submonoid of a free monoid is either at most one or infinity. Then, we proceed to consider L-primitive words, a generalized notion of primitive words, introduced by Krishna [2011]. Here, we study the distribution of L-primitive words in certain subsets of free monoids. In particular, we target to count the L-primitive words in the submonoids of free monoids.

5.1 Primitive words in submonoids

In this section, we quickly recall some necessary results on primitive words from Shallit [2008]. Further, we investigate the number of primitive words in the submonoids of a free monoid.

Definition 5.1.1. A nonempty word $x \in A^*$ is said to be a *primitive word* if x is not a power of any other word in A^*, i.e. for $u \in A^*$,

$$x = u^k \implies k = 1.$$

Let X be a subset of A^*. The set of primitive words in X is denoted by X_p.

Theorem 5.1.2. *Every nonempty word can be uniquely expressed as a power of a primitive word.*

Definition 5.1.3. Let $w \in A^*$ be a nonempty word. The unique primitive word $x \in A^*$ such that $w = x^k$, for some integer $k \geq 1$, is called the *primitive root* of w, and it is denoted by $\sqrt{w}$.

Definition 5.1.4. Let L be a subset of A^*. The *root* of L, denoted by $\sqrt{L}$, is defined as $\sqrt{L} = \{\sqrt{w} \in A_p^* \mid w \in L \setminus \{\varepsilon\}\}$.

Definition 5.1.5. Let L be a subset of A^*. The subset L is said to be *commutative* if $uv = vu$, for all $u, v \in L$.

Remark 5.1.6. Let L be a subset of A^*. The subset L is commutative if and only if there exists $w \in A^*$ such that $L \subseteq \{w\}^*$.

Theorem 5.1.7 (Shyr and Tseng [1984])**.** *Let H be a submonoid of A^*. H is noncommutative if and only if $|H_p| = \infty$.*

Now, we observe that a submonoid of A^* contains either at most one primitive word or infinitely many primitive words.

Theorem 5.1.8. *Let H be a submonoid of A^*; then either $|H_p| \leq 1$ or $|H_p| = \infty$.*

Proof. If $H = \{\varepsilon\}$, then $|H_p| = 0$. Let us assume that $H \neq \{\varepsilon\}$. If H is noncommutative, then by Theorem 5.1.7, we have $|H_p| = \infty$. Otherwise, by Remark 5.1.6, we have $H \subseteq \{w\}^*$, for some nonempty word $w \in A^*$. Without loss of generality, assume that $w \in A_p^*$. Thus, according to $w \in H$ or not, we have $|H_p| = 1$ or 0. □

Corollary 5.1.9. *If $H(\neq \{\varepsilon\})$ is a submonoid of A^*, then either $|\sqrt{H}| = 1$ or ∞.*

5.2 L-primitive words

The notion of L-primitive words has been introduced by Krishna [2011] as a generalization of the classical definition of primitive words. In this section, we recall the definitions and some basic properties related to L-primitive words from [Krishna, 2011]. In what follows, L always denotes a subset of A^*.

Definition 5.2.1. A nonempty word $x \in A^*$ is said to be an *L-primitive word* if x is not a power of any other word in L, i.e. for $u \in L$,

$$x = u^k \implies k = 1.$$

Notation 5.2.2. Let X be a subset of A^*.

(i) The set of L-primitive words in X is denoted by $X_{L\text{-}p}$.

(ii) The set of L-primitive words in X^*, i.e. $(X^*)_{L\text{-}p}$, is simply denoted by $X^*_{L\text{-}p}$.

(iii) The set of L-primitive words in X^c, the complement of X in A^*, is simply denoted by $X^c_{L\text{-}p}$.

Remark 5.2.3.

(i) If $L = \varnothing$, then $A^*_{L\text{-}p} = A^+$, the set of all nonempty words over A.

(ii) If $L = A^*$, then $A^*_{L\text{-}p} = A^*_p$, the set of all primitive words over A.

Proposition 5.2.4. *If L_1 and L_2 are two subsets of A^*, then*

$$L_1 \subseteq L_2 \implies A^*_{L_2\text{-}p} \subseteq A^*_{L_1\text{-}p}.$$

Proof. On the contrary, let us assume that $A^*_{L_2\text{-}p} \nsubseteq A^*_{L_1\text{-}p}$. Then there exists $w \in A^*_{L_2\text{-}p}$, but $w \notin A^*_{L_1\text{-}p}$. Since $w \notin A^*_{L_1\text{-}p}$, there exists $u \in L_1$ such that $w = u^k$, for some $k > 1$. In view of hypothesis, we have $u \in L_2$. Consequently, $w \notin A^*_{L_2\text{-}p}$; a contradiction. □

Corollary 5.2.5. *Every primitive word is an L-primitive word.*

Corollary 5.2.6. *If $|A| \geq 2$, then $|A^*_{L\text{-}p}| = \infty$.*

Remark 5.2.7. An L-primitive word is not necessarily a primitive word. For instance, let $L = \{abab\} \subseteq \{a, b\}^*$. Clearly, the word $abab$ is an L-primitive word, but not a primitive word.

Definition 5.2.8. Let $w \in A^*$ be a nonempty word. The set of *L-primitive roots* of w, denoted by $\sqrt[L]{w}$, is defined as

$$\sqrt[L]{w} = \{x \in A^*_{L\text{-}p} \mid x^k = w, \text{ for some } k \geq 1\}.$$

Remark 5.2.9. The primitive root of a nonempty word is an L-primitive root of the word. Thus, if $w \neq \varepsilon$, then $\sqrt[L]{w} \neq \varnothing$.

Definition 5.2.10. Let X be a subset of A^*. The *L-primitive root* of X, denoted by $\sqrt[L]{X}$, is defined as

$$\sqrt[L]{X} = \bigcup_{w \in X \setminus \{\varepsilon\}} \sqrt[L]{w}.$$

5.3 L-primitive words in L

In this section, we make an attempt to investigate L-primitive words in L and also in L^c. In this connection, we provide some sufficient conditions and characterizations. In fact, we give relation between L-primitive words and L-primitive roots in L.

Theorem 5.3.1. *If $\varepsilon \notin L$, then $L \neq \varnothing$ if and only if $L_{L\text{-}p} \neq \varnothing$.*

Proof. Let us assume that $L \neq \varnothing$ and choose $w \in L$. If $w \in L_{L\text{-}p}$, then we are through. Otherwise, there exists $u \in L$ such that $w = u^k$, for some $k > 1$. Clearly, $|u| < |w|$. If $u \in L_{L\text{-}p}$, then we are through. Otherwise, we continue to choose shorter words in L whose power is w. But this process terminates at a finite stage and eventually we get a word $x \in L_{L\text{-}p}$ and $w = x^m$, for some $m > 1$. Hence, $L_{L\text{-}p} \neq \varnothing$. The converse is straightforward. □

Theorem 5.3.2. *If $L \subseteq A^*$ is a prefix set such that $\varepsilon \notin L$, then $L = L_{L\text{-}p}$.*

Proof. Clearly, $L_{L\text{-}p} \subseteq L$. Let $x \in L$, but $x \notin L_{L\text{-}p}$. There exists a word $u \in L$ such that $x = u^k$, for some $k > 1$. Thus, the word $u \in L$ is a prefix of the word $x \in L$. This contradicts that L is a prefix set. Hence, $L = L_{L\text{-}p}$. □

Remark 5.3.3. The converse of Theorem 5.3.2 is not necessarily true. For instance, let $L = \{a, ab\} \subseteq \{a, b\}^*$. Clearly, $L = L_{L\text{-}p}$, but L is not a prefix set.

It is clear that $L_{L\text{-}p} \subseteq \sqrt[L]{L}$. Now, we explore the possibilities so that $L_{L\text{-}p} = \sqrt[L]{L}$. For this we need the notion of power of a subset of A^* introduced by Calbrix and Nivat [1996].

Definition 5.3.4. Let X be a subset of A^*. The *power* of X, denoted by $\mathrm{pow}(X)$, is defined as

$$\mathrm{pow}(X) = \{x^i \mid x \in X \text{ and } i \geq 1\}.$$

Remark 5.3.5. Clearly, $\mathrm{pow}(A^*_{L\text{-}p}) = A^+$.

Theorem 5.3.6.

(i) $\sqrt[L]{L} \subseteq L \iff L_{L\text{-}p} = \sqrt[L]{L}$.

(ii) $L^c = \mathrm{pow}(L^c) \implies L_{L\text{-}p} = \sqrt[L]{L}$.

(iii) $L \subseteq A^*_p \implies L_{L\text{-}p} = \sqrt[L]{L} = L$.

Proof. We first note that $L_{L\text{-}p} \subseteq \sqrt[L]{L}$.

(i) ($:\Longleftarrow$) Since $L_{L\text{-}p} \subseteq L$, from the hypothesis, we have $\sqrt[L]{L} \subseteq L$.

($\Longrightarrow:$) Let $x \in \sqrt[L]{L}$; then x is L-primitive word. Also, from the hypothesis, we have $x \in L$. Thus, $x \in L_{L\text{-}p}$. Hence, we have the part (i).

(ii) Let us assume that $x \in \sqrt[L]{L} \setminus L_{L\text{-}p}$. Since x is an L-primitive word and $x \notin L_{L\text{-}p}$, we have $x \notin L$. Then, from the hypothesis, we have $x^k \in L^c$, for all $k \geq 1$. But, since $x \in \sqrt[L]{L}$, we have $x \in \sqrt[L]{w}$, for some $w \in L$. That is, there is a number $t \geq 1$, such that $x^t = w (\in L)$; a contradiction. Hence, $\sqrt[L]{L} = L_{L\text{-}p}$.

(iii) Clearly, $L_{L\text{-}p} \subseteq L$. Let $x \in L$; from the hypothesis, we have $x \in A_p^*$. By Corollary 5.2.5, since every primitive word is an L-primitive word, we have $x \in L_{L\text{-}p}$. Thus, $L = L_{L\text{-}p}$.

It is clear that for $w \in A_p^*$, we have $\sqrt[L]{w} = \{w\}$. Since $L \subseteq A_p^*$, we have

$$\sqrt[L]{L} = \bigcup_{w \in L} \sqrt[L]{w} = \bigcup_{w \in L} \{w\} = L.$$

Hence, if $L \subseteq A_p^*$, we have $L_{L\text{-}p} = \sqrt[L]{L} = L$.

□

Corollary 5.3.7. $L = \sqrt[L]{L} \iff L_{L\text{-}p} = \sqrt[L]{L} = L$.

Remark 5.3.8. The converse of Theorem 5.3.6(ii) is not necessarily true. For instance, consider $L = \{a, b, a^6\} \subseteq \{a, b\}^*$. Observe that $L_{L\text{-}p} = \sqrt[L]{L} = \{a, b\}$. Clearly, since $a^2 \in L^c$, we have $a^6 \in \text{pow}(L^c)$; but, $a^6 \notin L^c$. Hence, $L^c \neq \text{pow}(L^c)$.

Theorem 5.3.9. $L = \text{pow}(L) \iff L^c_{L\text{-}p} = L^c$.

Proof.

($\Longrightarrow$:) Clearly, $L^c_{L\text{-}p} \subseteq L^c$. Let $x \in L^c$, but $x \notin L^c_{L\text{-}p}$. There exists a word $y \in L$ such that $x = y^k$, for some $k > 1$. Since $y \in L$, we have $y^k \in \text{pow}(L)$. It follows that $x \in \text{pow}(L)$. But, $L = \text{pow}(L)$, we have $x \in L$. This is a contradiction.

($:\Longleftarrow$) Clearly, $L \subseteq \text{pow}(L)$. Let $x \in \text{pow}(L)$, but $x \notin L$. There exists a word $y \in L$ such that $x = y^k$, for some $k > 1$. Since, $x \notin L$, we have $x \in L^c$. But, $L^c_{L\text{-}p} = L^c$, it follows that x is an L-primitive word; which is a contradiction.

□

5.4 L-primitive words in submonoids

In this section, we study the L-primitive words in the submonoids of a free monoid. We first count the L-primitive words in a submonoid of the free monoid over a unary alphabet. In this case, when L is finite, we prove that a submonoid has either at most one or infinitely many L-primitive words. Then, finally we leave certain remarks on estimating the number of L-primitive words over an arbitrary alphabet.

Let $A = \{a\}$ be a unary alphabet. It is well known that A^* is isomorphic to the additive monoid of natural numbers $\mathbb{N} = \{0, 1, 2, \ldots\}$ under the isomorphism given by $a^k \mapsto k$. Thus, each word a^k of A^* is characterized by its length $k \in \mathbb{N}$. Hence, we count the L-primitive words in the submonoids of $\mathbb{N}$, instead of A^*. In what follows, $H(\neq \{0\})$ denotes a submonoid of $\mathbb{N}$ and L a nonempty subset of $\mathbb{N}$.

To count L-primitive words in H, we use some properties of numerical monoids. In the following, first we present some necessary results on numerical monoids from [Rosales and García-Sánchez, 2009].

Definition 5.4.1. A *numerical monoid* is a submonoid of $\mathbb{N}$ whose complement in $\mathbb{N}$ is finite.

Theorem 5.4.2. *Every numerical monoid admits a unique finite minimal system of generators.*

Theorem 5.4.3. *Let X be a nonempty subset of $\mathbb{N}$. The submonoid generated by X, in $\mathbb{N}$ is a numerical monoid if and only if* $\gcd(X) = 1$.

Theorem 5.4.4. *Any nontrivial submonoid of* $\mathbb{N}$ *is isomorphic to a numerical monoid.*

Now, we count the number of L-primitive words in H. We begin with the following remark.

Remark 5.4.5. If $1 \in L$, then according to $1 \in H$ or not, we have $|H_{L\text{-}p}| = 1$ or $|H_{L\text{-}p}| = 0$, respectively.

Let us assume that $1 \notin L$. In view of Theorem 5.4.4 and Theorem 5.4.2, let Y be the finite minimal generating set of H.

Theorem 5.4.6. *If* $\gcd(Y) = 1$, *then* $|H_{L\text{-}p}| = \infty$.

Proof. If $\gcd(Y) = 1$, by Proposition 5.4.3, the submonoid H is a numerical monoid so that $|\mathbb{N} \setminus H| < \infty$. Thus, H contains infinitely many prime numbers. Since $1 \notin L$, every prime number is L-primitive. Hence, $|H_{L\text{-}p}| = \infty$. □

Theorem 5.4.7. *If* L *is a finite set and* $\gcd(Y) > 1$, *then* $|H_{L\text{-}p}| \leq 1$ *or* $|H_{L\text{-}p}| = \infty$.

Proof. We first assume that $l \nmid d$, for all $l \in L$ and claim that $|H_{L\text{-}p}| = \infty$. Let $\gcd(Y) = d$. Since $d \neq 1$, by Proposition 5.4.3, the submonoid H is not a numerical monoid. We define the function

$$f : H \longrightarrow \mathbb{N} \quad \text{by} \quad hf = \frac{h}{d}.$$

Clearly, f is a monomorphism and therefore the image of f, $\mathrm{Im}(f)$, is isomorphic to H. By Theorem 5.4.3, the submonoid $\mathrm{Im}(f)$ is a numerical monoid.

Clearly, $\mathrm{Im}(f)$ has infinitely many prime numbers. Let $p \in \mathrm{Im}(f)$ be a prime number such that $p > \max(L)$, then $pd \in H$. By Euclid's lemma, $l \nmid pd$, for all $l \in L$. Since $\mathrm{Im}(f)$ has infinitely many such prime numbers, we have $|H_{L\text{-}p}| = \infty$.

Now, we assume that $l \mid d$, for some $l \in L$. Here, we determine $|H_{L\text{-}p}|$ with respect to $d \in L$ or not. If $d \notin L$, then clearly $|H_{L\text{-}p}| = 0$. If $d \in L$, we consider the cases $d \in H$ or not. If $d \notin H$, then clearly $|H_{L\text{-}p}| = 0$. In case $d \in H$, if there is an $l'(\neq d)$ which divides d, then $|H_{L\text{-}p}| = 0$; otherwise $|H_{L\text{-}p}| = 1$. □

Remark 5.4.8. If L is an infinite subset of $\mathbb{N}$, then $|H_{L\text{-}p}|$ need not satisfy the Theorem 5.4.7. For instance, let H be the submonoid of $\mathbb{N}$ generated by the set $\{4, 6\}$ and $L = \{4\} \cup \{\mathfrak{P} \setminus \{2, 5\}\}$, where $\mathfrak{P}$ is the set of all prime numbers in $\mathbb{N}$. We observe that $H_{L\text{-}p} = \{4, 10\}$ and so $|H_{L\text{-}p}| = 2$. Similarly, if $L = \{4\} \cup \{\mathfrak{P} \setminus \{2, 5, 7\}\}$, then $H_{L\text{-}p} = \{4, 10, 14\}$ and so $|H_{L\text{-}p}| = 3$.

In the following, we make certain remarks on the number of L-primitive words in the submonoids of a free monoid over an alphabet of size at least two. First observe that if the submonoid H is $\{\varepsilon\}$, then $|H_{L\text{-}p}| = 0$. If $H \neq \{\varepsilon\}$, then by Corollary 5.2.5, we have the following remark.

Remark 5.4.9. If H is a noncommutative submonoid of A^*, where $|A| \geq 2$, then $|H_{L\text{-}p}| = \infty$.

5.5 Conclusion

In this chapter, we have considered a study on the number of primitive words in the languages of SFA. In fact, we extended the study to submonoids of free monoids and observed that the number is either at most one or infinite. Also, we have considered to study the number of L-primitive words in submonoids of free monoids. If L is a finite, we have counted the number of L-primitive words in the submonoids of a free monoid over a unary alphabet. When L is infinite, the problem appears to be more complicated and a systematic study in this regard is necessary. In case the alphabet size is at least two, we could remark only on the number of L-primitive words in noncommutative submonoids. One can consider the problem in commutative submonoids.

Bibliography

Beaudry, M. and Holzer, M.: 2011, On the size of inverse semigroups given by generators, *Theoret. Comput. Sci.* **412**(8-10), 765–772.

Berstel, J. and Perrin, D.: 1985, *Theory of codes*, Vol. 117 of *Pure and Applied Mathematics*, Academic Press Inc.

Berstel, J., Perrin, D. and Reutenauer, C.: 2010, *Codes and automata*, Vol. 129 of *Encyclopedia of Mathematics and its Applications*, Cambridge University Press, Cambridge.

Brzozowski, J. A. and Li, B.: 2012, Syntactic complexities of some classes of star-free languages, *DCFS*, pp. 117–129.

Brzozowski, J. A. and Liu, D.: 2012, Syntactic complexity of finite/cofinite, definite, and reverse definite languages, *CoRR* **abs/1203.2873**.

Brzozowski, J., Li, B. and Ye, Y.: 2012, Syntactic complexity of prefix-, suffix-, bifix-, and factor-free regular languages, *Theoret. Comput. Sci.* **449**, 37–53.

Brzozowski, J. and Ye, Y.: 2011, Syntactic complexity of ideal and closed languages, *Developments in language theory*, Vol. 6795 of *Lecture Notes in Comput. Sci.*, Springer, Heidelberg, pp. 117–128.

Calbrix, H. and Nivat, M.: 1996, Prefix and period languages of rational ω-languages, *Developments in language theory, II (Magdeburg, 1995)*, World Sci. Publ., River Edge, NJ, pp. 341–349.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C.: 2001, *Introduction to algorithms*, second edn, MIT Press.

Czeizler, E., Kari, L. and Seki, S.: 2010, On a special class of primitive words, *Theoret. Comput. Sci.* **411**(3), 617–630.

Domaratzki, M.: 2004, *Trajectory-based operations*, PhD thesis, Queen's University, Canada.

Dömösi, P., Horváth, S., Ito, M., Kászonyi, L. and Katsura, M.: 1993, Formal languages consisting of primitive words, *Fundamentals of computation theory (Szeged, 1993)*, Vol. 710 of *Lecture Notes in Comput. Sci.*, Springer, Berlin, pp. 194–203.

Dömösi, P. and Nehaniv, C. L.: 2005, *Algebraic theory of automata networks*, Vol. 11 of *SIAM Monographs on Discrete Mathematics and Applications*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Dubuc, L.: 1998, Sur les automates circulaires et la conjecture de Černý, *RAIRO Inform. Théor. Appl.* **32**(1-3), 21–34.

Dummit, D. S. and Foote, R. M.: 2004, *Abstract algebra*, third edn, John Wiley & Sons Inc., Hoboken, NJ.

Egri-Nagy, A.: 2005, *Algebraic hierarchical decompositions of finite state automata a computational approach*, PhD thesis, University of Hertfordshire.

Egri-Nagy, A. and Nehaniv, C. L.: 2005, Cycle structure in automata and the holonomy decomposition, *Acta Cybernet.* **17**(2), 199–211.

Egri-Nagy, A. and Nehaniv, C. L.: 2010, *SgpDec software package for hierarchical coordinatization of groups and semigroups, implemented in the GAP computer algebra system*, Version 0.5.38, http://sgpdec.sf.net.

Eilenberg, S.: 1976, *Automata, languages, and machines. Vol. B*, Academic Press, New York.

Friedman, J.: 2011a, Linear algebra and Hanna Neumann conjecture. Preprint.

Friedman, J.: 2011b, Sheaves on graphs, their homological invariants, and a proof of the Hanna Neumann conjecture. Preprint.
URL: *arXiv:1105.0129v2*

GAP: 2012, *GAP – Groups, algorithms, and programming, Version 4.5.4.*
URL: *http://www.gap-system.org*

Giambruno, L.: 2007, *Automata-theoretic methods in free monoids and free groups*, PhD thesis, Universit degli Studi di Palermo, Palermo, Italy.

Giambruno, L. and Restivo, A.: 2008, An automata-theoretic approach to the study of the intersection of two submonoids of a free monoid, *Theor. Inform. Appl.* **42**(3), 503–524.

Holcombe, M.: 1980, Holonomy decompositions of near-rings, *Proc. Edinburgh Math. Soc. (2)* **23**(1), 43–47.

Holzer, M. and König, B.: 2004, On deterministic finite automata and syntactic monoid size, *Theoret. Comput. Sci.* **327**(3), 319–347.

Howson, A. G.: 1954, On the intersection of finitely generated free groups, *J. London Math. Soc.* **29**, 428–434.

Hsiao, H. K., Huang, C. C. and Yu, S. S.: 2002, Word operation closure and primitivity of languages, *J.UCS* **8**(2), 243–256 (electronic). Advances and trends in automata and formal languages.

Ito, M.: 2004, *Algebraic theory of automata and languages*, World Scientific Publishing Co. Inc., River Edge, NJ.

Ito, M., Katsura, M., Shyr, H. J. and Yu, S. S.: 1988, Automata accepting primitive words, *Semigroup Forum* **37**(1), 45–52.

Karhumäki, J.: 1984, A note on intersections of free submonoids of a free monoid, *Semigroup Forum* **29**(1-2), 183–205.

Kari, L. and Thierrin, G.: 1998, Word insertions and primitivity, *Util. Math.* **53**, 49–61.

Krawetz, B., Lawrence, J. and Shallit, J.: 2005, State complexity and the monoid of transformations of a finite set, *Internat. J. Found. Comput. Sci.* **16**(3), 547–563.

Krishna, K. V.: 2011, L-primitive words. Preprint.

Krishna, K. V. and Chatterjee, N.: 2007, Holonomy decomposition of seminearrings, *Southeast Asian Bull. Math.* **31**(6), 1113–1122.

Krohn, K. and Rhodes, J.: 1965, Algebraic theory of machines. I. Prime decomposition theorem for finite semigroups and machines, *Trans. Amer. Math. Soc.* **116**, 450–464.

Lawson, M. V.: 2004, *Finite automata*, Chapman & Hall/CRC, Boca Raton, FL.

Lyndon, R. C. and Schützenberger, M. P.: 1962, The equation $a^M = b^N c^P$ in a free group, *Michigan Math. J.* **9**, 289–298.

Meakin, J. and Weil, P.: 2002, Subgroups of free groups: a contribution to the Hanna Neumann conjecture, *Geom. Dedicata* **94**, 33–43.

Mineyev, I.: 2011, Groups, graphs, and the Hanna Neumann conjecture. Preprint.

Mineyev, I.: 2012, Submultiplicativity and the Hanna Neumann conjecture, *Ann. of Math.* **175**(1), 393–414.

Neumann, H.: 1956, On the intersection of finitely generated subgroups of free groups, *Publ. Math. Debrecen* **4**, 186–189.

Neumann, W. D.: 1990, On intersections of finitely generated subgroups of free groups, *Groups—Canberra 1989*, Vol. 1456 of *Lecture Notes in Math.*, Springer, Berlin, pp. 161–170.

Pin, J.-E.: 1978, Sur un cas particulier de la conjecture de Cerny, *Automata, languages and programming (Fifth Internat. Colloq., Udine, 1978)*, Vol. 62 of *Lecture Notes in Comput. Sci.*, Springer, Berlin, pp. 345–352.

Pin, J.-E.: 1986, *Varieties of formal languages*, Foundations of Computer Science, Plenum Publishing Corp., New York.

Rosales, J. C. and García-Sánchez, P. A.: 2009, *Numerical semigroups*, Vol. 20 of *Developments in Mathematics*, Springer, New York.

Shallit, J. O.: 2008, *A Second Course in Formal Languages and Automata Theory*, Cambridge University Press.

Shyr, H. J. and Tseng, D. C.: 1984, Some properties of dense languages, *Soochow J. Math.* **10**, 127–131.

Tilson, B.: 1972, The intersection of free submonoids of a free monoid is free, *Semigroup Forum* **4**, 345–350.

Index

Bio-Data

Full Name	:	Shubh Narayan Singh
Education		
July, 2007 – Till Date	:	Research Scholar, Department of Mathematics Indian Institute of Technology Guwahati, Guwahati.
July, 2005 – May, 2007	:	M. Sc. in Mathematics University of Allahabad, Allahabad.
July, 2002 – May, 2005	:	B. Sc. with Mathematics, Physics, & Chemistry University of Allahabad, Allahabad.
Teaching Experience		
July, 2007 – May 2012	:	Teaching Assistant, Department of Mathematics IIT Guwahati, Guwahati.

Publications

1. *The Rank and Hanna Neumann Property of Some Submonoids of a Free Monoid*, Ann. Math. Inform., To appear.

2. *A Sufficient Condition for Hanna Neumann Property of Submonoids of a Free Monoid*, Semigroup Forum, To appear.

3. *Holonomy Decomposition of Circular Semi-Flower Automata*, Communicated.

4. *Syntactic Complexity of Circular Semi-Flower Automata*, Communicated.

5. *L-Primitive Words in Submonoids*, Communicated.

   All the above five articles are coauthored with K. V. Krishna.

Conferences/Workshops:

1. *On the Rank of the Intersection of Two Submonoids of a Free Monoid* , International Conference A^3: Abstract Algebra and Algorithms Conference, Eger, Hungary, August 14-17, 2011.

2. Research Promotion Workshop on Introduction to Graph and Geometric Algorithms, IIT Guwahati, India, October 21-23, 2011.

3. International workshop cum conference on Groups, Actions, Computations (GAC 2010), HRI Allahabad, India, September 1-12, 2010.