GRAPH AUTOMATIC SEMIGROUPS

Rachael Marie Carey

**A Thesis Submitted for the Degree of PhD
at the
University of St Andrews**

2016

Graph Automatic Semigroups

Rachael Marie Carey

University of St Andrews

This thesis is submitted in partial fulfilment for the degree of PhD
at the University of St Andrews

April 4, 2016

Abstract

In this thesis we examine properties and constructions of graph automatic semigroups, a generalisation of both automatic semigroups and finitely generated FA-presentable semigroups. We consider the properties of graph automatic semigroups, showing that they are independent of the choice of generating set, have decidable word problem, and that if we have a graph automatic structure for a semigroup then we can find one with uniqueness. Semigroup constructions and their effect on graph automaticity are considered. We show that finitely generated direct products, free products, finitely generated Rees matrix semigroup constructions, zero unions, and ordinal sums all preserve unary graph automaticity, and examine when the converse also holds. We also demonstrate situations where semidirect products, Bruck-Reilly extensions, and semilattice constructions preserve graph automaticity, and consider the conditions we may impose on such constructions in order to ensure that graph automaticity is preserved.

Unary graph automatic semigroups, that is semigroups which have graph automatic structures over a single letter alphabet, are also examined. We consider the form of an automaton recognising multiplication by generators in such a semigroup, and use this to demonstrate various properties of unary graph automatic semigroups. We show that infinite periodic semigroups are not unary graph automatic, and show that we may always find a uniform set of normal forms for a unary graph automatic semigroup. We also determine some necessary conditions for a semigroup to be unary graph automatic, and use this to provide examples of semigroups which are not unary graph automatic. Finally we consider semigroup constructions for unary graph automatic semigroups. We show that the free product of two semigroups is unary graph automatic if and only if both semigroups are trivial; that direct products do not always preserve unary graph automaticity; and that Bruck-Reilly extensions are never unary graph automatic.

Candidate's Declarations

I, Rachael Carey, hereby certify that this thesis, which is approximately 36,000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for a higher degree.

I was admitted as a research student in September 2011 and as a candidate for the degree of Ph.D in September 2012; the higher study for which this is a record was carried out in the University of St Andrews between 2011 and 2015.

Date:

Signature of candidate:

Supervisor's Declaration

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of Ph.D in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date:

Signature of supervisor:

Permission for Publication

Acknowledgements

My time in St Andrews would not have been possible, nor as enjoyable, without many people. I am grateful for the help, support, and friendship of all those who have had a part in my academic life.

In particular, thank you to my supervisor Nik for all his help and guidance throughout the course of my PhD, as well as for being an inspirational teacher.

Thank you to everyone in the maths department for making the last eight years so enjoyable. I am particularly grateful to the wonderful maths postgrad community for their friendship. Thanks for all the crosswords, the quizzes, the board games, and the pub lunches. In particular, thanks to the Ginger Office and everyone who has been a part of it during the last four years for providing a great place to work and plenty of cake. Particular thanks also to Jenny for being a fantastic flatmate for the last three years.

I am so grateful to my family for all their support throughout my studies in St Andrews, in particular to my parents, grandparents, and sister Katie. Everything from our first visit here nine years ago to today would not have been possible without them.

Most of all, thank you to Jonathan for all his love and support. Thank you for believing in me even when I didn't believe in myself, for making me see that it's okay to ask for help when I need it, and for always making me laugh. I could not have done this without you.

Contents

1 Introduction and Background **1**

1.1 Languages and Automata 3

1.1.1 Regular Languages 5

1.1.2 Finite State Automata 7

1.2 Semigroups 8

2 Graph Automatic Semigroups: An Introduction **11**

2.1 Definition and Examples 11

2.2 Concepts Related to Graph Automaticity 15

2.2.1 Automaticity 15

2.2.2 FA-Presentability 17

2.3 Further Examples 20

2.4 Properties of Graph Automatic Semigroups 25

2.4.1 Generating Sets 25

2.4.2 Preimages 27

2.4.3 Structures with Uniqueness 28

2.4.4 The Word Problem for Monoids 30

3 Subsemigroups and Extensions **33**

3.1 Subsemigroups and Ideals 33

3.2 Zeros and Identities 37

3.2.1 The Word Problem for Semigroups 41

3.3 Large Subsemigroups and Small Extensions 42

4 Constructions for Graph Automatic Semigroups: Products 49

4.1 Free Products 49

4.2 Semidirect Products 53

4.3 Direct Products 58

4.4 Bruck-Reilly Extensions 64

5 Constructions for Graph Automatic Semigroups: Unions 73

5.1 Zero Unions 73

5.2 Ordinal Sums 75

5.3 Rees Matrix Semigroups 77

5.4 Semilattices of Semigroups 82

6 Unary Graph Automatic Semigroups 91

6.1 Definition and Examples 91

6.2 Unary Automatic Semigroups and Unary
FA-Presentable Semigroups 95

6.3 Automata for Unary Graph Automatic
Structures 96

6.3.1 Examples of Automata 106

6.4 Periodicity 109

6.5 Proving Non-Unary Graph Automaticity 112

6.6 Disjoint unions of the free monogenic semigroup 120

6.7 Monogenic Subsemigroups 126

6.8 Normal Forms 127

7 Constructions for Unary Graph Automatic Semigroups 131

7.1 Zero Unions 132

7.2 Ordinal Sums 134

7.3 Semidirect and Direct Products 138

7.4 Free Products 145

7.5 Rees Matrix Semigroups 146
7.6 Semilattices of Semigroups 148
7.7 Bruck-Reilly Extensions 149

8 Conclusion and Overview of Open Questions 151

Chapter 1

Introduction and Background

Graph automatic semigroups are an extension of the concept of automaticity, as introduced for groups in [18] and for semigroups in [14]. They can also be seen as an extension of finitely generated FA-presentable semigroups, as considered in [9]. As a generalisation of both these concepts, it is interesting to examine which properties of FA-presentable semigroups and automatic semigroups extend to the graph automatic case.

Graph automatic groups and semigroups can also be considered as those whose Cayley graphs admit an automatic presentation, as in the sense of [26]. The concept of a Cayley FA-presentable group was considered in [7], where it is shown that there exist semigroups whose Cayley graphs are FA-presentable which are not automatic. The concept of the Cayley graph of a semigroup being FA-presentable was discussed in [9], where it is used to show that the classes of automatic semigroups and FA-presentable semigroups are incomparable.

The main study of graph automatic groups is in [25]. Here, the authors examine properties and constructions of graph automatic groups. As with automatic groups, the definition of graph automatic groups considers the group as a semigroup. Therefore it is natural to extend the concept of graph automaticity to semigroups, and examine whether the properties of graph automatic groups still hold in the semigroup case, and to examine further semigroup-theoretic constructions.

In this thesis we provide an introduction to the theory of graph automatic semigroups. We begin by providing some background definitions in this chapter.

In Chapter 2 we introduce the concept of graph automatic semigroups and give some initial examples. We compare the concept of graph automaticity with automaticity and FA-presentability, showing that any automatic semigroup is graph automatic, and that any finitely generated FA-presentable semigroup is graph automatic. We also show that graph automaticity of a semigroup is equivalent to FA-presentability of the Cayley graph of the semigroup, and so we may consider our graph automatic semigroups as FA-presentable structures. We then demonstrate some of the properties of graph automatic semigroups, namely that the existence of a graph automatic structure is independent of the choice of finite generating set, that every graph automatic semigroup has an injective graph automatic structure, and that graph automatic semigroups have decidable word problem.

In Chapter 3 we consider substructures of graph automatic semigroups. We examine the conditions under which subsemigroups and ideals of graph automatic semigroups can be shown to be graph automatic, in particular showing that left ideals and large subsemigroups preserve graph automaticity. We then go on to consider small extensions of graph automatic semigroups, and demonstrate some situations where these can be shown to preserve graph automaticity.

Chapters 4 and 5 consider semigroup constructions. We examine whether various semigroup constructions preserve graph automaticity, and whether graph automaticity of the semigroup construction implies graph automaticity of the original semigroups. In Chapter 4 we consider various products of semigroups, examining free products, semidirect products, direct products and Bruck-Reilly extensions. In Chapter 5 we examine unions of semigroups. We consider Rees matrix constructions, semilattices of semigroups, zero unions, and ordinal sums. Throughout both these chapters we also compare our results regarding constructions to the similar results for automatic and FA-presentable semigroups.

For the final two chapters we focus on graph automatic semigroups which have a graph automatic structure over a single letter alphabet. In Chapter 6 we introduce these unary graph automatic semigroups and consider some of their properties. We show that infinite unary graph automatic semigroups are not periodic and that we can find a uniform set of normal forms for their elements. We also show that there are restrictions on the forms of the automata which recognise multiplication in unary graph automatic structures, and provide examples of semigroups which are and which are not unary graph automatic. Chapter 7 then revisits the constructions from Chapters 4 and 5 in the context of unary graph automatic semigroups.

1.1 Languages and Automata

In this section we include some standard definitions and results from automata and language theory. More background to this topic can be found in any standard formal language theory book, such as [31], [23], [27] and the introductory chapter of [18].

An *alphabet* is a set of symbols, and a finite sequence of such symbols is called a *word*. The symbol ϵ represents the *empty word*, that is the word of length zero. We may *concatenate* two words $\alpha = a_1a_2 \dots a_m$ and $\beta = b_1b_2 \dots b_n$ by joining them together to form a new word $\alpha \cdot \beta = a_1a_2 \dots a_mb_1b_2 \dots b_n$, often denoted as just $\alpha\beta$. Note that $\epsilon \cdot \alpha = \alpha \cdot \epsilon = \alpha$ for any word α. The set of all possible words over an alphabet A is denoted A^*, and the set of all words with positive length (i.e all strings excluding the empty word) is denoted A^+.

A *language* is any subset of A^*. If L and K are languages over A then $L \cap K$, $L \cup K$ and $L \setminus K$ are also languages. We may also concatenate languages to get the *product* of two languages, written $K \cdot L$ or KL and defined to be

$$KL = \{\alpha\beta : \alpha \in K, \beta \in L\}.$$

We may take the *Kleene star* of a language, defined to be

$$\bigcup_{i \in \mathbb{N}_0} L^i$$

where $L^0 = \{\epsilon\}$ and $L^{i+1} = L^i \cdot L$. In a similar way, we have that

$$L^+ = \bigcup_{i \in \mathbb{N}} L^i.$$

A *many-variable language* is a language over an n-tuple of alphabets

$$(A_1, A_2, \ldots, A_n)$$

consisting of n-tuples of words $(\alpha_1, \alpha_2, \ldots, \alpha_n)$, with $\alpha_i \in A_i^*$. We may consider such a string as a string of n-tuples, in which case we must include a *padding symbol*, \$, not already in our alphabets, to account for the case where the words α_i have different lengths. Thus we consider words over the *padded alphabet*

$$B = (B_1 \times B_2 \times \cdots \times B_n) \setminus \{(\$, \$, \ldots, \$)\},$$

where $B_i = A_i \cup \{\$\}$. Then given a word $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ over B such that α_j is the longest word for $1 \leq j \leq n$, we pad each word α_i with \$ symbols at the end in order to make each word have the same length as α_j. In this way we may think of our words as the concatenation of letters which are n-tuples from B.

If we have languages K and L over the alphabets A_1 and A_2 respectively, then we may take their direct product to get a new language over the padded alphabet $((A_1 \cup \{\$\}) \times (A_2 \cup \{\$\})) \setminus (\$, \$)$.

1.1.1 Regular Languages

Regular expressions over a language A are defined as follows.

- Each of $\emptyset, \epsilon$ and a, for each $a \in A$ are regular expressions.
- If r and s are regular expressions, then so is $r \cup s$.
- If r and s are regular expressions, then so is rs.
- If r is a regular expression, then so is r^*.

Every regular expression can be derived by finitely many applications of the above steps, and any regular expression defines a language. A language is *regular* if it is the language defined by some regular expression.

If A, A_1 and A_2 are finite alphabets then regular languages have the following properties :

- $\emptyset, A^+$ and A^* are regular languages.
- any finite subset of A^* is a regular language.
- if $K \subseteq A^*$ and $L \subseteq A^*$ are regular, then $K \cup L, K \cap L, K \setminus L, KL$ and L^* are all regular.
- if $K \subseteq A_1^*$ is regular, and $\varphi : A_1^* \to A_2^*$ is a homomorphism (see Section 1.2), then φK is regular.
- if $K \subseteq A_2^*$ is regular, and $\varphi : A_1^* \to A_2^*$ is a homomorphism, then $\varphi^{-1}K$ is regular.
- if $K \subseteq A_1^*$ and $L \subseteq A_2^*$ are regular, then $K \times L$ is regular.
- if $U \subseteq A^* \times A^*$ is regular, then

  $$\{\alpha \in A^* : (\alpha, \beta) \in U \text{ for some } \beta \in A^*\}$$

  is regular.

See [22], for example, for proofs of the first five of these properties, and [5] for the proofs of the final two.

Note that when we refer to $K \times L$, we are implicitly taking the padded versions of these languages to allow for words of different lengths. If we have two regular languages which are padded products in this way, then the following result, which is Lemma 5.3 of [2], gives us a condition for regularity of their concatenation.

Proposition 1.1.1 (Theorem 3.3 of [17]). *Let A be an alphabet and let M and N be regular languages over $((A^* \cup \{\$\}) \times (A^* \cup \{\$\})) \setminus \{(\$, \$)\}$. If there exists a constant C such that for any two words $w_1, w_2 \in A^*$ we have that $(w_1, w_2) \in M$ implies that*

$$||w_1| - |w_2|| \leq C,$$

then the language MN is regular.

In particular, note that this condition will always be satisfied if the first of our languages is finite.

If we have a regular language R over the padded alphabet $A_1 \times A_1$ we denote the set of words which appear in the first component of R by

$$R^{(1)} = \{\alpha \in A_1^* : (\alpha, \beta) \in R \text{ for some } \beta \in A_2^*\}$$

and the set of words which appear in the second component of a word from R by

$$R^{(2)} = \{\beta \in A_2^* : (\alpha, \beta) \in R \text{ for some } \alpha \in A_1^*\}.$$

These are both regular.

The following result of [14], which follows from a result of [18], will also be useful.

Proposition 1.1.2 (Proposition 2.3 of [14]). *Let Σ be a finite alphabet. If*

$U, V \subseteq \Sigma^* \times \Sigma^*$ are regular, then

$$\{(\alpha, \gamma) \in \Sigma^* \times \Sigma^* : \text{ there exists } \beta \in \Sigma^* \text{ such that } (\alpha, \beta) \in U \text{ and } (\beta, \gamma) \in V\}$$

is regular.

Proof. This follows from Theorem 1.4.6 of [18]. □

1.1.2 Finite State Automata

Definition 1.1.3. A *(deterministic) finite state automaton* is a 5-tuple $\mathscr{A} = (\Sigma, S, q_0, F, \delta)$, where

- Σ is a finite alphabet,
- S is a finite set of states,
- $q_0 \in S$ is a distinguished *start state*,
- $F \subseteq S$ is a set of *accept states*, and
- $\delta : S \times \Sigma \to S$ is a *transition function* (which may be a partial function).

We will often refer just to an automaton when we mean a deterministic finite state automaton.

Let $\mathscr{A} = (\Sigma, S, q_0, F, \delta)$ be a finite state automaton. Then we say that $\mathscr{A}$ *accepts* a word $\alpha = a_1 a_2 \dots a_n \in \Sigma^*$ if there is a sequence of states $q_0, q_1, q_2, \dots, q_n$ such that $\delta(q_i, a_{i+1}) = q_{i+1}$ and $q_n \in F$.

If L is the set of words accepted by $\mathscr{A}$ then we say L is the language accepted or *recognised* by $\mathscr{A}$.

An automaton is *nondeterministic* if the transition function can also read the empty word, and may output multiple states, that is $\delta : S \times (\Sigma \cup \{\epsilon\}) \to \mathcal{P}(S)$. Every nondeterministic automaton is equivalent to a deterministic automaton, and so we may assume that our automata are deterministic.

Proposition 1.1.4 (Kleene's Theorem)**.** *Regular languages are precisely the languages recognised by finite state automata.*

1.2 Semigroups

In this section we introduce some basic definitions of semigroup theory. These definitions are standard and can be found in any introductory book on semigroup theory, such as [24], [15] and [19].

A *semigroup* is a set with an associative binary operation $\cdot$, that is

$$(r \cdot s) \cdot t = r \cdot (s \cdot t)$$

for all $r, s, t \in S$. We will frequently omit the operator and merely write st for $s \cdot t$. We often refer to our binary operation as multiplication. A semigroup is *commutative* if in addition we have that

$$st = ts$$

for all $s, t \in S$.

Note that throughout this thesis we shall assume that our semigroups are non-empty.

If S contains an element e such that $es = s$ for all $s \in S$ then e is a *left identity* of S. Similarly, if $se = s$ for all $s \in S$ then e is a *right identity* of S, and if e is both a left and a right identity then we call it the *identity* of S, often denoted 1. Note that a semigroup can have at most one identity element. A semigroup with an identity is called a *monoid*.

A *group* is a monoid with the additional condition that for each $s \in S$ there is a unique element s^{-1} such that $ss^{-1} = 1 = s^{-1}s$.

If S contains an element z such that $zs = z$ for all $s \in S$ then z is a *left zero*. Similarly, if $sz = z$ for all $s \in S$ then z is a *right zero*, and if z is both a left and a right zero for S then it is a *zero*.

An *idempotent* is an element $e \in S$ such that $e^2 = e$.

A non-empty subset T of S is a *subsemigroup* if it is closed under multiplication, that is $st \in T$ for all $s, t \in T$. If a subsemigroup of S is a group with respect to the operation of S then it is a *subgroup* of S. A non-empty subset I of S is a *left ideal* if $SI \subseteq I$, and a *right ideal* if $IS \subseteq I$. If I is both a left and a right ideal then S is a (two-sided) *ideal*. Every left, right or two-sided ideal is also a subsemigroup.

An element of a semigroup $c \in S$ is *left cancellative* if for all $s, t \in S$ we have that $cs = ct$ implies $s = t$. The element c is *right cancellative* if $sc = tc$ implies that $s = t$, and is *cancellative* if it is both left and right cancellative. A semigroup S is cancellative (or left cancellative or right cancellative) if all elements of S are cancellative (respectively left cancellative or right cancellative). Groups are always cancellative.

If S and T are semigroups then a *semigroup homomorphism* is a map $\varphi : S \to T$ such that $\varphi(s)\varphi(t) = \varphi(st)$ for all $s, t \in S$. If S and T are monoids then a *monoid homomorphism* has the additional property that $\varphi(1_S) = 1_T$, where 1_S and 1_T are the identity elements of S and T respectively. In either case, if φ is a bijection then we call φ an *isomorphism*, and if $S = T$ then φ is an *endomorphism*. If φ is an isomorphism and $S = T$ then it is an *automorphism*.

A *generating set* X for a semigroup S is a set such that any element of S is a finite product of elements from X. If X is a generating set of S then we write $S = \langle X \rangle$. If there is a finite set $X = \{x_1, x_2, \ldots, x_n\}$ such that X generates S then S is *finitely generated*, and we may write $S = \langle x_1, x_2, \ldots, x_n \rangle$. A semigroup is *monogenic* if it is generated by a single element.

Given a set X, the set X^+ under the operation of concatenation is the *free semigroup on* X, and the set X^* under concatenation is the *free monoid on* X.

A *relation* R on a set X is a subset of $X \times X$. Two elements x and y are *related* if $(x, y) \in R$. This is often written as xRy. A relation is *reflexive* if $(x, x) \in R$ for all $x \in X$, *symmetric* if $(x, y) \in R$ implies that $(y, x) \in R$, and *transitive* if $(x, y) \in R$ and $(y, z) \in R$ implies that $(x, z) \in R$, for all $x, y, z \in X$. A relation which is reflexive, symmetric and transitive is an *equivalence relation*. A *congruence* ρ on a semigroup S is an equivalence relation on S such that if $(x, y) \in \rho$ then $(zx, zy) \in \rho$ and $(xz, yz) \in \rho$ for all $z \in S$. The set of all equivalence classes of a congruence ρ forms a semigroup S/ρ, known as the *quotient* of S by ρ.

A *semigroup presentation* for a semigroup S is a pair $\langle X \mid R\rangle$, where X is a set and $R \subseteq X^+ \times X^+$ is a set of relations. This defines the semigroup isomorphic to the quotient X^+/ρ, where ρ is the least congruence containing R. A semigroup is *finitely presented* if both X and R are finite.

A *monoid presentation* for a monoid is $\langle X \mid R\rangle$, where X is a set and $R \subseteq X^* \times X^*$ is a set of relations. This defines the semigroup isomorphic to the quotient X^*/ρ, where ρ is the least congruence containing R.

The *semigroup free product* of semigroups S and T is the set of all finite strings $s_1s_2\ldots s_n$ for $n \geq 1$, where $s_i \in S \cup T$ for $1 \leq i \leq n$ and $s_i \in S$ if and only if $s_{i+1} \in T$. Then multiplication is defined as

$$st = \begin{cases} s_1s_2\ldots s_nt_1t_2\ldots t_m, & \text{if } s_n \in S \text{ and } t_1 \in T \text{ or } s_n \in T \text{ and } t_1 \in S \\ s_1s_2\ldots s_{n-1}ut_2\ldots t_m, & \text{if } s_n, t_1 \in S \text{ or } s_n, t_1 \in T \end{cases}$$

for $s = s_1s_2\ldots s_n$ and $t = t_1t_2\ldots t_m$, and $u = s_nt_1$ when both s_n and t_1 are in S or both are in T.

The *direct product* of semigroups S and T is the set $S \times T$ with multiplication

$$(s,t)(s',t') = (ss',tt')$$

for $s, s' \in S$ and $t, t' \in T$.

The *Cayley graph* of a semigroup S with respect to a finite generating set X is the directed graph whose vertices are the elements of the semigroup S, with edges labelled by the elements of X, where there is a directed edge between two vertices s and t labelled by $x \in X$ if and only if $sx = t$.

Chapter 2

Graph Automatic Semigroups: An Introduction

In this chapter we will introduce the concept of graph automatic semigroups, provide some examples of such semigroups, and examine some of their properties. We will also consider some concepts which are related to graph automaticity, namely automaticity and FA-presentability, and examine the relationships between these concepts and graph automaticity.

2.1 Definition and Examples

In [25], the authors define a graph automatic group and examine various properties and constructions relating to these groups. However, the definition of a graph automatic group does not rely on the existence of an identity or inverses, it merely requires that the multiplication in the group is recognisable. Thus the definition given in [25] treats the group as a semigroup, and so would seem more natural as a semigroup definition. Hence we may use the same definition for a graph automatic semigroup.

Definition 2.1.1. Let S be a semigroup generated by a finite set X. We call S *Cayley graph automatic* (or merely *graph automatic*) if and only if there exists a finite alphabet Σ, a regular language $R \subseteq \Sigma^*$, and a surjective mapping $\nu : R \to S$, for which the sets

$$R_= = \{(\alpha, \beta) \in R \times R : \nu(\alpha) = \nu(\beta)\}$$

and

$$R_x = \{(\alpha, \beta) \in R \times R : \nu(\alpha)x = \nu(\beta)\}$$

are regular, for each $x \in X$. In this case we say that (X, Σ, R, ν) is a *graph automatic structure* for S.

For brevity, we will usually omit the word Cayley, referring to our semigroups as graph automatic.

Note that this definition seemingly relies on our choice of generating set, X. For the moment, we say that a semigroup is graph automatic as long as it has a graph automatic structure for some finite generating set. We will examine the effect that the choice of generating set has on graph automaticity in Section 2.4.

We now give some examples of graph automatic semigroups. We begin by showing that any finite semigroup is graph automatic.

Example 2.1.2. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a finite semigroup. Then S is finitely generated by the set S itself. Let $\Sigma = \{a_1, a_2, \ldots, a_n\}$ and $R = \Sigma$. Define $\nu : R \to S$ by

$$\nu(a_i) = s_i.$$

This is clearly surjective. Then the set

$$\begin{aligned} R_= &= \{(\alpha, \beta) \in R \times R : \nu(\alpha) = \nu(\beta)\} \\ &= \{(a_i, a_i) : 1 \leq i \leq n\} \end{aligned}$$

is finite, and so is regular. For each $s \in S$, we have that

$$\begin{aligned} R_s =& \{(\alpha, \beta) \in R \times R : \nu(\alpha)s = \nu(\beta)\} \\ =& \{(a_i, a_j) \in R \times R : s_i s = s_j\} \end{aligned}$$

is also finite, and so is regular. Hence S is graph automatic, and

$$(S, \Sigma = \{a_1, a_2, \ldots, a_n\}, R = \{a_1, a_2, \ldots, a_n\}, \nu)$$

is a graph automatic structure for S.

Next we consider the *bicyclic monoid*. This is the semigroup defined by the monoid presentation

$$B = \langle b, c \mid bc = 1 \rangle.$$

Note that every element of B can be expressed as $c^i b^j$ for some $i, j \in \mathbb{N}_0$. More information regarding the bicyclic monoid can be found in [15] or [24]. In the following example we show that the bicyclic monoid is graph automatic.

Example 2.1.3. Consider the bicyclic monoid, $B = \{c^i b^j : i, j \in \mathbb{N}_0\}$. Then B is generated as a semigroup by the set $X = \{b, c\}$. Let $\Sigma = \{\beta, \gamma\}$ and let $R = \gamma^* \beta^*$. We define $\nu : R \to B$ by

$$\nu(\gamma^i \beta^j) = c^i b^j$$

which is surjective, and we have that

$$\begin{aligned} R_= =& \{(\gamma^i \beta^j, \gamma^k \beta^l) \in R \times R : \nu(\gamma^i \beta^j) = \nu(\gamma^k \beta^l)\} \\ =& \{(\gamma^i \beta^j, \gamma^i \beta^j) : i, j \in \mathbb{N}_0\} \\ =& \{(\alpha, \alpha) : \alpha \in R\} \end{aligned}$$

is regular. Then

$$\begin{aligned}R_b =&\{(\gamma^i\beta^j,\gamma^k\beta^l)\in R\times R : \nu(\gamma^i\beta^j)b=\nu(\gamma^k\beta^l)\}\\ =&\{(\gamma^i\beta^j,\gamma^i\beta^{j+1}) : i,j\in\mathbb{N}_0\}\\ =&\{(\alpha,\alpha):\alpha\in R\}(\$,\beta)\end{aligned}$$

is regular, and

$$\begin{aligned}R_c =&\{(\gamma^i\beta^j,\gamma^k\beta^l)\in R\times R : \nu(\gamma^i\beta^j)c=\nu(\gamma^k\beta^l)\}\\ =&\{(\gamma^i\beta^j,\gamma^i\beta^{j-1}) : i\in\mathbb{N}_0, j\in\mathbb{N}\}\cup\{(\gamma^i,\gamma^{i+1}):i\in\mathbb{N}_0\}\\ =&\{(\alpha,\alpha):\alpha\in R\}(\beta,\$)\cup(\gamma,\gamma)^*(\$,\gamma)\end{aligned}$$

is regular. Hence B is graph automatic, with graph automatic structure

$$(X=\{b,c\},\Sigma=\{\beta,\gamma\},R=\gamma^*\beta^*,\nu).$$

In our next example we consider finitely generated free semigroups. Recall that a semigroup F is *free* if it is given by the semigroup presentation $\langle X \mid \quad \rangle$. We now show that any finitely generated free semigroup is graph automatic.

Example 2.1.4. Let F_X be the free semigroup generated by the set $X = \{x_1, x_2, \ldots, x_n\}$. Let $\Sigma = \{a_1, a_2, \ldots, a_n\}$ and $R = \Sigma^+$. We define $\nu : R \to F_X$ by

$$\nu(a_{k_1}a_{k_2}\ldots a_{k_m}) = x_{k_1}x_{k_2}\ldots x_{k_m}$$

for $k_i \in \{1, 2, \ldots, n\}$ and $m \in \mathbb{N}$. Note that this is surjective. Then

$$\begin{aligned}R_= &= \{(a_{k_1}\ldots a_{k_m}, a_{j_1}\ldots a_{j_l})\in R\times R : \nu(a_{k_1}\ldots a_{k_m}) = \nu(a_{j_1}\ldots a_{j_l})\}\\ &= \{(a_{k_1}\ldots a_{k_m}, a_{k_1}\ldots a_{k_m}) : 1\le k_i\le n\}\\ &= \{(\alpha,\alpha):\alpha\in R\}\end{aligned}$$

is regular, and for any $x_i \in X$ we have that

$$
\begin{aligned}
R_{x_i} &= \{(a_{k_1} \dots a_{k_m}, a_{j_1} \dots a_{j_l}) \in R \times R : \nu(a_{k_1} \dots a_{k_m})x_i = \nu(a_{j_1} \dots a_{j_l})\} \\
&= \{(a_{k_1} \dots a_{k_m}, a_{k_1} \dots a_{k_m} a_i) : 1 \leq k_i \leq n\} \\
&= \{(\alpha, \alpha) : \alpha \in R\}(\$, a_i)
\end{aligned}
$$

is regular. Thus F_X is graph automatic, with graph automatic structure

$$(X = \{x_1, x_2, \dots, x_n\}, \Sigma = \{a_1, a_2, \dots, a_n\}, R = \Sigma^+, \nu).$$

In a similar way, we may also show that any finitely generated free monoid is also graph automatic, by replacing Σ^+ in the above example by Σ^* and defining $\nu(\epsilon) = 1$, then proceeding in a similar way. We will see an alternative way of demonstrating this fact in Section 3.2.

Note that in Section 2.2 we will see an alternative way of showing that the semigroups in Examples 2.1.3 and 2.1.4 are graph automatic, as a consequence of Proposition 2.2.2.

2.2 Concepts Related to Graph Automaticity

We now examine some other ways of representing semigroups with regular languages and see how they relate to graph automaticity. In particular we look at automaticity and FA-presentability.

2.2.1 Automaticity

The concept of automatic groups was first introduced in [18], and this idea was extended to semigroups in [14]. We consider the definition of an automatic semigroup, as given in [14].

Definition 2.2.1. Let S be a semigroup, A be a finite set, L be a regular subset of A^+, and $\varphi : A^+ \to S$ be a homomorphism with $\varphi(L) = S$. Then (A, L, φ) is an *automatic structure* for S if

$$L_{=} = \{(\alpha, \beta) \in L \times L : \varphi(\alpha) = \varphi(\beta)\}$$

and

$$L_a = \{(\alpha, \beta) \in L \times L : \varphi(\alpha)a = \varphi(\beta)\}$$

for each $a \in A$ are regular. A semigroup is *automatic* if it has an automatic structure for some A and L.

Note that this definition is quite similar to our definition of a graph automatic semigroup. However, for an automatic semigroup the alphabet and the generating set are conflated, and the surjective map becomes a homomorphism.

In [14], the authors examine numerous properties of automatic semigroups, in some case showing that properties can be generalised from automatic groups but also highlighting properties which do not generalise. These results include the fact that that the word problem for automatic semigroups is decidable in quadratic time; that automatic semigroups have a structure with uniqueness (that is a structure where the homomorphism is also a bijection); that the existence of an automatic structure does depend on the choice of generating set; and that, unlike automatic groups, automatic semigroups are not necessarily finitely presented. The authors also consider semigroup constructions which preserve automaticity. For example, it is shown that adding and removing identity and zero elements preserves automaticity, and that the free product of two semigroups is automatic if and only if each semigroup is automatic. Further constructions have been considered elsewhere, for example in [12] it is shown that the direct product of two automatic semigroups is automatic if and only if it is finitely generated; in [13] it is shown that completely-simple semigroups are automatic if and only if their base group is automatic; and in [21] it is shown that automaticity is preserved by small extensions and large subsemigroups. A survey of results regarding constructions of automatic semigroups is given in [2].

In [25] it is stated that any automatic group is also a graph automatic group.

This is also true for semigroups.

Proposition 2.2.2. *Every automatic semigroup is graph automatic.*

Proof. Let S be an automatic semigroup. Then S has an automatic structure (A, L, φ) and comparing the two definitions we see that A can be identified with X, then $\Sigma = A$ and $R = L$, and $\nu = \varphi$, so S has a graph automatic structure (A, A, L, φ). □

However, the converse does not hold; there are semigroups which are graph automatic but not graph automatic. We shall see examples of such semigroups in Section 2.3.

2.2.2 FA-Presentability

We now consider FA-presentability, a concept which is defined for a general relational structure. We begin by considering FA-presentable structures in general, after which we will examine particular types of FA-presentable structures, namely semigroups and Cayley graphs.

FA-presentable structures were first introduced in [26]. They are defined as follows.

Definition 2.2.3. Let $\mathcal{S} = (S, R_1, R_2, \ldots, R_n)$ be a relational structure, where each relation R_i is of arity r_i. Let A be a finite alphabet and $L \subseteq A^*$ be a regular language over A. Let $\psi : L \to S$ be a surjective mapping. Then (L, ψ) is an *automatic presentation* for $\mathcal{S}$ if the relations

$$L_= = \{(\alpha, \beta) \in L \times L : \psi(\alpha) = \psi(\beta)\}$$

and

$$L_{R_i} = \{(\alpha_1, \alpha_2, \ldots, \alpha_{r_i}) \in L^{r_i} : (\psi(\alpha_1), \psi(\alpha_2), \ldots, \psi(\alpha_{r_i})) \in R_i\}$$

are regular for each R_i. A structure with an automatic presentation is *FA-*

presentable.

It is shown in [26] that any FA-presentable structure has an injective automatic presentation, and in [6] it is shown that any FA-presentable structure has an automatic presentation over a two letter alphabet. These may be obtained simultaneously, by first finding an FA-presentable structure over a binary alphabet and then restricting this to an injective automatic presentation.

We now consider FA-presentable semigroups, which were studied in [9]. In this setting, we view our semigroup as a relational structure, where the binary operation $\circ$ is viewed as a ternary relation.

Definition 2.2.4. Let S be a semigroup, let L be a regular language over a finite alphabet A and let $\varphi : L \to S$ be a surjective mapping. Then (L, ψ) is an automatic presentation for S if

$$L_= = \{(\alpha, \beta) \in L \times L : \psi(\alpha) = \psi(\beta)\}$$

and

$$L_\circ = \{(\alpha, \beta, \gamma) \in L^3 : \psi(\alpha)\psi(\beta) = \psi(\gamma)\}$$

are both regular. A semigroup is FA-presentable if it has an automatic presentation for some L and ψ.

Comparing this to our definition of a graph automatic semigroup, we see that while graph automaticity requires us to be able to recognise only multiplication by generators, an FA-presentable structure recognises all multiplication in the semigroup.

Constructions of FA-presentable semigroups are examined in [10]. Among the results it is shown that the direct product of two FA-presentable semigroups is FA-presentable, but the converse does not hold; that the free product of two semigroups is FA-presentable if and only if the semigroups are both trivial; that adding and removing zero and identity elements preserves FA-presentability; and that FA-presentable semigroups are not closed under taking small extensions.

Note that an FA-presentable semigroup need not be finitely generated. In the case where an FA-presentable semigroup is finitely generated, however, we have that it is also graph automatic.

Proposition 2.2.5. *Finitely generated FA-presentable semigroups are graph automatic.*

Proof. Let S be a finitely generated FA-presentable semigroup with automatic presentation (L, ψ). Then $L_=$ is regular, so we merely need to show that L_x is regular for each x in our finite generating set X. Fix β such that $\psi(\beta) = x$ for some $x \in X$. Then

$$(L \times \{\beta\} \times L) \cap L_\circ$$

is regular and we may obtain L_x by reading only the first and third components. Hence L_x is regular and (X, A, L, ψ) is a graph automatic structure for S. $\square$

Note that the classes of automatic semigroups and FA-presentable semigroups are incomparable. This is demonstrated in [9], by considering the Cayley graphs of semigroups as FA-presentable structures. We will see that the Cayley graph of a semigroup being FA-presentable is equivalent to the semigroup being graph automatic.

We may view a directed, labelled graph as a relational structure. If $\Gamma = (V, E)$ and the edges of Γ are labelled by some finite set L, then the underlying set is the set of vertices, and our relations are

$$E_l = \{(u, v) \in V \times V : (u, v) \in E \text{ and is labelled } l\}$$

for each $l \in L$.

In particular, we consider the case where Γ is the Cayley graph of some finitely generated semigroup S. In this case we see that S is graph automatic precisely if Γ is FA-presentable

Proposition 2.2.6. *A finitely generated semigroup is graph automatic if and only if the Cayley graph of the semigroup is FA-presentable.*

Proof. Let S be a graph automatic semigroup with structure (X, Σ, R, ν). Then $R_=$ and R_x are regular, for each $x \in X$. The Cayley graph of S is given by $\Gamma = (S, E)$, with $(s, t) \in E$ if and only if $sx = t$ for some $x \in X$. So $(\alpha, \beta) \in R_x$ if and only if $(\nu(\alpha), \nu(\beta)) \in E_x$. So we have that $R_x = R_{E_x}$. Thus we have that (R, ν) is an automatic presentation for Γ.

Conversely, if L is a regular language over a finite alphabet A and (L, ψ) is an automatic presentation for the Cayley graph $\Gamma = (S, E)$ labelled by X, then each of the relations $L_=$ and L_{E_x} are regular, where

$$\begin{aligned} L_{E_x} &= \{(\alpha, \beta) \in L \times L : (\psi(\alpha), \psi(\beta)) \in E \text{ is labelled by } x\} \\ &= \{(\alpha, \beta) \in L \times L : \psi(\alpha)x = \psi(\beta)\} \\ &= L_x \end{aligned}$$

and so (X, A, L, ψ) is a graph automatic structure for S. □

Thus we may consider our graph automatic semigroups as semigroups whose Cayley graphs are FA-presentable. This means that our graph automatic semigroups immediately inherit the properties of FA-presentable structures. For example, we can immediately say that every graph automatic structure has an injective binary graph automatic structure.

2.3 Further Examples

In this section we consider further examples of graph automatic semigroups. Propositions 2.2.2 and 2.2.5 immediately allow us to find multiple examples. For example, finite semigroups, the bicyclic monoid, and finitely generated free semigroups are all examples of automatic semigroups given in [14]. Thus we could have used Proposition 2.2.2 to immediately determine that they were graph automatic, rather than proving it directly as we did in Examples 2.1.2, 2.1.3, and 2.1.4.

Using results from Section 2.2, we may show that any finitely generated commutative semigroup is graph automatic, with the help of the following proposition from [9].

Proposition 2.3.1 (Theorem 6.1 of [9])**.** *Every finitely generated commutative semigroup admits an automatic presentation.*

Now as every finitely generated FA-presentable semigroup is graph automatic (Proposition 2.2.5), we have that:

Corollary 2.3.2. *Every finitely generated commutative semigroup is graph automatic.*

Proof. Every finitely generated FA-presentable semigroup is graph automatic, by Proposition 2.2.5. Then as every finitely generated commutative semigroup is FA-presentable, it is also graph automatic. □

Note that such semigroups are not necessarily automatic. In [20] the authors show that there exist finitely generated commutative semigroups which are not automatic by finding an example of such a semigroup, which is given below.

Example 2.3.3 (Example 4.1 of [20])**.** The semigroup defined by the presentation

$$\begin{aligned}\langle a, b, x, y \mid & aax = bx, bby = ay, ab = ba, ax = xa, \\ & ay = ya, bx = xb, by = yb, xy = yx\rangle\end{aligned}$$

is commutative but not automatic.

This provides us with an example of a semigroup which is graph automatic but not automatic.

Proposition 2.3.4. *There exist graph automatic semigroups which are not automatic.*

Proof. The semigroup in Example 2.3.3 is a finitely generated commutative semigroup, thus is graph automatic by Corollary 2.3.2. However, it is not automatic, as shown in [20]. □

We also have examples of semigroups which are graph automatic but not FA-presentable. To find an example of a graph automatic semigroup which is not FA-presentable, we use the following proposition.

Proposition 2.3.5 (Proposition 4.1 of [10])**.** *The semigroup free product of two semigroups S and T is FA-presentable if and only if S and T are trivial.*

This allows us to show that there are semigroups which are graph automatic but not FA-presentable.

Proposition 2.3.6. *There exist graph automatic semigroups which are not FA-presentable.*

Proof. The free group on two generators, F_2, is an example of a semigroup which is graph automatic but not FA-presentable. In Example 2.1.4 we saw that any finitely generated free semigroup is graph automatic, so in particular F_2 is graph automatic. However, as F_2 is the free product of two monogenic free semigroups, it is not FA-presentable by Proposition 2.3.5. In fact, this can be extended to say that any F_X with $|X| > 1$ is graph automatic but not FA-presentable. □

Thus we see that the class of graph automatic semigroups is distinct from both automatic and FA-presentable semigroups.

Another, non-commutative, example of a semigroup which is graph automatic but not automatic is the Heisenberg group $\mathcal{H}_3(\mathbb{Z})$, that is the subgroup of $\mathrm{SL}(3, \mathbb{Z})$ consisting of upper triangular matrices of the form

$$\begin{pmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{pmatrix}.$$

In [18] the authors show that the Heisenberg group is not automatic. However, it is shown to be a graph automatic group in [7]. Thus it is also a graph automatic semigroup, which we demonstrate directly below.

Example 2.3.7. We show that $\mathcal{H}_3(\mathbb{Z})$ is a graph automatic semigroup. Let $\mathcal{H}_3(\mathbb{Z})$ be generated by $X = \{A, B, C\}$, where

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, B = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \text{ and } C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Let $\Sigma = \{0, 1\}$ and $L = \Sigma^*$. Then we may take $R = L^3$ over a padded alphabet Σ^3 and define $\nu : R \to \mathcal{H}_3(\mathbb{Z})$ by

$$\nu(\alpha, \beta, \gamma) = \begin{pmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{pmatrix}$$

where α, β and γ are the binary representations of a, b and c respectively. Then

$$\begin{aligned} R_= &= \{((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2)) \in R \times R : \nu(\alpha_1, \beta_1, \gamma_1) = \nu(\alpha_2, \beta_2, \gamma_2)\} \\ &= \{((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2)) \in R \times R : \alpha_1 = \alpha_2, \beta_1 = \beta_2, \text{ and } \gamma_1 = \gamma_2)\} \\ &= \{(\alpha, \alpha) : \alpha \in R\} \end{aligned}$$

is regular as we can easily check equality of binary words using automata. Then for our generators, we have that

$$\begin{aligned} R_A &= \{((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2)) \in R \times R : \nu(\alpha_1, \beta_1, \gamma_1)A = \nu(\alpha_2, \beta_2, \gamma_2)\} \\ &= \{((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2)) \in R \times R : \alpha_1 + 1 = \alpha_2, \beta_1 = \beta_2, \text{ and } \gamma_1 = \gamma_2\} \\ &= \{((\alpha_1, \beta_1, \gamma_1), (\alpha_1 + 1, \beta_1, \gamma_1)) : \alpha_1, \beta_1, \gamma_1 \in L\}, \end{aligned}$$

and

$$\begin{aligned}R_B &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_2,\beta_2,\gamma_2)) \in R\times R : \nu(\alpha_1,\beta_1,\gamma_1)B = \nu(\alpha_2,\beta_2,\gamma_2)\}\\ &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_2,\beta_2,\gamma_2)) \in R\times R : \alpha_1=\alpha_2, \beta_1+1=\beta_2, \text{ and } \gamma_1=\gamma_2\}\\ &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_1,\beta_1+1,\gamma_1)) : \alpha_1,\beta_1,\gamma_1 \in L\}\end{aligned}$$

and

$$\begin{aligned}R_C &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_2,\beta_2,\gamma_2)) \in R\times R : \nu(\alpha_1,\beta_1,\gamma_1)C = \nu(\alpha_2,\beta_2,\gamma_2)\}\\ &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_2,\beta_2,\gamma_2)) \in R\times R : \alpha_1=\alpha_2, \beta_1=\beta_2, \text{ and } \gamma_1+1=\gamma_2\}\\ &= \{((\alpha_1,\beta_1,\gamma_1),(\alpha_1,\beta_1,\gamma_1+1)) : \alpha_1,\beta_1,\gamma_1 \in L\}\end{aligned}$$

are regular, as we can check equality of binary words, and use automata to add one to a binary word. Thus

$$(\{A,B,C\},\Sigma^3,R,\nu)$$

is a graph automatic structure for $\mathcal{H}_3(\mathbb{Z})$.

In fact, we may easily extend this to any Heisenberg group $\mathcal{H}_n(\mathbb{Z})$, consisting of $n\times n$ matrices with entries 1 on the diagonal and 0 everywhere else other than the first row and last column, as noted in [25]. In this case, the generating set consists of all the $n\times n$ matrices that differ from the identity matrix only by containing a single additional entry of 1, either in the first row or last column.

As well as not being automatic, we also have that the Heisenberg is not FA-presentable. This follows from the classification of finitely generated FA-presentable groups, given in [28]. A group is *virtually abelian* if it has an abelian subgroup of finite index. We then have the following result:

Proposition 2.3.8 (Theorem 8 of [28]). *Let G be a finitely generated group; then G has an automatic presentation if and only if G is virtually abelian.*

As the Heisenberg group is not virtually abelian, it is therefore not FA-presentable. Thus the Heisenberg group provides us with an example of a group, and hence a semigroup, which is neither graph automatic nor FA-presentable.

We will see an example of a non-group semigroup which is graph automatic but neither automatic nor FA-presentable in Section 4.1.

2.4 Properties of Graph Automatic Semigroups

In this section we will examine some of the properties of graph automatic semigroups. We will show that graph automaticity does not depend on our choice of generating set; that each element of a graph automatic semigroup is represented by a regular language; that we can always find a graph automatic structure where our map is a bijection; and that graph automatic monoids have decidable word problem.

2.4.1 Generating Sets

We begin by looking at the impact of the choice of generating set on graph automaticity. Note that Definition 2.1.1 requires that we merely have a graph automatic structure for some choice of generating set, and so we ask whether this choice of generating set impacts our ability to find a graph automatic structure. Certainly for automatic semigroups the choice of generating set does have an effect on our ability to find an automatic structure. In [14], the authors provide an example of a semigroup where the existence of an automatic structure depends on the choice of generating set. This example is given below.

Example 2.4.1 (Example 4.5 of [14])**.** Let F be the free semigroup generated by $\{a, b, c\}$, and consider the subsemigroup S of F generated by $X = \{c, ac, ca, ab, baba\}$. Then S has an automatic structure with respect to this generating set, thus is an automatic semigroup. However, S does not have an automatic structure with respect to the generating set $X \cup \{abab\}$.

Thus the choice of generating set does affect our ability to find an automatic structure, and we say that a semigroup is automatic as long as it has an automatic structure for some finite generating set.

In [9] it is shown that if the Cayley graph of a semigroup is FA-presentable then this does not depend on the choice of generating set. Thus, as this is equivalent to a semigroup being graph automatic, we may immediately say that the existence of a graph automatic structure is independent of the choice of generating set. However, we will show this directly, illustrating how a graph automatic structure for a new generating set may be found.

We begin by noting that if we have a graph automatic structure then the language recognising multiplication by any element is regular, not merely the languages recognising multiplication by a generator.

Proposition 2.4.2. *If S is graph automatic with structure (X, Σ, R, ν) and $y \in S$ then $R_y = \{(\alpha, \beta) \in R \times R : \nu(\alpha)y = \nu(\beta)\}$ is regular.*

Proof. We follow the proof of Proposition 3.2 of [14]. Let $y = x_1 x_2 \cdots x_n$ for some $x_i \in X$. As S is graph automatic we have that

$$\begin{aligned} R_{x_1} =& \{(\alpha, \alpha_1) \in R \times R : \nu(\alpha)x_1 = \nu(\alpha_1)\} \\ R_{x_2} =& \{(\alpha_1, \alpha_2) \in R \times R : \nu(\alpha_1)x_2 = \nu(\alpha_2)\} \\ &\vdots \\ R_{x_n} =& \{(\alpha_{n-1}, \beta) \in R \times R : \nu(\alpha_{n-1})x_n = \nu(\beta)\} \end{aligned}$$

are regular. Then by Proposition 1.1.2 we have that the sets

$$\begin{aligned} R_{x_1x_2} = \{(\alpha, \alpha_2) \in R \times R :& \text{ there is a } \alpha_1 \in R \text{ such that } (\alpha, \alpha_1) \in R_{x_1} \\ & \text{and } (\alpha_1, \alpha_2) \in R_{x_2}\} \\ R_{x_1x_2x_3} = \{(\alpha, \alpha_3) \in R \times R :& \text{ there is a } \alpha_2 \in R \text{ such that } (\alpha, \alpha_2) \in R_{x_1x_2} \\ & \text{and } (\alpha_2, \alpha_3) \in R_{x_3}\} \\ &\vdots \end{aligned}$$

$$R_{x_1x_2\cdots x_n} = \{(\alpha, \beta) \in R \times R : \text{ there is a } \alpha_{n-1} \in R \text{ such that } (\alpha, \alpha_{n-1}) \in R_{x_1\ldots x_{n-1}} \text{ and } (\alpha_{n-1}, \beta) \in R_{x_n}\}$$

are regular. Hence $R_y = R_{x_1x_2\cdots x_n}$ is regular. □

We now use this to show that graph automaticity is independent of our choice of generating set.

Theorem 2.4.3. *Let S be a graph automatic semigroup with respect to some finite generating set X. Then S is graph automatic with respect to any finite generating set.*

Proof. Let (X, Σ, R, ν) be a graph automatic structure for S. Let Y be a different finite generating set for S and let $y \in Y$. Then y can be written as $y = x_1x_2 \cdots x_n$ for some $x \in X$, and by Proposition 2.4.2 we have that R_y is regular. Thus (Y, Σ, R, ν) is also a graph automatic structure for S. □

Note that Proposition 2.4.2 holds for automatic semigroups. However, upon change of generating set, while we still have the map ν that is required for a graph automatic semigroup, we may not be able to find a homomorphism.

2.4.2 Preimages

We now consider the set of words representing a particular element in a graph automatic semigroup. We shall see that the preimage of each element is a regular language, in the same way as was shown for automatic semigroups in [14].

Proposition 2.4.4. *Let S be a graph automatic semigroup with graph automatic structure (X, Σ, R, ν). Then for any $s \in S$ the set $\{\alpha \in R : \nu(\alpha) = s\}$ is regular.*

Proof. We follow the proof of Lemma 3.1 in [14]. Let $\beta \in R$ such that $\nu(\beta) = s$. Then for any $\alpha \in R$ we have that $(\alpha, \beta) \in R_=$ if and only if $\nu(\alpha) = s$. Let

$$L = \{(\alpha, \beta) : \alpha \in R, \nu(\alpha) = s\}$$

$$= R_= \cap (R \times \{\beta\}).$$

So L is regular. Thus

$$L^{(1)} = \{\alpha \in \Sigma^* : (\alpha, \delta) \in L \text{ for some } \delta \in \Sigma^*\},$$

which is obtained by reading the first tape of L, is regular. But $(\alpha, \delta) \in L$ means we must have $\delta = \beta$ and so

$$L^{(1)} = \{\alpha \in \Sigma^* : (\alpha, \beta) \in L\}$$
$$= \{\alpha \in R : \nu(\alpha) = s\}.$$

Thus $\{\alpha \in R : \nu(\alpha) = s\}$ is regular. □

This means that each element is represented by a regular language, a fact which will become useful when we later consider various semigroup constructions.

2.4.3 Structures with Uniqueness

A semigroup has a graph automatic structure with uniqueness if there exists a graph automatic structure where each element of the semigroup is represented by precisely one word in our language. Alternatively, we can think of this as a structure where the map between the language and the semigroup is also injective (and hence is a bijection). Automatic semigroups have structures with uniqueness [14], as do FA-presentable structures [6]. As graph automatic semigroups are those whose Cayley graphs have an FA-presentable structure by Proposition 2.2.6, we must also have a structure with uniqueness for any graph automatic semigroup. We illustrate this below, demonstrating how we may turn any graph automatic structure into one with uniqueness.

We will use the *shortlex* ordering on elements of our language. Take a language R over an alphabet Σ. We introduce an ordering $<_\Sigma$ on the elements of Σ. We say that α precedes β lexicographically if $\alpha = \alpha_1\alpha_2\ldots\alpha_n$ and $\beta = \beta_1\beta_2\ldots\beta_n$, $\alpha_i = \beta_i$ for $i = 1, 2, \ldots, k$ for some $k < n$, and in the first place where they differ $\alpha_{k+1} <_\Sigma \beta_{k+1}$ with respect to the ordering on Σ. Then for words $\alpha, \beta \in R$ we have that $\alpha <_s \beta$ if and only if $|\alpha| < |\beta|$ or $|\alpha| = |\beta|$ and α precedes β lexicographically, where $<_s$ denotes the shortlex order.

We impose this ordering on the language for our graph automatic structure, and use it to find a structure with uniqueness.

Proposition 2.4.5. *Let S be a graph automatic semigroup. Then S has a graph automatic structure with uniqueness.*

Proof. Let (X, Σ, R, ν) be a graph automatic structure for S. Consider

$$L = \{\alpha \in R : \text{ if } (\alpha, \beta) \in R_= \text{ for any } \beta \in R \text{ then } \alpha \leq_s \beta\},$$

where $\alpha \leq_s \beta$ indicates that α is less than or equal to β in the shortlex order. We will show that L is regular. Consider

$$\begin{aligned} K &= R_= \cap \{(\alpha, \beta) : \alpha >_s \beta\} \\ &= \{(\alpha, \beta) : (\alpha, \beta) \in R_= \text{ and } \alpha >_s \beta\} \end{aligned}$$

(where again $>_s$ indicates the shortlex order). As S is graph automatic $R_=$ is regular, and as the shortlex order is recognisable by a finite state automaton $\{(\alpha, \beta) : \alpha >_s \beta\}$ is regular. Thus K is regular. Now let

$$\begin{aligned} J &= \{\alpha \in \Sigma^* : (\alpha, \beta) \in K \text{ for some } \beta \in \Sigma^*\} \\ &= \{\alpha \in R : (\alpha, \beta) \in R_= \text{ and } \alpha >_s \beta \text{ for some } \beta \in R\}. \end{aligned}$$

Note that $J = K^{(1)}$, thus is regular. Then

$$\begin{aligned} L &= R \setminus J \\ &= \{\alpha \in R : \text{ if } (\alpha, \beta) \in R_= \text{ for any } \beta \in R \text{ then } \alpha \leq_s \beta\} \end{aligned}$$

is regular.

Now consider the restriction map $\nu|_L : L \to S$. This is a surjective map where each element of S is represented by precisely one element of L. Then

$$L_= = R_= \cap (L \times L)$$

and

$$L_x = R_x \cap (L \times L)$$

are regular for $x \in X$. So $(X, \Sigma, L, \nu|_L)$ is a graph automatic structure for S with uniqueness. □

So we may take any graph automatic structure for a semigroup and turn it into a structure with uniqueness by taking only the shortest word (with respect to the shortlex ordering) representing each element of our semigroup. Hence from this point onwards we may assume, without loss of generality, that all our graph automatic structures are structures with uniqueness.

2.4.4 The Word Problem for Monoids

The *word problem* for semigroups asks whether, given two strings of generators for a semigroup, we can determine whether these represent the same element of our semigroup.

In [25] it is shown that graph automatic groups have soluble word problem. In this proof, the generating set can be considered as a monoid generating set, and so the same proof can be used to show that graph automatic monoids have decidable word problem. We first require the following lemma.

Lemma 2.4.6 (Lemma 8.1 of [25]). *Let Σ be a finite alphabet, $n \in \mathbb{N}$, $D \subseteq (\Sigma^*)^n$, and $f : D \to \Sigma^*$ be a function whose graph is FA-recognisable. Then there exists a linear time algorithm that given $d \in D$ computes the value $f(d)$. Furthermore, there is a constant K such that $|f(d)| \leq |d| + K$ for any $d \in D$, where $|d| = \max\{|d_i| : i = 1 \dots n\}$ for $d = (d_1, \dots, d_n) \in (\Sigma^*)^n$.*

We now reproduce the proof of Theorem 8.2 from [25], to show that graph automatic monoids have decidable word problem.

Proposition 2.4.7. *The word problem for graph automatic monoids is decidable in quadratic time.*

Proof. Let S be a graph automatic semigroup with graph automatic structure with uniqueness (X, Σ, R, ν). By Proposition 2.4.5 we may assume that ν is a bijection. Let $w = x_1 x_2 \dots x_n$ be a word made up of generators $x_i \in X$, and let $\bar{w}$ be the element of S defined by w. We want to find the unique word $\nu^{-1}(\bar{w}) \in R$ representing $\bar{w}$. Let $\alpha_i = \nu^{-1}(\bar{w}_i)$ for $0 \leq i \leq n$, where $w_i = x_1 x_2 \dots x_i$, and $\bar{w}_i$ is the element of the semigroup defined by w_i, with $\bar{w}_0 = 1$.

For every $x \in X$, the function $f_x : R \to R$ defined by $f_x(\alpha) = \nu^{-1}(\nu(\alpha)x)$ is FA-recognisable, as it takes the word in the second component of R_x which corresponds to α in the first component. As ν is a bijection, there is precisely one such word. Hence by Lemma 2.4.6, given $\alpha \in R$ we can compute $\beta = f_x(\alpha)$ in time $C_x|\alpha|$ for some constant C_x, and there is a constant K_x such that $|f_x(\alpha)| \leq |\alpha| + K_x$ for every $\alpha \in R$. Now as X is finite, we let $C = \max C_x$ and $K = \max K_x$. Then $|\alpha_{i+1}| \leq |\alpha_i| + K \leq |\alpha_0| + (i+1)K$. Thus we may compute α_n in time $\sum_{i=1}^{n} C(|\alpha_0| + iK) = \mathcal{O}(n^2)$.

Hence, to check if two elements $\bar{w}$ and $\bar{v}$ are equal, we may compute their representatives α and β in quadratic time, and then input (α, β) into the automaton recognising $R_=$. $\square$

In fact, we will see in Subsection 3.2.1 that we may generalise this further to show that all graph automatic semigroups have decidable word problem, by converting our semigroup into a monoid whilst preserving graph automaticity.

Note that automatic semigroups also have word problems which are decidable in quadratic time [14].

Chapter 3

Subsemigroups and Extensions

In this chapter we will consider various types of substructures of graph automatic semigroups. We will begin by examining how graph automaticity is preserved by subsemigroups and ideals. We then consider how graph automaticity is preserved when adding or removing finitely many elements, beginning with identities and zeroes, then continuing on to look at more general small extensions.

3.1 Subsemigroups and Ideals

We consider subsemigroups of graph automatic semigroups. We shall see that, similarly to both automatic semigroups and graph automatic groups, there are certain situations when we can ensure that graph automaticity is preserved. We begin by defining regular subsemigroups, generalising the definition of regular subgroups as defined in [25].

Definition 3.1.1. Let S be a graph automatic semigroup with graph automatic structure (X, Σ, R, ν). A finitely generated subsemigroup T of S is *regular* if

$$\nu^{-1}T = \bigcup_{t \in T} \nu^{-1}(t) \subseteq R$$

is a regular language.

In [25] it is shown that that regular subgroups of graph automatic groups are themselves graph automatic. We now see that this also holds in the semigroup case.

Theorem 3.1.2. *Let S be a graph automatic semigroup with graph automatic structure (X, Σ, R, ν). Let T be a finitely generated regular subsemigroup of S. Then T is graph automatic.*

Proof. Let T be generated by a finite set Y. By Theorem 2.4.3 we have a graph automatic structure for S with respect to any finite generating set, and so we may assume that $Y \subseteq X$. Let $L = \nu^{-1}T$. Then L is regular by assumption, and $\nu|_L : L \to T$ is onto. Now

$$L_= = R_= \cap (L \times L)$$

and

$$L_y = R_y \cap (L \times L)$$

for $y \in Y$ are regular, so $(Y, \Sigma, L, \nu|_L)$ is a graph automatic structure for T. □

If a subsemigroup of a graph automatic semigroup is not regular, it does not necessarily mean that it is not graph automatic, as we see in the following example.

Example 3.1.3. Let $S = \mathbb{N}_0 \times \mathbb{N}_0$ under addition. We shall see that S is graph automatic, but has a subsemigroup which is not regular, despite being graph automatic in its own right.

We first show that S is graph automatic. Let $X = \{(0,0), (1,0), (0,1)\}$. This is a generating set for S. Let $\Sigma = \{a, b\}$ and $R = a^*b^*$. We define $\nu : R \to S$ by

$$\nu(a^m b^n) = (m, n),$$

which is a bijective map. Hence

$$R_= = \{(\alpha, \alpha) : \alpha \in R\}$$

is regular, and $R_{(0,0)} = R_=$ is regular. Next we have that

$$\begin{aligned} R_{(1,0)} &= \{(a^i b^j, a^k b^l) \in R \times R : k = i + 1 \text{ and } l = j\} \\ &= \{(a^i b^j, a^{i+1} b^j) : i, j \in \mathbb{N}_0\} \\ &= (a, \$)(a, a)^*(b, b)^*, \end{aligned}$$

which is regular. Finally we have that

$$\begin{aligned} R_{(0,1)} &= \{(a^i b^j, a^k b^l) \in R \times R : k = i \text{ and } l = j + 1\} \\ &= \{(a^i b^j, a^i b^{j+1}) : i, j \in \mathbb{N}_0\} \\ &= (a, a)^*(b, b)^*(\$, b), \end{aligned}$$

which is regular. Thus we have that $S = \mathbb{N}_0 \times \mathbb{N}_0$ has graph automatic structure $(X, \{a, b\}, a^*b^*, \nu)$.

Now we show that S has a non-regular subsemigroup. Consider the subsemigroup

$$T = \{(n, n) : n \in \mathbb{N}_0\} \subset S.$$

Then we have that

$$\nu^{-1}T = a^n b^n$$

and it is well known that this is not a regular language (see [31], for example). Hence T is not a regular subsemigroup. However, T is isomorphic to $\mathbb{N}_0$, which does have a graph automatic structure, as we will see in Section 3.2. Thus we see that it is possible to have a graph automatic subsemigroup which is not a regular subsemigroup with respect to our original graph automatic structure.

This example shows that if a subsemigroup is not regular it does not necessarily mean that it is not graph automatic. So it may be possible to find a graph automatic structure for the subsemigroup which is completely unrelated to that of the semigroup. However, this also means that it is difficult to say that such a subsemigroup is not graph automatic, as it is necessary to rule out every possible choice of language and surjection. Thus it is difficult to comment on graph automaticity of non-regular subsemigroups, and we focus on situations where our subsemigroup can be shown to always be regular.

One case in which a subsemigroup is always regular is when it is a finitely generated left ideal, as we see below.

Theorem 3.1.4. *If a finitely generated subsemigroup of a graph automatic semigroup is a left ideal, then it is a regular subsemigroup.*

Proof. Let S be graph automatic with structure (X, Σ, R, ν). Let T be a left ideal of S generated as a subsemigroup by a finite subset Y. Without loss of generality we may assume that $Y \subseteq X$. Recall that

$$\begin{aligned} R_x^{(2)} &= \{\alpha \in R : \nu(\alpha) \in Sx\} \\ &= \{\alpha \in \Sigma^* : (\beta, \alpha) \in R_x \text{ for some } \beta \in \Sigma^*\} \end{aligned}$$

is the second component of R_x, and so is regular. Then

$$\nu^{-1}Y \cup \bigcup_{y \in Y} R_y^{(2)} = \{\alpha \in R : \nu(\alpha) \in T\}$$

is regular, as $\nu^{-1}Y$ is finite and each $R_y^{(2)}$ is regular. □

It then follows that finitely generated left ideals of graph automatic semigroups are themselves graph automatic.

Corollary 3.1.5. *Finitely generated left ideals of graph automatic semigroups are graph automatic.*

Proof. A finitely generated left ideal of a graph automatic semigroup is regular, by Theorem 3.1.4, and hence is graph automatic, by Theorem 3.1.2. □

It is natural to ask whether an analogous result holds for right ideals.

Question 3.1.6. Are finitely generated right ideals of graph automatic semigroups necessarily graph automatic?

Note that the proof of Theorem 3.1.4 relies on the fact that the automata recognising multiplication by elements in the left ideal must only accept representatives of elements of the left ideal on the second tape. This results from our automata accepting languages that represent multiplication by a generator on the right. Thus we cannot use the same method for right ideals, and so in order to answer this question it would be necessary to take a different approach. In fact, it may be possible that even if right ideals are graph automatic they may not be regular like left ideals.

Question 3.1.7. Are finitely generated right ideals of graph automatic semigroups always regular?

3.2 Zeros and Identities

In this section we begin to examine the effect of adding and removing elements to a graph automatic semigroup. We consider what happens when we add and remove certain distinguished elements, namely identities and zeroes.

Note that adding and removing identities and zeroes preserves both automaticity and FA-presentability of semigroups (see [14] and [10] respectively). Thus it seems likely that the same results will hold for graph automatic semigroups.

We begin by considering the case where we add and remove a zero element. We may add a zero element to any semigroup S by taking an element z which does not belong to S and defining multiplication for $s, t \in S \cup \{z\}$ by $st = z$ if $s = z$ or $t = z$ or both, and multiplication is as in S otherwise. We denote this new semigroup by S^0.

We examine the relationship between S and S^0 when one or the other is graph automatic, and see that graph automaticity of one implies that the other is also graph automatic.

Proposition 3.2.1. *A semigroup S is graph automatic if and only S^0 is graph automatic.*

Proof. Suppose S is graph automatic with structure (X, Σ, R, ν). Consider $S^0 = S \cup \{z\}$, which is generated by $X' = X \cup \{z\}$. Let $\Sigma' = \Sigma \cup \{\zeta\}$ for some $\zeta \notin \Sigma$ and let $R' = R \cup \{\zeta\}$. As R is regular we have that R' is also regular. Define $\nu' : R' \to S^0$ by

$$\nu'(\alpha) = \begin{cases} \nu(\alpha), & \alpha \in R \\ z, & \alpha = \zeta \end{cases}$$

and note that surjectivity of ν implies that ν' is also surjective.

Now

$$\begin{aligned} R'_= &= \{(\alpha, \beta) \in R' \times R' : \nu'(\alpha) = \nu'(\beta)\} \\ &= R_= \cup \{(\zeta, \zeta)\}, \end{aligned}$$

which is regular. We also have that

$$\begin{aligned} R'_z &= \{(\alpha, \beta) \in R' \times R' : z = \nu'(\beta)\} \\ &= \{(\alpha, \zeta) : \alpha \in R'\} \\ &= R' \times \{\zeta\}, \end{aligned}$$

which is regular. Finally, for $x \in X$ we have

$$R'_x = \{(\alpha, \beta) \in R' \times R' : \nu'(\alpha)x = \nu'(\beta)\}$$

$$= R_x \cup \{(\zeta, \zeta)\},$$

as for $\alpha \neq \zeta$ we cannot have $\nu'(\alpha)x = \nu'(\zeta) = z$ or $\nu'(\alpha) = \nu'(\zeta)x = z$. Thus R_x is regular, and S^0 is graph automatic with structure (X', Σ', R', ν').

Conversely, suppose S^0 is graph automatic with structure (X, Σ, R, ν). Let

$$\begin{aligned} Z =& \nu^{-1}(z) \\ =& \{\alpha \in R : \nu(\alpha) = z\}. \end{aligned}$$

This is regular by Proposition 2.4.4. Let $Y = X \setminus \{z\}$, so Y generates S. As Z is regular $R \setminus Z$ is also regular and thus S is a regular subsemigroup of S^0. Hence, by Theorem 3.1.2, we have that S is also graph automatic with graph automatic structure $(Y, \Sigma, R \setminus Z, \nu|_{R \setminus Z})$. □

Next, we consider the analogous case for identities. We may add an identity element to any semigroup S by taking an element e which is not an element of S and defining multiplication for $s \in S \cup \{e\}$ by $se = es = s$ and multiplication is as in S otherwise. We denote this new semigroup by S^1.

We consider the relationship between S and S^1, and see that graph automaticity is again preserved by the addition or removal of the identity.

Proposition 3.2.2. *A semigroup S is graph automatic if and only if S^1 is graph automatic.*

Proof. Let S have graph automatic structure (X, Σ, R, ν). Then $S^1 = S \cup \{e\}$ is generated by $X' = X \cup \{e\}$. Let $\Sigma' = \Sigma \cup \{\eta\}$ for some $\eta \notin \Sigma$. Then $R' = R \cup \{\eta\}$ is regular. Now define $\nu' : R' \to S^1$ by

$$\nu'(\alpha) = \begin{cases} \nu(\alpha), & \alpha \in R \\ e, & \alpha = \eta \end{cases}$$

and note that ν' is surjective due to the surjectivity of ν. Now

$$\begin{aligned} R'_= &= \{(\alpha,\beta) \in R' \times R' : \nu'(\alpha) = \nu'(\beta)\} \\ &= R_= \cup \{(\eta,\eta)\} \end{aligned}$$

as we cannot have $\nu(\eta) = \nu(\alpha)$ for $\alpha \in R$. So $R_=$ is regular. Then we have that

$$\begin{aligned} R'_e =&\{(\alpha,\beta) \in R' \times R' : \nu'(\alpha)e = \nu'(\beta)\} \\ =&R'_= \end{aligned}$$

is regular also. Finally, for $x \in X$ we have

$$\begin{aligned} R'_x &= \{(\alpha,\beta) \in R' \times R' : \nu'(\alpha)x = \nu'(\beta)\} \\ &= R_x \cup \{(\alpha,\eta) \in R \times \{\eta\} : \nu(\alpha)x = e\} \cup \{(\eta,\beta) \in \{\eta\} \times R : x = \nu(\beta)\}. \end{aligned}$$

But $e = \nu(\alpha)x$ is impossible as $\nu(\alpha)$ and x are both elements of S but $e \notin S$, so this second set is empty. Then, as

$$\{(\eta,\beta) : \beta \in R, x = \nu(\beta)\} = \{\eta\} \times \{\beta \in R : \nu(\beta) = x\}$$

is regular by Proposition 2.4.4 and R_x is regular, we have that R'_x is regular. Thus S^1 is graph automatic with structure (X', Σ', R', ν').

Conversely, suppose that S^1 is graph automatic with structure (X, Σ, R, ν). Then $Y = X \setminus \{1\}$ generates S. Let

$$L = \{\alpha \in R : \nu(\alpha) = e\}.$$

This is regular by Proposition 2.4.4. Then $R \setminus L$ is a regular language representing S, and so S is a regular subsemigroup of S^1. Hence S is graph automatic by Theorem 3.1.2, with structure $(Y, \Sigma, R \setminus L, \nu|_{R\setminus L})$ □

Thus we have seen that adding and removing both zeroes and identities preserves graph automaticity. These results now allow us to form new examples of graph automatic semigroups by adding zeroes and identities to the examples given in Chapter 2. In particular, we may now easily show that finitely generated free monoids are graph automatic.

Example 3.2.3. Let M_X be the free monoid generated by a finite set X. Then M_X is the free semigroup generated by X with an identity adjoined. We have seen that finitely generated free semigroups are graph automatic in Example 2.1.4. Thus by Proposition 3.2.2, we have that M_X is also graph automatic.

3.2.1 The Word Problem for Semigroups

We may also use Proposition 3.2.2 to extend our result from Chapter 2 regarding decidability of the word problem for graph automatic monoids.

We have seen that adding and removing identities preserves graph automaticity. We may now use this fact to generalise Proposition 2.4.7, which states that graph automatic monoids have decidable word problem, in order to show that graph automatic semigroups also have decidable word problem. Note that this approach echoes that of [14], where the authors show that automatic monoids have decidable word problem and then use the analogous result to Proposition 3.2.2 for automatic semigroups to generalise the result to semigroups.

Theorem 3.2.4. *The word problem in graph automatic semigroups is decidable in quadratic time.*

Proof. Let S be a graph automatic semigroup with structure (X, Σ, R, ν). We may adjoin an identity to S, preserving graph automaticity by Proposition 3.2.2. Then S^1 has graph automatic structure (X', Σ', R', ν'). We now have a graph automatic monoid, which has decidable word problem by Proposition 2.4.7. Given strings $w = x_1x_2 \dots x_n$ and $v = y_1y_2 \dots y_m$ for some $x_i, y_i \in X$, we may find representatives $\alpha, \beta \in R \subset R'$ for $\bar{w}$ and $\bar{v}$ in quadratic time, and check their equality by inputting (α, β) into the automaton recognising $(R')_=$. These represent the same element of S if and only if they represent the same element of S^1, and so the word problem of S is also decidable in quadratic time. □

Note that this provides us with one way of showing that a semigroup is not graph automatic. Any semigroup whose word problem is not solvable in quadratic time cannot be graph automatic.

3.3 Large Subsemigroups and Small Extensions

In the previous section we have seen that we may add and remove identity and zero elements of our semigroup and preserve graph automaticity. We now expand on this, and consider what happens when we add or remove finitely many elements. In particular, we consider subsemigroups of finite Rees index, and consider whether such constructions preserve graph automaticity.

Let S be a semigroup with a subsemigroup T. We call $|S \setminus T|$ the *Rees index* of T in S. If T has finite Rees index in S then we call S a *small extension* of T, and say that T is a *large subsemigroup* of S.

It is shown in [21] that a semigroup containing a subsemigroup of finite Rees index is automatic if and only if the subsemigroup of finite Rees index is also automatic. Large subsemigroups of FA-presentable semigroups are also FA-presentable, as shown in [10]. However, FA-presentability is not preserved by small extensions. An example is given in [10] of a small extension which does not preserve FA-presentability. Note that this example is not finitely generated, and thus is not graph automatic. We will now examine whether large subsemigroups and small extensions preserve graph automaticity.

We begin by considering whether large subsemigroups inherit the property of being graph automatic.

Theorem 3.3.1. *Let S be a graph automatic semigroup and T be a subsemigroup of finite Rees index in S. Then T is graph automatic.*

Proof. Let S have graph automatic structure (X, Σ, R, ν). Let $A = S \setminus T$. Then

$$K = \{\alpha \in R : \nu(\alpha) \in A\}$$

$$= \bigcup_{a \in A} \{\alpha \in R : \nu(\alpha) = a\}$$

is regular as A is finite and for each $a \in A$ the set $\{\alpha \in R : \nu(\alpha) = a\}$ is regular by Proposition 2.4.4. So

$$\begin{aligned} \nu^{-1}T &= \{\alpha \in R : \nu(\alpha) \notin A\} \\ &= R \setminus K \end{aligned}$$

is regular. Thus by Theorem 3.1.2, T is graph automatic. □

Next we consider small extensions. Note that small extensions can be viewed as the semigroup analogue of finite group extensions. In [25], the authors show that finite extensions of graph automatic groups are graph automatic. It may therefore be expected that a similar result holds in the semigroup case, and small extensions preserve graph automaticity. However, this is not immediately clear, as it is difficult to show that our small extension is regular without knowing how the adjoined elements interact with the original elements of the semigroup. Thus we cannot at present say that small extensions of graph automatic semigroups are always graph automatic.

In some specific cases we are able to show that graph automaticity is preserved, if there are restrictions placed on how our elements can interact. In particular, when our large subsemigroup is a right ideal then we see that graph automaticity is preserved.

Proposition 3.3.2. *Let S be a semigroup with a graph automatic right ideal of finite Rees index. Then S is graph automatic.*

Proof. Let T be a graph automatic right ideal of finite Rees index in S, with graph automatic structure (X, Σ, R, ν). Let

$$S \setminus T = C = \{c_1, c_2, \dots, c_n\},$$

and let $\bar{C} = \{\bar{c}_1, \bar{c}_2, \dots, \bar{c}_n\}$ be a set of symbols in one-to-one correspondence with C such that $\bar{C} \cap \Sigma = \emptyset$. Then S is generated by $X \cup C$ and $\bar{R} = R \cup \bar{C}$ is a regular language over the alphabet $\Sigma \cup \bar{C}$. Define $\bar{\nu} : \bar{R} \to S$ by

$$\bar{\nu}(\alpha) = \begin{cases} \nu(\alpha), & \alpha \in R \\ c_i, & \alpha = \bar{c}_i \end{cases}$$

and note that $\bar{\nu}$ is surjective by the surjectivity of ν. Now

$$\begin{aligned} \bar{R}_= &= \{(\alpha, \beta) \in \bar{R} \times \bar{R} : \bar{\nu}(\alpha) = \bar{\nu}(\beta)\} \\ &= R_= \cup \{(\bar{c}_i, \bar{c}_i) : \bar{c}_i \in \bar{C}\} \end{aligned}$$

is regular. Let $x \in X$. Then

$$\begin{aligned} \bar{R}_x =& \{(\alpha, \beta) \in \bar{R} \times \bar{R} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\} \\ =& R_x \cup \{(\alpha, \beta) \in \bar{C} \times \bar{C} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\} \\ & \cup \bigcup_{1 \leq i \leq n} (\{\bar{c}_i\} \times \{\beta \in R : c_i x = \nu(\beta)\}) \end{aligned}$$

Then R_x is regular, $\{(\alpha, \beta) \in \bar{C} \times \bar{C} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\}$ is finite, and for each $1 \leq i \leq n$ we have that $\{\beta \in R : c_i x = \nu(\beta)\}$ is empty if $c_i x \notin T$, or is regular by Proposition 2.4.4 if $c_i x \in T$. Thus $\bar{R}_x$ is regular. Note that we cannot have $\bar{\nu}(\alpha)x = \bar{\nu}(\beta)$ for $\alpha \in R$ and $\beta \in \bar{C}$ because T is a subsemigroup, hence is closed. Now let $c \in C$. Then

$$\begin{aligned} \bar{R}_c =& \{(\alpha, \beta) \in \bar{R} \times \bar{R} : \bar{\nu}(\alpha)c = \bar{\nu}(\beta)\} \\ =& R_c \cup \{(\alpha, \beta) \in \bar{C} \times \bar{C} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\} \\ & \cup \bigcup_{1 \leq i \leq n} (\{\bar{c}_i\} \times \{\beta \in R : c_i c = \nu(\beta)\}) . \end{aligned}$$

Note that we cannot have $\bar{\nu}(\alpha)c = \bar{\nu}(\beta)$ for $\alpha \in R$ and $\beta \in \bar{C}$ because T is a right ideal, and that for each $1 \leq i \leq n$ we again have that $\{\beta \in R : c_i c = \nu(\beta)\}$

is empty if $c_i c \notin T$ and is regular by Proposition 2.4.4 otherwise. Then

$$\begin{aligned}
R_c =& \{(\alpha,\beta) \in R \times R : \nu(\alpha)c = \nu(\beta)\} \\
=& \bigcup_{x \in X} \{(\alpha,\beta) \in R \times R : \exists \gamma \in R \text{ such that } \nu(\gamma)x = \nu(\alpha) \text{ and } \nu(\gamma)xc = \nu(\beta)\} \\
& \cup \{(\alpha,\beta) \in \nu^{-1}X \times R : \nu(\alpha)c = \nu(\beta)\} \\
=& \bigcup_{x \in X} \Big(\{(\alpha,\beta) \in R \times R : \exists \gamma \in R \text{ such that } (\alpha,\gamma) \in \hat{R}_x \text{ and } (\gamma,\beta) \in R_{xc}\} \\
& \cup \big(\nu^{-1}(x) \times \nu^{-1}(xc)\big) \Big)
\end{aligned}$$

where $\hat{R}_x = \{(\alpha,\gamma) \in R \times R : \nu(\alpha) = \nu(\gamma)x\}$. Note that $\hat{R}_x$ is regular, as it is merely the language R_x with the components exchanged, and as x and xc are fixed, each of the languages $\nu^{-1}(x)$ and $\nu^{-1}(xc)$ are regular by Proposition 2.4.4. Then as T is a right ideal, we have that $xc \in T$ so R_{xc} is regular. Thus R_c is regular. This means that $\bar{R}_c$ is also regular, as it is a union of finitely many regular languages. Hence $(X \cup C, \Sigma \cup \bar{C}, \bar{R}, \bar{\nu})$ is a graph automatic structure for S. □

Comparing this with Corollary 3.1.5, we note that the asymmetric nature of the definition of graph automaticity results in the situation where we can restrict a graph automatic semigroup to a left ideal and preserve graph automaticity, and can extend a right ideal to preserve graph automaticity. However we cannot at present say whether these results hold the other way round.

Given an arbitrary small extension of a graph automatic semigroup, we cannot show that the extension is graph automatic in the same way. If we take an infinite semigroup T with graph automatic structure (X, Σ, R, ν) and adjoin elements $C = \{c_1, \ldots, c_n\}$ then a problem arises as we do not know how the elements in C multiply with elements from T. Thus showing that the language accepting multiplication by an element from C is regular becomes problematic. In particular, for a fixed $c_1, c_2 \in C$ there may be infinitely many solutions to $tc_1 = c_2$ and so we cannot ensure that the language $L = \{\alpha \in R : \nu(\alpha)c_1 = c_2\}$

is regular. Similarly, if T is an arbitrary semigroup we are unable to show that the language $K = \{(\alpha, \beta) \in R \times R : \nu(\alpha)c = \nu(\beta)\}$ is regular.

To work around these problems we introduce some additional conditions on our small extension. In the case where T is a group we may use the fact that $\nu(\alpha)$ has an inverse to show that the set K must be empty, else $\nu(\alpha)^{-1}\nu(\beta) = c \in T$ which is a contradiction. To ensure that the set L is regular, we consider the case where our semigroup is of finite geometric type, which will force L to be finite, hence regular.

A semigroup S is said to be of *finite geometric type* if it is finitely generated and for every $p \in S$ there exists $k \in \mathbb{N}$ such that the equation $xp = q$ has at most k solutions for every $q \in S$. This concept was introduced in [30] in order to provide a geometric characterisation of automatic monoids in a similar way to the geometric characterisation of automatic groups.

Note that if a finitely generated semigroup is of finite geometric type, this means that the in-degree of each vertex of the Cayley graph is finite.

We now show that with these restrictions, we can demonstrate the graph automaticity of a small extension.

Proposition 3.3.3. *Let S be a semigroup of finite geometric type with a subgroup T of finite Rees index in S. If T is graph automatic then S is graph automatic.*

Proof. Let T be a subgroup of finite index in S, with graph automatic structure with uniqueness (X, Σ, R, ν). Let

$$S \setminus T = C = \{c_1, \ldots, c_n\}$$

and let $\bar{C}$ be a set of symbols in one-to-one correspondence with C such that $\Sigma \cap \bar{C} = \emptyset$. Now $\bar{X} = X \cup C$ is a finite generating set for S. Let $\bar{\Sigma} = \Sigma \cup \bar{C}$ and

$\bar{R} = R \cup \bar{C}$. Then $\bar{R}$ is regular and $\bar{\nu} : \bar{R} \to S$ defined by

$$\bar{\nu}(\alpha) = \begin{cases} \nu(\alpha), & \alpha \in R \\ c_i, & \alpha = \bar{c}_i \end{cases}$$

is surjective by the surjectivity of ν.

Now

$$\bar{R}_= = R_= \cup \{(\bar{c}_i, \bar{c}_i) : \bar{c}_i \in \bar{C}\}$$

is regular. Let $x \in X$. Then

$$\begin{aligned} \bar{R}_x =& R_x \cup \{(\alpha, \beta) \in R \times \bar{C} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\} \\ & \cup \{(\alpha, \beta) \in \bar{C} \times R : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\} \\ & \cup \{(\alpha, \beta) \in \bar{C} \times \bar{C} : \bar{\nu}(\alpha)x = \bar{\nu}(\beta)\}. \end{aligned}$$

The first of these sets is regular because T is graph automatic. The second is empty, as multiplying two elements of T cannot give an element of $S \setminus T$. The third set equals

$$\bigcup_{c_i} (\{\bar{c}_i\} \times \{\beta \in R : c_i x = \bar{\nu}(\beta)\}),$$

which is regular as either $c_i x \in T$ and regularity follows from Proposition 2.4.4, or this set is empty. Finally, the fourth set is finite and so is regular. Thus $\bar{R}_x$ is regular.

Now let $c \in C$. Then

$$\begin{aligned} \bar{R}_c =& R_c \cup \{(\alpha, \beta) \in R \times \bar{C} : \bar{\nu}(\alpha)c = \bar{\nu}(\beta)\} \\ & \cup \{(\alpha, \beta) \in \bar{C} \times R : \bar{\nu}(\alpha)c = \bar{\nu}(\beta)\} \\ & \cup \{(\alpha, \beta) \in \bar{C} \times \bar{C} : \bar{\nu}(\alpha)c = \bar{\nu}(\beta)\}. \end{aligned}$$

Then R_c is empty as T is a subgroup, so we cannot have $\nu(\alpha)c = \nu(\beta)$ or we would have $c = \nu(\alpha)^{-1}\nu(\beta) \in T$, a contradiction. The second set is equal to

$$\bigcup_{c_i}(\{\alpha \in R : \bar{\nu}(\alpha)c = c_i\} \times \{\bar{c}_i\}),$$

which is regular as there are only finitely many α satisfying $\bar{\nu}(\alpha)c = c_i$ because S is of finite geometric type. The latter two sets are regular as in the case of $\bar{R}_x$. Hence $\bar{R}_c$ is regular, and so S is graph automatic with structure $(\bar{X}, \bar{\Sigma}, \bar{R}, \bar{\nu})$. □

Thus we have certain situations where we can ensure that small extensions preserve graph automaticity. However, it remains to be seen whether this is the case in general.

Question 3.3.4. Are small extensions of graph automatic semigroups always graph automatic?

Chapter 4

Constructions for Graph Automatic Semigroups: Products

In this chapter we consider how a number of semigroup constructions preserve graph automaticity. In particular we examine constructions which can be considered as products of semigroups. We look at free products, semidirect products, direct products and Bruck-Reilly extensions. We also compare these results to the parallel results for automatic and FA-presentable semigroups.

4.1 Free Products

We begin by considering the free product of two graph automatic semigroups, and see that this preserves graph automaticity.

Theorem 4.1.1. *If S_1 and S_2 are graph automatic semigroups then the free product $S_1 * S_2$ is graph automatic.*

Proof. We follow the idea of the proof of Theorem 6.1 in [14]. Let $(X_1, \Sigma_1, R_1, \nu_1)$ and $(X_2, \Sigma_2, R_2, \nu_2)$ be graph automatic structures with uniqueness, with $\epsilon \notin$

R_1, R_2, for the semigroups S_1 and S_2 respectively. Without loss of generality we may assume $\Sigma_1 \cap \Sigma_2 = \emptyset$. Let $X = X_1 \cup X_2$, so X generates $S = S_1 * S_2$. Let $\Sigma = \Sigma_1 \cup \Sigma_2$ and define $R \subseteq \Sigma^*$ to be

$$R = (R_1 \cup \{\epsilon\})(R_2R_1)^*(R_2 \cup \{\epsilon\}) \setminus \{\epsilon\}.$$

This is the language consisting of strings where words from R_1 and R_2 alternate, and is regular. Now define $\nu : R \to S$ by

$$\nu(\alpha) = \begin{cases} \nu_1(\alpha_1)\nu_2(\alpha_2)\cdots\nu_{(n \bmod 2)+2}(\alpha_n), & \alpha_1 \in R_1 \\ \nu_2(\alpha_1)\nu_1(\alpha_2)\cdots\nu_{(n \bmod 2)+1}(\alpha_n), & \alpha_1 \in R_2 \end{cases}$$

for $\alpha = \alpha_1\alpha_2\cdots\alpha_n$ with $\alpha_i \in R_1$ if and only if $\alpha_{i+1} \in R_2$ for $1 \leq i \leq n-1$. As ν_1 and ν_2 are surjections we have that ν is also surjective, and as each $s_1 \in S_1$ and $s_2 \in S_2$ is represented by a unique element of R_1 and R_2 respectively we must have that each $s \in S$ is represented by a unique element of R. Thus

$$R_= = \{(\alpha, \alpha) : \alpha \in R\}$$

is regular. Now let

$$K_1 = (R_1 \cup \{\epsilon\})(R_2R_1)^* \setminus \{\epsilon\}$$

and

$$K_2 = (R_1 \cup \{\epsilon\})(R_2R_1)^*R_2.$$

So K_1 is the set of words in R which end with a word from R_1, and K_2 is the set of words in R which end with a word from R_2. Then $K_1 \cap K_2 = \emptyset$ and $K_1 \cup K_2 = R$. Let $x \in X_1$ and let ξ be the unique word in R_1 which represents x. Then

$$\begin{aligned} R_x =& \{(\alpha, \beta) \in R \times R : \nu(\alpha)x = \nu(\beta)\} \\ =& (R_1)_x \cup \{(\alpha, \alpha) : \alpha \in K_2\}((R_1)_x \cup \{(\$, \xi)\}). \end{aligned}$$

So R_x is regular for $x \in X_1$. Similarly, if $y \in X_2$ and $\zeta \in R_2$ is the unique word representing y then

$$R_y = (R_2)_y \cup \{(\alpha, \alpha) : \alpha \in K_1\}((R_2)_y \cup \{(\$, \zeta)\}).$$

Thus (X, Σ, R, ν) is a graph automatic structure for S. □

In [25] it is shown that taking the group free product of two graph automatic groups preserves graph automaticity. Note that if we want to take the monoid free product then the proof of Theorem 10.8 from [25] for the group free product still holds for the monoid case. We reproduce the proof below.

Proposition 4.1.2. *The monoid free product of two graph automatic semigroups is graph automatic.*

Proof. Let M_1 and M_2 be graph automatic monoids with structures with uniqueness $(X_1, \Sigma_1, R_1, \nu_1)$ and $(X_2, \Sigma_2, R_2, \nu_2)$ respectively, such that $R_1 \cap R_2 = \{\epsilon\}$ and in each structure ϵ is the representative of the identity element, 1. Each non-identity element of $M_1 * M_2$ has a normal form $m_1 m_2 \ldots m_n$, where each $m_i \in M_1 \cup M_2$, with $m_i \neq 1$ and $m_i \in M_1$ if and only if $m_{i+1} \in M_2$, and $m_i \in M_2$ if and only if $m_{i+1} \in M_1$, for all $i \geq 1$. Now, as in Theorem 4.1.1, we let $X = X_1 \cup X_2$, let $\Sigma = \Sigma_1 \cup \Sigma_2$ and define $R \subseteq \Sigma^*$ to be

$$R = (R_1 \cup \{\epsilon\})(R_2 R_1)^*(R_2 \cup \{\epsilon\}).$$

Now define $\nu : R \to S$ by

$$\nu(\alpha) = \begin{cases} \nu_1(\alpha_1)\nu_2(\alpha_2)\cdots\nu_{(n \bmod 2)+2}(\alpha_n), & \alpha_1 \in R_1 \\ \nu_2(\alpha_1)\nu_1(\alpha_2)\cdots\nu_{(n \bmod 2)+1}(\alpha_n), & \alpha_1 \in R_2 \\ 1, & \alpha = \epsilon \end{cases}$$

for $\alpha = \alpha_1 \alpha_2 \cdots \alpha_n$ with $\alpha_i \in R_1$ if and only if $\alpha_{i+1} \in R_2$ for $1 \leq i \leq n-1$. Then the uniqueness of our graph automatic structures together with our normal forms give us that ν is a bijection and so

$$R_= = \{(\alpha, \alpha) : \alpha \in R\}$$

is regular. Now, similarly to the semigroup case, we let

$$K_1 = (R_1 \cup \{\epsilon\})(R_2R_1)^* \setminus \{\epsilon\}$$

and

$$K_2 = (R_1 \cup \{\epsilon\})(R_2R_1)^* R_2$$

and so we have that $K_1 \cup K_2 \cup \{\epsilon\} = R$. Then

$$R_x = \{(\alpha, \alpha) : \alpha \in K_2 \cup \{\epsilon\}\}((R_1)_x \cup \{(\epsilon, \xi)\}),$$

where $\nu_1(\xi) = x$, and

$$R_y = \{(\alpha, \alpha) : \alpha \in K_1 \cup \{\epsilon\}\}((R_2)_y \cup \{(\epsilon, \zeta)\})$$

where $\nu_2(\zeta) = y$. These are both regular languages, hence $M_1 * M_2$ is graph automatic. □

Note that taking the free product of two automatic semigroups preserves automaticity, and in this case the converse also holds, as shown in [14].

Proposition 4.1.3 (Theorem 6.1 of [14])**.** *Let S_1 and S_2 be semigroups. Then $S_1 * S_2$ is automatic if and only if both S_1 and S_2 are automatic.*

The equivalent result does not hold for FA-presentable semigroups. In fact, FA-presentability is only preserved by free products in the trivial case.

Proposition 4.1.4 (Proposition 4.1 of [10])**.** *The semigroup free product of two semigroups S_1 and S_2 is FA-presentable if and only if S_1 and S_2 are trivial.*

We may use these results to construct examples of semigroups which are graph automatic but neither automatic nor FA-presentable.

Example 4.1.5. Let S be the (semigroup) free product of the Heisenberg group $\mathcal{H}_3(\mathbb{Z})$ with a finite semigroup T. Then both $\mathcal{H}_3(\mathbb{Z})$ and T are graph automatic, and so their free product is graph automatic by Theorem 4.1.1. Now as $\mathcal{H}_3(\mathbb{Z})$ is not trivial, S cannot be FA-presentable by Proposition 4.1.4. Additionally, as $\mathcal{H}_3(\mathbb{Z})$ is not automatic, S cannot be automatic by Proposition 4.1.3. Hence S is graph automatic, but neither automatic nor FA-presentable.

We now ask whether graph automaticity of a free product implies that the original semigroups are graph automatic, as is the case for automatic semigroups. In [14], automaticity of the semigroups in the product is demonstrated by showing that their preimages are regular languages. However, when we do not have a homomorphism we cannot easily demonstrate that our semigroups are regular, so we ask:

Question 4.1.6. If the free product of two finitely generated semigroups is graph automatic, are the semigroups themselves graph automatic? If the monoid free product of two finitely generated monoids is graph automatic, are the monoids graph automatic?

4.2 Semidirect Products

We now consider semidirect products of graph automatic semigroups. We will see that graph automaticity is preserved by semidirect products under certain conditions. We begin by recalling the definitions of semigroup actions and semidirect products.

A semigroup S *acts* on a set X on the left if there is a function $\sigma : S \times X \to X$, defined as $\sigma(s,x) = {}^{s}x$, such that for any $x \in X$ and $s, t \in S$ we have

$$^{st}x = {}^{s}({}^{t}x).$$

If the arbitrary set X is replaced by a semigroup T, then we have that the action of each element of S is equivalent to an endomorphism of T, which is denoted $\tau(s) : T \to T$ and defined to be

$$\tau(s)(t) = {}^{s}t.$$

Let S and T be semigroups and $\tau : S \to \mathrm{End}(T)$ be a homomorphism from S into the endomorphism semigroup of T. The semidirect product $T \ltimes_\tau S$ of S and T over τ is the set $T \times S$ with multiplication

$$(t_1, s_1)(t_2, s_2) = (t_1({}^{s_1}t_2), s_1 s_2)$$

where ${}^{s}t$ denotes the left action of s on t.

We also define two homomorphisms, φ_1 and φ_2, which will be used throughout this section and the next section.

Definition 4.2.1. Let Σ_1 and Σ_2 be finite alphabets, and let

$$\Sigma = ((\Sigma_1 \cup \{\$\}) \times (\Sigma_2 \cup \{\$\})) \setminus \{(\$, \$)\}.$$

Then we define $\varphi_1 : \Sigma^* \times \Sigma^* \to \Sigma_1^* \times \Sigma_1^*$ and $\varphi_2 : \Sigma^* \times \Sigma^* \to \Sigma_2^* \times \Sigma_2^*$ by

$$\begin{aligned} \varphi_1 : &((\kappa_1, \lambda_1), (\kappa_2, \lambda_2)) \mapsto (\kappa_1, \kappa_2), \\ &((\kappa_1, \lambda_1), \$) \mapsto (\kappa_1, \$), \\ &(\$, (\kappa_2, \lambda_2)) \mapsto (\$, \kappa_2) \end{aligned}$$

and

$$\begin{aligned} \varphi_2 : &((\kappa_1, \lambda_1), (\kappa_2, \lambda_2)) \mapsto (\lambda_1, \lambda_2), \\ &((\kappa_1, \lambda_1), \$) \mapsto (\lambda_1, \$), \\ &(\$, (\kappa_2, \lambda_2)) \mapsto (\$, \lambda_2). \end{aligned}$$

Now if K and L are regular languages over the alphabets Σ_1 and Σ_2 respectively, and $R = K \times L$, then we have that R is also a regular language, and we may apply φ_1 and φ_2 to $R \times R$. Then $\varphi_1(R \times R) \subseteq K \times K$ and $\varphi_2(R \times R) \subseteq L \times L$ are regular languages, as are $\varphi_1^{-1}(K \times K) \subseteq R \times R$ and $\varphi_2^{-1}(L \times L) \subseteq R \times R$, as direct products, homomorphic images, and homomorphic preimages of regular languages all preserve regularity.

We now examine when graph automaticity is preserved by semidirect products.

Theorem 4.2.2. *Let S be a finite semigroup and let T be a graph automatic semigroup. If the semidirect product $T \ltimes_\tau S$ is finitely generated then it is graph automatic.*

Proof. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a finite semigroup, and let T be a graph automatic semigroup with graph automatic structure with uniqueness (Y, Σ_1, K, ν). As S is finite it is also graph automatic, with structure $(S, \Sigma_2 = \{\alpha_1, \ldots, \alpha_n\}, L = \{\alpha_1, \ldots, \alpha_n\}, \mu)$, where $\mu(\alpha_i) = s_i$. Let X be a finite generating set for $T \ltimes_\tau S$, let $\Sigma = ((\Sigma_1 \cup \{\$\}) \times (\Sigma_2 \cup \{\$\})) \setminus (\$, \$)$ and let $R = K \times L$. We define $\psi : R \to T \ltimes_\tau S$ by

$$\psi(\alpha) = (\nu(\kappa), \mu(\lambda))$$

for $\alpha = (\kappa, \lambda) \in R$. Then

$$R_= = \{(\alpha, \alpha) : \alpha \in R\}$$

is regular by the uniqueness of our original structures, and for $(a, b) \in X$ we have that

$$\begin{aligned} R_{(a,b)} &= \{((\beta, \alpha_i), (\gamma, \alpha_j)) \in R \times R : \nu(\beta)^{\mu(\alpha_i)}a = \nu(\gamma) \text{ and } \mu(\alpha_i)b = \mu(\alpha_j)\} \\ &= \{((\beta, \alpha_i), (\gamma, \alpha_j)) \in R \times R : (\beta, \gamma) \in K_{^{s_i}a} \text{ and } s_ib = s_j\} \\ &= \bigcup_{1 \le i \le n} (\varphi_1^{-1}(K_{^{s_i}a}) \cap \varphi_2^{-1}(\{\alpha_i\} \times \{\alpha_j \in L : s_ib = s_j\})) \end{aligned}$$

Now as S is finite, the set $\{\alpha_j \in L : s_ib = s_j\}$ is finite also. We also have that $\varphi_1^{-1}(K_{^{s_i}a})$ is regular, as the homomorphic preimage of a regular language is regular. Thus $R_{(a,b)}$ is a finite union of regular languages, thus is regular, and so we have that $T \ltimes_\tau S$ is graph automatic with structure (X, Σ, R, ψ). □

In [25] the authors show that graph automaticity is preserved under certain conditions when taking the semidirect product of two groups. Namely, graph automaticity is preserved when the automorphism involved in the semidirect product is automatic, that is the graph of the automorphism is a recognisable language. Thus we consider whether this same result will hold for semigroups. In the proof above, we relied on the fact that one of our semigroups was finite in order to show that our language recognising multiplication by a generator was regular. However, we cannot compare this directly with the group case, as in [25], they consider the semidirect product $S \rtimes_\tau T$, that is where S acts on T on the right. In the groups case, a right action is equivalent to a left action. However, for semigroups we must instead consider the semidirect product $S \rtimes_\tau T$ as a separate case. Here we replace our left action by a right action.

A semigroup S acts on a set X on the right if there is a function $\sigma : X \times S \to X$, defined as $\sigma(x, s) = x^s$, such that for any $x \in X$ and $s, t \in S$ we have

$$x^{st} = (x^s)^t.$$

As before, when X is a semigroup we have that the action of each element of S is equivalent to an endomorphism of X, and in this case we define the semidirect product to be the set $S \times T$ with multiplication

$$(s_1, t_1)(s_2, t_2) = (s_1 s_2, (t_1^{s_2})t_2).$$

In this case we consider the case when the endomorphisms involved in the semidirect product are recognisable.

In general, we will call a homomorphism *automatic* if the homomorphism can be represented by a recognisable language. More explicitly, let $\varphi : S \to T$ be a homomorphism, and let K and L be regular languages over the alphabets

Σ_1 and Σ_2 respectively, such that there exist surjective maps $\nu : K \to S$ and $\mu : L \to T$. Then φ is automatic if the set

$$\{(\alpha, \beta) \in K \times L : \varphi(\nu(\alpha)) = \mu(\beta)\}$$

is a regular language. In this case we say that φ is *automatic with respect to* (Σ_1, K, ν) *and* (Σ_2, L, μ). If the semigroups S and T have graph automatic structures (X, Σ_1, K, ν) and (Y, Σ_2, L, μ) respectively, we may say that φ is automatic with respect to the graph automatic structures of S and T. Note that if $S = T$ then we may omit the reference to the second structure.

In particular, in the case where S is a graph automatic semigroup with structure (X, Σ, K, ν), and our homomorphism is a right action on the semigroup S, we require that the language

$$\{(\alpha, \beta) \in K \times K : \nu(\alpha)^s = \nu(\beta)\}$$

is regular with respect to the graph automatic structure of S.

We now see that graph automaticity is preserved in the semigroup case if our endomorphism is automatic, analogously to the group case.

Theorem 4.2.3. *Let S and T be graph automatic semigroups such that the semidirect product $S \rtimes_\tau T$ is finitely generated by some set Y. For each $s \in S$ such that $(s, t) \in Y$ for some $t \in T$ let $\tau(s)$ be an automatic homomorphism with respect to the graph automatic structure of T. Then $S \rtimes_\tau T$ is graph automatic.*

Proof. Let S and T be graph automatic with graph automatic structures with uniqueness (X_1, Σ_1, L, μ) and (X_2, Σ_2, K, ν) respectively. Let Y be a finite generating set for $S \rtimes_\tau T$. Now let $\Sigma = ((\Sigma_1 \cup \{\$\}) \times (\Sigma_2 \cup \{\$\})) \setminus (\$, \$)$, let $R = L \times K$, and define $\psi : R \to S \rtimes_\tau T$ by

$$\psi(\alpha) = (\mu(\lambda), \nu(\kappa))$$

for $\alpha = (\lambda, \kappa) \in R$. Then

$$R_{=} = \{(\alpha, \alpha) : \alpha \in R\}$$

is regular by the uniqueness of our original structures, and for $(a, b) \in Y$ we have

$$\begin{aligned} R_{(a,b)} =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : \mu(\alpha_1)a = \mu(\beta_1) \text{ and } \nu(\alpha_2)^a b = \nu(\beta_2)\} \\ =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : (\alpha_1, \beta_1) \in L_a \text{ and there exists some} \\ & \gamma \in K \text{ such that } (\alpha_2, \gamma) \in E_a \text{ and } (\gamma, \beta_2) \in K_b\} \\ =& \varphi_1^{-1}(L_a) \cap \varphi_2^{-1}\{(\alpha_2, \beta_2) \in K \times K : \text{ there exists some } \gamma \in K \text{ such} \\ & \text{that } (\alpha_2, \gamma) \in E_a \text{ and } (\gamma, \beta_2) \in K_b\}, \end{aligned}$$

where E_a is the language recognising the endomorphism $\tau(a)$. As E_a is regular by assumption, we have that $R_{(a,b)}$ is regular. Hence (Y, Σ, R, ψ) is a graph automatic structure for $S \rtimes_\tau T$. □

Note that the differences in Propositions 4.2.2 and 4.2.3 come from the fact that in the definition of a graph automatic semigroup we are recognising multiplication on the right. The consequence of this is that the languages which we wish to recognise differ depending on whether we have a right action or a left action.

4.3 Direct Products

We now consider the direct product. Direct products are equivalent to the semidirect product where $\tau(s)$ is the identity map for all $s \in S$. However, we shall consider them in their own right. In [14] it is shown that the direct product of two automatic monoids is automatic, and in [12] this result is extended to show that the direct product of two automatic semigroups is automatic if and only if it is finitely generated. We will show that the analogous results holds for graph automatic semigroups. We begin by considering the monoid case, as the direct product of two graph automatic (and thus finitely generated) monoids is always finitely generated and we can easily find a finite generating set when attempting to find a graph automatic structure.

Proposition 4.3.1. *Let M_1 and M_2 be graph automatic monoids. Then $M_1 \times M_2$ is graph automatic.*

Proof. Let $(X_1, \Sigma_1, K, \nu_1)$ and $(X_2, \Sigma_2, L, \nu_2)$ be graph automatic structures with uniqueness for M_1 and M_2 respectively. Let $M = M_1 \times M_2$. If e_1 and e_2 are the identities of M_1 and M_2 respectively then we have that M is generated by $X = (X_1, e_2) \cup (e_1, X_2)$. Let $\Sigma = ((\Sigma_1 \cup \{\$\}) \times (\Sigma_2 \times \{\$\})) \setminus (\$, \$)$ and let $R = K \times L$. Define $\nu : R \to M$ by

$$\nu(\alpha, \beta) = (\nu_1(\alpha), \nu_2(\beta)).$$

Then as ν_1 and ν_2 are surjective ν is also surjective. Let φ_1 and φ_2 be the homomorphisms given in Definition 4.2.1.

Now

$$R_= = \{(\alpha, \alpha) : \alpha \in R\}$$

by the uniqueness of our original structures and so $R_=$ is regular. Let $y = (x, e_2) \in X$. Then

$$\begin{aligned} R_y =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : (\nu_1(\alpha_1)x, \nu_2(\alpha_2)) = (\nu_1(\beta_1), \nu_2(\beta_2))\} \\ =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : \nu_1(\alpha_1)x = \nu_1(\beta_1) \text{ and } \nu_2(\alpha_2) = \nu_2(\beta_2)\} \\ =& \varphi_1^{-1}(K_x) \cap \varphi_2^{-1}(L_=). \end{aligned}$$

Now let $z = (e_1, x) \in X$. Then

$$\begin{aligned} R_z =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : (\nu_1(\alpha_1), \nu_2(\alpha_2)x) = (\nu_1(\beta_1), \nu_2(\beta_2))\} \\ =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) \in R \times R : \nu_1(\alpha_1) = \nu_1(\beta_1) \text{ and } \nu_2(\alpha_2)x = \nu_2(\beta_2)\} \end{aligned}$$

$$=\varphi_1^{-1}(K_=) \cap \varphi_2^{-1}(L_x).$$

So R_x is regular for any $x \in X$ and (X, Σ, R, ν) is a graph automatic structure for M. □

In the semigroup case, the direct product is not necessarily finitely generated. Conditions for finite generation of direct products are given in [29]. In particular, we have:

Proposition 4.3.2 (Lemma 2.3 of [29]). *Let S and T be two semigroups. If T is infinite and $S \times T$ is finitely generated, then $S^2 = S$.*

This leads to the following result for the case where we have two infinite semigroups.

Proposition 4.3.3 (Theorem 2.1 of [29]). *Let S and T be infinite semigroups. Then $S \times T$ is finitely generated if and only if both S and T are finitely generated and $S^2 = S$ and $T^2 = T$.*

Furthermore, we have a way of finding a finite generating set for our direct product $S \times T$ based on the generating sets of S and T. To do so we require that our semigroups have a *full generating set*, that is a generating set A such that $A^2 \subseteq A$. If our direct product is finitely generated, then we can find such generating sets due to the following proposition.

Proposition 4.3.4 (Proposition 2.10 of [29]). *A semigroup S has a full generating set A if and only if $S^2 = S$. Furthermore, if S is finitely generated, A can be chosen to be finite.*

This now allows us to find our finite generating set for the direct product using the following proposition.

Proposition 4.3.5 (Corollary 2.11 of [29]). *Let S and T be two semigroups with $S^2 = S$ and $T^2 = T$, and let A and B be full generating sets for S and T respectively. Then the set $A \times B$ is a full generating set for $S \times T$. Moreover, if $S \times T$ is finitely generated, then the sets A and B can be chosen to be finite.*

We now may use these results to show that the finitely generated direct product of graph automatic semigroups is also graph automatic, and in particular we may use the generating sets for our original semigroups to find our new graph automatic structure.

Theorem 4.3.6. *Let S and T be graph automatic semigroups. Then $S \times T$ is graph automatic if and only if it is finitely generated.*

Proof. Let S and T be graph automatic semigroups with structures (A, Σ_1, K, ν) and (B, Σ_2, L, μ) respectively and suppose that $S \times T$ is finitely generated.

If S and T are both infinite and $S \times T$ is finitely generated then $S^2 = S$ and $T^2 = T$ and we may choose A and B in such a way that $A \times B$ is a finite generating set for $S \times T$ by Proposition 4.3.5. Let $\Sigma = ((\Sigma_1 \cup \{\$\}) \times (\Sigma_2 \times \{\$\})) \setminus (\$, \$)$ and $R = K \times L$. Define $\psi : R \to S \times T$ by

$$\psi(\alpha) = (\nu(\kappa), \mu(\lambda))$$

for $\alpha = (\kappa, \lambda) \in R$. As both ν and μ are surjective ψ is also a surjection. Let φ_1 and φ_2 be homomorphisms as defined in Definition 4.2.1. Now

$$\begin{aligned} R_= =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) : \nu(\alpha_1) = \nu(\beta_1) \text{ and } \mu(\alpha_2) = \mu(\beta_2)\} \\ =& \varphi_1^{-1} K_= \cap \varphi_2^{-1} L_= \end{aligned}$$

so $R_=$ is regular. Now let $(a, b) \in A \times B$. Then

$$\begin{aligned} R_{(a,b)} =& \{((\alpha_1, \alpha_2), (\beta_1, \beta_2)) : \nu(\alpha_1)a = \nu(\beta_1) \text{ and } \mu(\alpha_2)b = \mu(\beta_2)\} \\ =& \varphi_1^{-1} K_a \cap \varphi_2^{-1} L_b \end{aligned}$$

so $R_{(a,b)}$ is regular. Thus $(A \times B, \Sigma, R, \psi)$ is a graph automatic structure for $S \times T$.

Now suppose that S is finite and T is infinite. Then $S \times B$ is a finite generating set for $S \times T$ and the rest follows as in the previous case.

Finally, if S and T are finite then so is $S \times T$, thus it is graph automatic.

Conversely, suppose that $S \times T$ is not finitely generated. Then $S \times T$ cannot be graph automatic, as graph automaticity requires a finite generating set. □

It is natural to now ask whether the converse to this result holds: if we have a direct product $S \times T$ which is graph automatic, are the semigroups S and T graph automatic also? Note that this does not hold for FA-presentable semigroups, as a counterexample is given in [10].

We see that if the direct product of two monoids, where one is finite, is graph automatic, then we may show that both monoids must also be graph automatic.

Proposition 4.3.7. *Let M_1 and M_2 be monoids, with M_1 finite, such that $M_1 \times M_2$ is graph automatic. Then M_1 and M_2 are graph automatic.*

Proof. As M_1 is finite it is graph automatic. We will show that M_2 must also be graph automatic.

Let $M_1 \times M_2$ be graph automatic with structure with uniqueness (X, Σ, R, ν). Let

$$\pi : M_1 \times M_2 \to M_2$$

be the projection onto M_2. Then define $\mu = \pi \circ \nu : R \to M_2$. So μ is surjective, and M_2 is generated by $Y = \pi(X)$, which is finite as X is finite. We claim that (Y, Σ, R, μ) is a graph automatic structure for M_2.

Let

$$\bar{R}_= = \{(\alpha, \beta) \in R \times R : \mu(\alpha) = \mu(\beta)\}$$

and

$$\bar{R}_y = \{(\alpha, \beta) \in R \times R : \mu(\alpha)y = \mu(\beta)\}$$

for $y \in Y$. Note that we use the notation $\bar{R}_=$ and $\bar{R}_x$ to distinguish these languages which detect equality and multiplication of words from R under the map μ from $R_=$ and R_x which recognise equality and multiplication of words from R under the map ν. We now show that these are regular languages. As $\bar{R}_= = \bar{R}_1$ we need only consider the $\bar{R}_y$, as we may assume that $1 \in Y$.

Now as M_1 is finite, $\bigcup_{a\in M_1} R_{(a,1)}$ is a finite union of regular languages, thus is regular. Recall that

$$\hat{R}_x = \{(\alpha,\beta)\in R\times R : \nu(\alpha) = \nu(\beta)x\}.$$

Then $\hat{R}_x$ is regular if and only if R_x is regular and so we also have that $\bigcup_{a\in M_1} \hat{R}_{(a,1)}$ is a regular language. Let

$$\begin{aligned} W_y =& \{(\alpha,\beta)\in\Sigma^*\times\Sigma^* : \exists\gamma\in\Sigma^* \text{ such that } (\alpha,\gamma)\in\bigcup_{a\in M_1}\hat{R}_{(a,1)} \\ &\quad \text{and } (\gamma,\beta)\in\bigcup_{a\in M_1} R_{(a,y)}\} \\ =&\{(\alpha,\beta)\in R\times R : \exists\gamma\in R \text{ such that } (\alpha,\gamma)\in\bigcup_{a\in M_1}\hat{R}_{(a,1)} \\ &\quad \text{and } (\gamma,\beta)\in\bigcup_{a\in M_1} R_{(a,y)}\}. \end{aligned}$$

Note that W_y is regular. Now we claim that $(\alpha,\beta)\in\bar{R}_y$ if and only if $(\alpha,\beta)\in W_y$.

Let $(\alpha,\beta)\in\bar{R}_y$, with $\nu(\alpha) = (m_\alpha,n_\alpha)$ and $\nu(\beta) = (m_\beta,n_\beta)$. So $\mu(\alpha)y = \mu(\beta)$. Let $\gamma = \nu^{-1}(1,n_\alpha)$. Then

$$\begin{aligned} \nu(\alpha) &= (m_\alpha,n_\alpha) \\ &= (1,n_\alpha)(m_\alpha,1) \\ &= \nu(\gamma)(m_\alpha,1) \end{aligned}$$

so $(\alpha,\gamma)\in\hat{R}_{(m_\alpha,1)}$, and

$$\begin{aligned} \nu(\gamma)(m_\beta,y) &= (1,n_\alpha)(m_\beta,y) \\ &= (m_\beta,n_\alpha y) \\ &= (m_\beta,n_\beta) \\ &= \nu(\beta) \end{aligned}$$

so $(\gamma, \beta) \in R_{(m_\beta,1)}$. Thus $(\alpha, \beta) \in W_y$

Now let $(\alpha, \beta) \in W_y$. So there exist $\gamma \in R$ and $a, b \in M_1$ such that $\nu(\alpha) = \nu(\gamma)(a, 1)$ and $\nu(\gamma)(b, y) = \nu(\beta)$. So $(m_\alpha, n_\alpha) = (m_\gamma a, n_\gamma)$ and $(m_\gamma b, n_\gamma y) = (m_\beta, n_\beta)$. Thus

$$\mu(\alpha)y = n_\alpha y = n_\gamma y = n_\beta = \mu(\beta)$$

so $(\alpha, \beta) \in \bar{R}_y$.

Thus we have shown that $(\alpha, \beta) \in W_y$ if and only $(\alpha, \beta) \in \bar{R}_y$. Then as W_y is regular we must have that $\bar{R}_y$ is also regular, and so we have that $M_1 \times M_2$ is graph automatic with graph automatic structure (Y, Σ, R, μ). □

In fact we may extend this to the case where we have the direct product of a finite monoid with an infinite semigroup.

Corollary 4.3.8. *Let M be a finite monoid and S be a semigroup such that $M \times S$ is graph automatic. Then S is graph automatic.*

Proof. Let $(m_1, s_1) \in M \times S$ and $(m_2, s_2) \in M \times S^1$. Then $(m_1, s_1)(m_2, s_2) = (m_1 m_2, s_1 s_2) \in M \times S$ and so $M \times S$ is a right ideal in $M \times S^1$. As M is finite we have that $M \times S$ has finite Rees index in $M \times S^1$. So as $M \times S$ is graph automatic, $M \times S^1$ is also graph automatic by Proposition 3.3.2. Now we are in the monoid case, and so S^1 is graph automatic by Proposition 4.3.7. Hence S is graph automatic by Proposition 3.2.2. □

The question remains whether this result will hold in general for the direct product of two semigroups.

Question 4.3.9. If S and T are semigroups and $S \times T$ is graph automatic then are S and T necessarily graph automatic?

4.4 Bruck-Reilly Extensions

We now consider Bruck-Reilly extensions of monoids, and examine when they preserve graph automaticity.

Recall that the Bruck-Reilly extension of a monoid $M = \langle A \mid R \rangle$ determined by the homomorphism $\theta : M \to M$ is defined as

$$\mathrm{BR}(M, \theta) = \langle A, b, c \mid R, bc = 1, ac = c(\theta a), ba = (\theta a)b \text{ for } a \in A \rangle.$$

This gives us the set

$$\mathbb{N}_0 \times M \times \mathbb{N}_0$$

with multiplication defined as

$$(m, s, n)(p, t, q) = (m - n + k, \theta^{k-n}(s)\theta^{k-p}(t), q - p + k)$$

for $k = \max\{n, p\}$.

In particular, note that we can write $a = (0, a, 0)$ for $a \in A$, $b = (0, 1, 1)$, and $c = (1, 1, 0)$, and so multiplication by generators is defined as follows:

$$(m, s, n)(0, a, 0) = (m, s\theta^n(a), n)$$

$$(m, s, n)(0, 1, 1) = (m, s, 1 + n)$$

$$(m, s, n)(1, 1, 0) = \begin{cases} (m, s, n - 1), & n \geq 1 \\ (m + 1, \theta(s), 0), & n = 0 \end{cases}$$

for any $(m, s, n) \in M$.

In [8] it is shown that if a Bruck-Reilly extension is automatic then its base semigroup is automatic. We also have that if a Bruck-Reilly extension is FA-presentable, then its base semigroup also admits an automatic presentation, as shown in [10].

We consider whether this is also the case for graph automatic semigroups, and see that if θ is an automorphism then the base semigroup of a graph automatic Bruck-Reilly extension is also graph automatic.

Theorem 4.4.1. *Let T be a monoid and θ be an automorphism of T. Let*

$S = \mathrm{BR}(T, \theta)$ be graph automatic. Then T is also graph automatic.

Proof. Let (X, Σ, R, ν) be a graph automatic structure for S. Consider the second component of R_b, denoted $R_b^{(2)}$. This is a regular language which represents all left multiples of b. As every element of S can be expressed as $c^i t b^j$ for some $t \in T$ and $i, j \in \mathbb{N}_0$ we get that

$$K = R \setminus R_b^{(2)}$$

is a regular language representing all the elements of S where $j = 0$. Now consider

$$K_c = R_c \cap (K \times K).$$

The second component of this, $K_c^{(2)}$, is a regular language representing all elements of the form $c^i tc = c^{i+1}\theta(t)$ for $t \in T, i \in \mathbb{N}_0$. As θ is an automorphism, this is all elements of the form $c^i t$ where $i \geq 1$. Then

$$L = K \setminus K_c^{(2)}$$

is a regular language representing all elements of S where $i = j = 0$. Thus L is a regular language representing the subgroup T and so T has a graph automatic structure by Theorem 3.1.2. □

Note that we do not require θ to be an automorphism in order to preserve graph automaticity. If our Bruck-Reilly extension uses the trivial homomorphism, we can also show that the base semigroup is graph automatic.

Proposition 4.4.2. *Let T be a monoid and $\theta : T \to \{1_T\}$ be the trivial homomorphism. Let $S = \mathrm{BR}(T, \theta)$ be graph automatic. Then T is also graph automatic.*

Proof. Let (X, Σ, R, ν) be a graph automatic structure for S with uniqueness. Note that as θ is the trivial homomorphism we have that $tc = c$ and $bt = b$ for any $t \in T$. Let $\gamma \in R$ be the unique word such that $\nu(\gamma) = c$. Consider

$$\begin{aligned} L &= (R \times \{\gamma\}) \cap R_c \\ &= \{(\alpha, \gamma) : \alpha \in R \text{ and } \nu(\alpha)c = c\}. \end{aligned}$$

Now for any $s \in S$ we have that $s = c^i t b^j$ for some $t \in T$ and $i, j \in \mathbb{N}_0$, and so $sc = (c^i t b^j)c = c$ if and only if $i = j = 0$ or $s = cb$. Let $\delta \in R$ be the unique word such that $\nu(\delta) = cb$. Then

$$L = (\nu^{-1}(T) \cup \{\delta\}) \times \{\gamma\}$$

and so

$$\nu^{-1}(T) = L^{(1)} \setminus \{\delta\},$$

where $L^{(1)}$ is the first component of L, is regular. Thus, by Theorem 3.1.2, T is graph automatic. $\square$

Thus Theorem 4.4.1 does not only hold when we have an automorphism, and so we ask whether this is the case for Bruck-Reilly extensions in general.

Question 4.4.3. If we have a graph automatic Bruck-Reilly extension with any homomorphism, is the base semigroup necessarily graph automatic?

We next consider the converse of this, namely whether the Bruck-Reilly extension of a graph automatic monoid M is graph automatic. We begin by considering the case where our homomorphism θ is automatic with respect to the graph automatic structure of M, that is M is graph automatic with structure (X, Σ, R, ν) and the set $\{(\alpha, \beta) \in R \times R : \theta(\nu(\alpha)) = \nu(\beta)\}$ is regular.

Note that if θ is regular then θ^n is regular for any $n \in \mathbb{N}$. However, in order to use this to show that our Bruck-Reilly extension is automatic, we would need to take the union of infinitely many regular languages, one for each θ^n. As we cannot ensure that this union is regular, we instead consider the case where the powers of θ eventually stabilise, that is there exists some constant $k \in \mathbb{N}$

such that $\theta^n = \theta^{n+1}$ for all $n \geq k$. In this case we can now show that our Bruck-Reilly extension is graph automatic.

Theorem 4.4.4. *Let T be a graph automatic monoid, $\theta : T \to T$ be a homomorphism which is automatic with respect to the graph automatic structure of T, and $m \in \mathbb{N}$ be a constant such that $\theta^n = \theta^{n+1}$ for all $n \geq m$. Then* $\mathrm{BR}(T, \theta)$ *is graph automatic.*

Proof. Let T be graph automatic with structure with uniqueness (X, Σ, R, ν). Let $X' = X \cup \{b, c\}$, let $\Sigma' = \Sigma \cup \{\beta, \gamma\}$, and let

$$\begin{aligned} L &= \{\gamma^i \beta^j \alpha : i, j \in \mathbb{N}_0, \alpha \in R\} \\ &= \gamma^* \beta^* R. \end{aligned}$$

Then $\mu : L \to \mathrm{BR}(T, \theta)$, defined by

$$\mu(\gamma^i \beta^j \alpha) = (i, \nu(\alpha), j),$$

is a surjection.

Now

$$\begin{aligned} L_= &= \{(\gamma^i \beta^j \alpha_1, \gamma^k \beta^l \alpha_2) \in L \times L : (i, \nu(\alpha_1), j) = (k, \nu(\alpha_2), l)\} \\ &= \{(\alpha, \alpha) : \alpha \in L\}, \end{aligned}$$

as $\mu(\gamma^i \beta^j \alpha_1) = \mu(\gamma^k \beta^l \alpha_2)$ gives us $i = j, k = l$ and $\nu(\alpha_1) = \nu(\alpha_2)$, so $\alpha_1 = \alpha_2$ by the uniqueness of our original structure.

Then for $x \in X$ we have that $(0, x, 0)$ is a generator of T and

$$\begin{aligned} L_{(0,x,0)} =& \{(\gamma^i \beta^j \alpha_1, \gamma^k \beta^l \alpha_2) \in L \times L : (i, \nu(\alpha_1), j)(0, x, 0) = (k, \nu(\alpha_2), l)\} \\ =& \{(\gamma^i \beta^j \alpha_1, \gamma^k \beta^l \alpha_2) \in L \times L : (i, \nu(\alpha_1)\theta^j(x), j) = (k, \nu(\alpha_2), l)\} \\ =& \{(\gamma^i \beta^j \alpha_1, \gamma^k \beta^l \alpha_2) \in L \times L : (\alpha_1, \alpha_2) \in R_{\theta^j(x)}\} \end{aligned}$$

$$=(\gamma,\gamma)^*(\beta,\beta)^m(\beta,\beta)^* R_{\theta^m(x)} \cup \bigcup_{j<m} (\gamma,\gamma)^*(\beta,\beta)^j R_{\theta^j(x)}.$$

Hence $L_{(0,x,0)}$ is regular.

Next

$$\begin{aligned} L_b &= L_{(0,1,1)} \\ &= \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : (i, \nu(\alpha_1), j)(0,1,1) = (k, \nu(\alpha_2), l)\} \\ &= \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : (i, \nu(\alpha_1), j+1) = (k, \nu(\alpha_2), l)\} \\ &= \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : k = i, l = j+1 \text{ and } (\alpha_1, \alpha_2) \in R_=\} \\ &= (\gamma,\gamma)^*(\beta,\beta)^*(\$,\beta)\{(\alpha,\alpha) : \alpha \in R\} \end{aligned}$$

is regular, and

$$\begin{aligned} L_c =& L_{(1,1,0)} \\ =& \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : (i, \nu(\alpha_1), j)(1,1,0) = (k, \nu(\alpha_2), l)\} \\ =& \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : j \geq 1, (i, \nu(\alpha_1), j-1) = (k, \nu(\alpha_2), l)\} \\ & \cup \{(\gamma^i\alpha_1, \gamma^k\alpha_2) \in L \times L : (i-1, \theta(\nu(\alpha_1)), 0) = (k, \nu(\alpha_2), l)\} \\ =& \{(\gamma^i\beta^j\alpha_1, \gamma^k\beta^l\alpha_2) \in L \times L : k = i, l = j-1 \text{ and } (\alpha_1, \alpha_2) \in R_=\} \\ & \cup \{(\gamma^i\alpha_1, \gamma^k\alpha_2) \in L \times L : k = i+1 \text{ and } \theta(\nu(\alpha_1)) = \nu(\alpha_2)\} \\ =& (\gamma,\gamma)^*(\beta,\beta)^*(\beta,\$)\{(\alpha,\alpha) : \alpha \in R\} \\ & \cup (\gamma,\gamma)^*(\$,\gamma)\{(\alpha_1,\alpha_2) \in R \times R : \theta(\nu(\alpha_1)) = \nu(\alpha_2)\} \end{aligned}$$

is regular, as the set $\{(\alpha_1, \alpha_2) \in R \times R : \theta(\nu(\alpha_1)) = \nu(\alpha_2)\}$ is regular by the automaticity of θ. Thus $\mathrm{BR}(T, \theta)$ is graph automatic with structure (X', Σ', L, μ).

□

In particular, if θ is the identity homomorphism then we may use this to show that our extension is graph automatic.

Proposition 4.4.5. *Let T be a graph automatic monoid and $\theta : T \to T$ be the* *identity homomorphism. Then* $S = \mathrm{BR}(T, \theta)$ *is graph automatic.*

Proof. Let T be a graph automatic monoid with graph automatic structure with uniqueness (X, Σ, R, ν). Let $\theta : T \to T$ be defined by $\theta(t) = t$ for all $t \in T$. Then the language recognising θ is

$$\begin{aligned} L_\theta =& \{(\alpha, \beta) \in R \times R : \theta(\nu(\alpha)) = \nu(\beta)\} \\ =& \{(\alpha, \alpha) : \alpha \in R\}, \end{aligned}$$

which is regular, thus θ is a regular homomorphism. We also have that $\theta^n = \theta$ for any $n \in \mathbb{N}$ and so by Theorem 4.4.4 we have that $\mathrm{BR}(T, \theta)$ is graph automatic. □

In a similar way, if θ is the trivial homomorphism we can also show that a Bruck-Reilly extension is graph automatic.

Proposition 4.4.6. *Let* T *be a graph automatic monoid and* $\theta : T \to T$ *be defined by* $\theta(t) = 1$ *for all* $t \in T$*. Then* $S = \mathrm{BR}(T, \theta)$ *is graph automatic.*

Proof. Let T be a graph automatic monoid with graph automatic structure with uniqueness (X, Σ, R, ν). Let $\theta : T \to T$ be defined by $\theta(t) = 1$ for all $t \in T$. Then the language recognising θ is

$$L_\theta = R \times \{\eta\},$$

where η is the unique element of R such that $\nu(\eta) = 1$. Then L_θ is regular, and $\theta^n = \theta$ for any $n \in \mathbb{N}$, so by Theorem 4.4.4 we have that $\mathrm{BR}(T, \theta)$ is graph automatic. □

It is natural to ask whether this type of result holds in general. Note that Bruck-Reilly extensions of automatic monoids are not necessarily automatic; an example of such an extension is given in [8]. In [10] an example of a Bruck-Reilly extension of an FA-presentable semigroup which is not FA-presentable is given.

Therefore, it seems likely that Bruck-Reilly extensions of graph automatic semigroups are not always graph automatic, and that it is necessary to have a regularity condition on the homomorphism. We ask whether it is necessary that our homomorphism is regular. However, it is not clear how we may isolate a language representing the homomorphism in order to show that it is regular.

Question 4.4.7. Are Bruck-Reilly extensions of graph automatic monoids always graph automatic?

Chapter 5

Constructions for Graph Automatic Semigroups: Unions

In this chapter we consider some further semigroup constructions and whether they preserve graph automaticity. In particular, we examine those constructions which can be considered as taking the union of semigroups. We look at zero unions, ordinal sums, Rees matrix constructions, and semilattice constructions.

5.1 Zero Unions

In this section we will consider the zero union of graph automatic semigroups.

Let S and T be semigroups and let 0 be an element not in S or T. The *zero union* of S and T, denoted $S \cup_0 T$, is the disjoint union $S \cup T \cup \{0\}$ with

$$x \cdot y = \begin{cases} xy \text{ as in } S, & \text{if } x, y \in S \\ xy \text{ as in } T, & \text{if } x, y \in T \\ 0, & \text{otherwise .} \end{cases}$$

We will now see that zero unions preserve graph automaticity.

Proposition 5.1.1. *The zero union $S \cup_0 T$ of semigroups S and T is graph automatic if and only if S and T are both graph automatic.*

Proof. Let S and T be graph automatic semigroups with structures (X, Σ_1, K, ν) and (Y, Σ_2, L, μ) respectively. Without loss of generality assume that $K \cap L = \emptyset$. Let $Z = X \cup Y \cup \{0\}$ be a generating set for $S \cup_0 T$, where 0 is an element disjoint from both S and T. Let $\Sigma = \Sigma_1 \cup \Sigma_2 \cup \{\zeta\}$ and $R = K \cup L \cup \{\zeta\}$, where ζ is some symbol not contained in either of K or L. Define $\lambda : R \to S \cup_0 T$ by

$$\lambda(\alpha) = \begin{cases} \nu(\alpha), & \text{if } \alpha \in K \\ \mu(\alpha), & \text{if } \alpha \in L \\ 0, & \text{if } \alpha = \zeta \end{cases}$$

so R is regular and λ is surjective. Now we have that

$$R_= = K_= \cup L_= \cup \{(\zeta, \zeta)\},$$

and

$$R_0 = R \times \{\zeta\},$$

which are both regular. Then for $x \in X$ we have that

$$R_x = K_x \cup (L \times \{\zeta\})$$

and for $y \in Y$ we have that

$$R_y = L_y \cup (K \times \{\zeta\}).$$

Hence R_x and R_y are also regular. Thus $(X \cup Y \cup \{0\}, \Sigma, R, \lambda)$ is a graph automatic structure for $S \cup_0 T$.

Conversely, suppose that $S \cup_0 T$ is graph automatic with structure (X, Σ, R, ν)

with uniqueness. Let $\zeta \in R$ denote the unique element such that $\nu(\zeta) = 0$. Let $X_S \subseteq X$ be the subset of X which generates S, and let $x \in X_S$. Now

$$R_0 = R \times \{\zeta\}$$

is regular and so

$$\begin{aligned} L =& R_x \setminus R_0 \\ =& \{(\alpha, \beta) \in R \times R : \nu(\alpha)x = \nu(\beta) \text{ such that } \nu(\beta) \neq 0\} \end{aligned}$$

is regular. Then to have $\nu(\alpha)x \neq 0$ we must have $\nu(\alpha) \in S$. So

$$\begin{aligned} L^{(1)} &= \{\alpha \in R : (\alpha, \beta) \in L \text{ for some } \beta \in R\} \\ &= \nu^{-1}S \end{aligned}$$

is regular. Then as S is a regular subsemigroup of $S \cup_0 T$ it is graph automatic by Theorem 3.1.2. Similarly, T can be shown to be regular, hence graph automatic, in the same way. □

5.2 Ordinal Sums

We now consider ordinal sums of graph automatic semigroups.

Let S and T be semigroups. Then the *ordinal sum* of S and T with ordering $S > T$ is the disjoint union $S \cup T$ with multiplication

$$x \cdot y = \begin{cases} xy \text{ as in } S, & \text{if } x, y \in S \\ xy \text{ as in } T, & \text{if } x, y \in T \end{cases}$$

and if $x \in S$ and $y \in T$ then

$$x \cdot y = y \cdot x = y.$$

We now see that ordinal sums preserve graph automaticity.

Proposition 5.2.1. *The ordinal sum of two semigroups is graph automatic if and only if the two semigroups are graph automatic.*

Proof. Suppose that S and T are graph automatic semigroups with structures (X, Σ_1, K, ν) and (Y, Σ_2, L, μ) respectively. Let U be the ordinal sum of S and T with ordering $S > T$. Then U is finitely generated by $X \cup Y$. Let $\Sigma = \Sigma_1 \cup \Sigma_2$, and let $R = K \cup L$. Define $\psi : R \to U$ by

$$\psi(\alpha) = \begin{cases} \nu(\alpha), & \text{if } \alpha \in K \\ \mu(\alpha), & \text{if } \alpha \in L. \end{cases}$$

By the uniqueness of our original structures, $R_=$ is regular. Let $x \in X$. Then

$$\begin{aligned} R_x &= \{(\alpha, \beta) \in K \times K : \nu(\alpha)x = \nu(\beta)\} \cup \{(\alpha, \beta) \in L \times L : \mu(\alpha)x = \mu(\beta)\} \\ &= K_x \cup L_=. \end{aligned}$$

Let $y \in Y$. Then

$$\begin{aligned} R_y &= \{(\alpha, \beta) \in K \times L : \nu(\alpha)y = \mu(\beta)\} \cup \{(\alpha, \beta) \in L \times L : \mu(\alpha)y = \mu(\beta)\} \\ &= (K \times \mu^{-1}(y)) \cup L_y. \end{aligned}$$

Thus R_x and R_y are regular and so $(X \cup Y, \Sigma, R, \psi)$ is a graph automatic structure for U.

Conversely, suppose that U is graph automatic, with structure (X, Σ, R, ν). Then T is a finitely generated ideal of U, and so by Theorem 3.1.4 we have that $\nu^{-1}T$ is regular, and so T is graph automatic by Theorem 3.1.2. We then have that $\nu^{-1}S = R \setminus \psi^{-1}T$, and as this is regular we also have that S is graph automatic by Theorem 3.1.2. ☐

Note that this now provides an alternative way to show that adding or removing zeroes and identities preserves graph automaticity. We have that S^1

is the ordinal sum of $\{1\}$ and S, with ordering $\{1\} > S$, and so we may use Proposition 5.2.1 to immediately deduce that adding an identity preserves graph automaticity, as previously demonstrated in Proposition 3.2.2. Similarly, S^0 is the ordinal sum of $\{0\}$ and S with ordering $S > \{0\}$, and so can be shown to be graph automatic using Proposition 5.2.1, as an alternative method to Proposition 3.2.1.

5.3 Rees Matrix Semigroups

We now consider Rees matrix semigroups.

Recall that a *Rees matrix semigroup* $M[S; I, J; P]$ is the set $I \times S \times J$, where S is a semigroup, I and J are index sets, and $P = (p_{ji})_{j \in J, i \in I}$ is a matrix with entries from S. Multiplication is given by

$$(i, s, j)(k, t, l) = (i, sp_{jk}t, l).$$

In [13] it is shown that if S is a group and I and J are finite sets, then $M[S; I, J; P]$ is automatic. This is extended to the case where S is a semigroup in [16], provided $M[S; I, J; P]$ is finitely generated. We consider the graph automatic case, and show that if our Rees matrix semigroup is finitely generated then graph automaticity is preserved. In [4] the authors give conditions for a Rees matrix semigroup to be finitely generated, namely:

Proposition 5.3.1 (Main Theorem of [4]). *Let S be a semigroup, let I and J be index sets, let $P = (p_{ji})_{j \in J, i \in I}$ be a $J \times I$ matrix with entries from S, and let U be the ideal of S generated by the set $\{p_{ji} : j \in J, i \in I\}$ of all entries of P. Then the Rees matrix semigroup $M[S; I, J; P]$ is finitely generated if and only if the following three conditions are satisfied:*

- *both I and J are finite;*
- *S is finitely generated; and*

- *the set $S \setminus U$ is finite.*

However, we will only require the first condition in order to show that Rees matrix constructions preserve graph automaticity. Also note that the second condition is immediately satisfied if S is graph automatic.

Theorem 5.3.2. *Let S be a graph automatic semigroup. Then any finitely generated Rees matrix semigroup $M[S; I, J; P]$ is graph automatic.*

Proof. Let $T = M[S; I, J; P]$ be a finitely generated Rees matrix semigroup with finite generating set Y. By Proposition 5.3.1, I and J must be finite for T to be finitely generated. As S is graph automatic it has a graph automatic structure with uniqueness, (X, Σ, R, ν). As a set, we have that $T = I \times S \times J$, and so for each $i \in I$ and $j \in J$ we introduce a new alphabet Σ_{ij} such that there is a bijection $\psi_{ij} : \Sigma \to \Sigma_{ij}$ defined by $\psi_{ij}(a) = a_{ij}$ for all letters $a \in \Sigma$. Then we may extend each of the ψ_{ij} to an isomorphism $\bar{\psi}_{ij} : \Sigma^* \to \Sigma_{ij}^*$. Then $\bar{\psi}_{ij}(R) = R_{ij}$ is a regular language isomorphic to R.

Let $\Pi = \bigcup_{i \in I, j \in J} \Sigma_{ij}$ and $L = \bigcup_{i \in I, j \in J} R_{ij}$, which is regular as I and J are finite. Define $\mu : L \to T$ by

$$\mu(\alpha_{ij}) = (i, \nu(\alpha), j)$$

for $\alpha_{ij} \in R_{ij}$. Thus if an element $s \in S$ is represented by a word $\alpha \in R$ then the element (i, s, j) is represented by $\bar{\psi}_{ij}(\alpha) = \alpha_{ij}$.

Now

$$L_{=} = \{(\alpha, \alpha) : \alpha \in L\},$$

as $\mu(\alpha_{ij}) = \mu(\beta_{mn})$ gives $(i, \nu(\alpha), j) = (m, \nu(\beta), n)$ so we must have $i = m, j = n$ and $\alpha = \beta$ by the uniqueness of the graph automatic structure for S.

Now let $y_{kl} = (k, y, l) \in Y$ be a generator of T. Then if $(\alpha_{ij}, \beta_{mn}) \in L_{y_{kl}}$ we have that $(i, \nu(\alpha)p_{jk}y, l) = (m, \nu(\beta), n)$ so we must have $m = i, n = l$ and

$(\alpha, \beta) \in R_{p_{jk}y}$. Let $\varphi_{ij,il} : \Sigma^* \times \Sigma^* \to \Sigma_{ij} \times \Sigma_{il}$ be a homomorphism defined by

$$\varphi_{ij,il}(\alpha, \beta) = (\alpha_{ij}, \beta_{il}).$$

Note that the image of $R \times R$ under $\varphi_{ij,il}$ will be $R_{ij} \times R_{il}$. Then we have that

$$\varphi_{ij,il}(R_x) = \{(\alpha_{ij}, \beta_{il}) \in R_{ij} \times R_{il} : (\alpha, \beta) \in R_x\}$$

and so

$$\begin{aligned} L_{y_{kl}} =& \{(\alpha, \beta) \in L \times L : (i, \nu(\alpha)p_{jk}y, l) = (i, \nu(\beta), l)\} \\ =& \bigcup_{i \in I, j \in J} \{(\alpha_{ij}, \beta_{il}) \in L \times L : (\alpha, \beta) \in R_{p_{jk}y}\} \\ =& \bigcup_{i \in I, j \in J} \varphi_{ij,il}(R_{p_{jk}y}) \end{aligned}$$

is a regular language, as k and y are fixed and for each $j \in J$ we have that $R_{p_{jk}y}$ is regular. Hence T is graph automatic, with structure (Y, Π, L, μ). □

We now ask whether the converse holds.

Question 5.3.3. If a Rees matrix semigroup is graph automatic, is the base semigroup necessarily graph automatic also?

Note that if we consider a completely simple semigroup S, that is a Rees matrix semigroup $S = M[G; I, J; P]$ over a group G, then it was shown in [13] that S is automatic if and only if G is automatic. Similarly, it was shown in [10] that the corresponding result holds for FA-presentable semigroups.

In order to show that this also holds for graph automatic semigroups, we would like to isolate a copy of G, for example the set $\{1\} \times G \times \{1\}$. It is easy to isolate $I \times G \times \{1\}$, as this is a left ideal, and so is a regular subsemigroup. The difficulty arises when we wish to isolate $\{1\} \times G \times J$, as we do not know how to show that this is regular. It is only in the case where $I = \{1\}$ that we know this must be regular and so can show that G is graph automatic.

Proposition 5.3.4. *Let $S = M[G; \{1\}, J; P]$ be a graph automatic Rees matrix semigroup over a group G. Then G is also graph automatic.*

Proof. Let $S = M[G; \{1\}, J; P]$ be graph automatic with structure (X, Σ, R, ν). Note that G must be finitely generated by Proposition 5.3.1. Consider the set $\{1\} \times G \times \{1\}$. This is isomorphic to the group G, and so if we can show that it is a regular subsemigroup of S then we must have that G is graph automatic.

Let $(1, g, 1) \in \{1\} \times G \times \{1\}$ and $(1, h, j) \in S$. Then

$$(1, h, j)(1, g, 1) = (1, hp_{j1}g, 1)$$

and so the set $\{1\} \times G \times \{1\}$ is a left ideal in S. Thus by Theorem 3.1.4, we have that $\{1\} \times G \times \{1\}$ is a regular subsemigroup of S, and thus G is graph automatic. □

Note that this result does not necessarily follow for semigroups, as if we have a Rees matrix semigroup $M[S; I, J; P]$ then $\{1\} \times S \times \{1\}$ is not necessarily isomorphic to S when S is not a group.

We next consider the *Rees matrix semigroup with zero*, $M^0[S; I, J; P]$, which is the set $(I \times S \times J) \cup \{0\}$ for some element $0 \notin S$, where S is a semigroup, I and J are sets, and $P = (p_{ji})_{j \in J, i \in I}$ is a matrix with entries from $S \cup \{0\}$. Multiplication is given by

$$(i, s, j)(k, t, l) = \begin{cases} (i, sp_{jk}t, l), & \text{if } p_{jk} \neq 0, \\ 0 & \text{if } p_{jk} = 0 \end{cases}$$

and

$$0(i, s, j) = (i, s, j)0 = 0^2 = 0.$$

Note that Proposition 5.3.1 still holds for Rees matrix semigroups with zero, as shown in [4].

We now see that if the base semigroup is graph automatic, then any finitely generated Rees matrix semigroup with zero is also graph automatic.

Theorem 5.3.5. *Let S be a graph automatic semigroup. Then any finitely generated Rees matrix semigroup with zero $M^0[S; I, J; P]$ is graph automatic.*

Proof. Let $T = M^0[S; I, J; P]$ be a finitely generated Rees matrix semigroup with zero, generated by the finite set Y. By Proposition 5.3.1, I and J must be finite for T to be finitely generated. Let S be graph automatic with structure with uniqueness, (X, Σ, R, ν).

As in Theorem 5.3.2, we introduce a new alphabet Σ_{ij} for each $i \in I$ and $j \in J$, and define maps $\psi_{ij} : \Sigma \to \Sigma_{ij}$ defined by $\psi_{ij}(a) = a_{ij}$ for all letters $a \in \Sigma$. We then extend each map to an isomorphism $\bar{\psi}_{ij} : \Sigma^* \to \Sigma_{ij}^*$, so we have that $\bar{\psi}_{ij}(R) = R_{ij}$ is a regular language isomorphic to R. Let $\Pi = \bigcup_{i \in I, j \in J} \Sigma_{ij} \cup \{\zeta\}$ and $L = \bigcup_{i \in I, j \in J} R_{ij} \cup \{\zeta\}$. Then L is regular as I and J are finite. Define $\mu : L \to T$ by

$$\mu(\alpha) = \begin{cases} (i, \nu(\alpha), j), & \alpha = \alpha_{ij} \in R_{ij} \\ 0, & \alpha = \zeta. \end{cases}$$

Now

$$L_= = \{(\alpha, \alpha) : \alpha \in L\}$$

as in Theorem 5.3.2, and

$$L_0 = L \times \{\zeta\},$$

which are both regular.

Let $y_{kl} = (k, y, l) \in Y \setminus \{0\}$ be a generator of T and let $\varphi_{ij,il} : \Sigma^* \times \Sigma^* \to \Sigma_{ij} \times \Sigma_{il}$ be the homomorphism defined by

$$\varphi_{ij,il}(\alpha, \beta) = (\alpha_{ij}, \beta_{il}),$$

as in Theorem 5.3.2. We have that

$$L_{y_{kl}} = \bigcup_{i \in I, j \in J} K_{ij} \cup \{(\zeta, \zeta)\}$$

where

$$K_{ij} = \begin{cases} \varphi_{ij,jl}(R_{p_{jk}y}), & p_{jk} \neq 0 \\ R_{ij} \times \{\zeta\}, & p_{jk} = 0. \end{cases}$$

As $\varphi_{ij,jl}(R_{p_{jk}y})$ is regular for each choice of i and j, we have a finite union of regular languages, and so $L_{y_{kl}}$ is regular. Hence T is graph automatic with structure (Y, Π, L, μ). □

5.4 Semilattices of Semigroups

We now consider semilattices of semigroups. In [10], the authors classify finitely generated FA-presentable Clifford semigroups, that is semigroups which are strong semilattices of groups, and also show that FA-presentable semigroups are not closed under strong semilattices in general. We shall consider semilattices and strong semilattices of semigroups, rather than just those of groups.

Recall that a *semilattice* is a commutative semigroup of idempotents. A semigroup S is a *semilattice of semigroups* if S can be decomposed into a disjoint union of semigroups $\bigcup_{u \in Y} S_u$ for a semilattice Y, such that if $s \in S_u$ and $t \in S_v$ we have $st \in S_{uv}$.

We see that if a semilattice of semigroups is graph automatic then each constituent semigroup is also graph automatic.

Theorem 5.4.1. *Let $S = \bigcup_{u \in Y} S_u$ be a semilattice of semigroups over a finite semilattice Y. If S is graph automatic and S_u is finitely generated for each $u \in Y$ then each S_u is graph automatic.*

Proof. Let S have graph automatic structure (X, Σ, R, ν). For each $u \in Y$ let S_u be generated by X_u. Let 0 be the bottom element of the semilattice. Note that there must be such an element as Y is finite. Then S_0 is a finitely generated ideal of S and so is graph automatic by Corollary 3.1.5. Let $u \in Y \setminus \{0\}$. Let I be the ideal of S generated by S_u. This ideal is generated by $\bigcup_{v \leq u} X_v$. By Theorem 3.1.4, we have that $\nu^{-1}I$ is a regular language. Let J be the ideal generated by $\bigcup_{v<u} X_v$. Then $\nu^{-1}J$ is also regular. Thus $\nu^{-1}S_u = \nu^{-1}I \setminus \nu^{-1}J$ is regular, and so by Theorem 3.1.2 we have that S_u is graph automatic. □

We next examine the case where we have a strong semilattice of semigroups.

If a semigroup S can be decomposed into a semilattice of semigroups $S = \bigcup_{u \in Y} S_u$ for a semilattice Y, and in addition we have homomorphisms $\varphi_{u,v} : S_u \to S_v$ for $u \geq v$ satisfying

$$\varphi_{u,u} = \mathrm{id}_{S_u}$$

and

$$\varphi_{v,w} \circ \varphi_{u,v} = \varphi_{u,w}$$

for $u \geq v \geq w$, such that for $s \in S_u$ and $t \in S_v$ our multiplication is given by

$$st = \varphi_{u,uv}(s)\varphi_{v,uv}(t),$$

we have a *strong semilattice of semigroups.*

In [3] the authors give the following condition for finite generation of strong semilattices of semigroups.

Proposition 5.4.2 (Theorem 6.1 of [3])**.** *A strong semilattice of semigroups $S = \bigcup_{u \in Y} S_u$ is finitely generated if and only if Y is finite and every semigroup S_u for $u \in Y$ is finitely generated.*

Thus, as graph automaticity implies finite generation, we have the following corollary to Theorem 5.4.1.

Corollary 5.4.3. *Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups. If S is graph automatic then each S_u is graph automatic.*

We shall now consider the converse, that is given a semigroup which can be expressed as a semilattice of graph automatic semigroups, is the semigroup itself necessarily graph automatic? In particular we consider the case where we have a strong semilattice of semigroups.

In [13] the authors consider Clifford semigroups, that is semigroups which are strong semilattices of groups. An example is given of a semilattice of two automatic groups which is not automatic. We consider if this is also the case for graph automatic semigroups.

We begin by looking at the case where all our homomorphisms $\varphi_{u,v}$ are automatic with respect to the graph automatic structures of the corresponding semigroups S_u and S_v. That is, if S_u and S_v are graph automatic with structures $(X_u, \Sigma_u, R_u, \nu_u)$ and $(X_v, \Sigma_v, R_v, \nu_v)$ respectively, and $\varphi_{u,v} : S_u \to S_v$, we have that

$$\{(\alpha, \beta) \in R_u \times R_v : \varphi_{u,v}(\nu_u(\alpha)) = \nu_v(\beta)\}$$

is regular.

Proposition 5.4.4. *Let Y be a finite semilattice and let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice over Y where each S_u is graph automatic. If each homomorphism $\varphi_{u,v} : S_u \to S_v$ is automatic with respect to the graph automatic structures of S_u and S_v, then S is graph automatic.*

Proof. Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups over the finite semilattice Y, such that each of the S_u is graph automatic. Thus each S_u has graph automatic structure $(X_u, \Sigma_u, R_u, \nu_u)$. Let $X = \bigcup_{u \in Y} X_u$. Then as S is the union of the S_u we have that X is a generating set for S, which is finite as each of the X_u are finite. Let $\Sigma = \bigcup_{u \in Y} \Sigma_u$ and $R = \bigcup_{u \in Y} R_u$. As each R_u is regular then R is also regular. Define $\nu : R \to S$ by

$$\nu(\alpha) = \nu_u(\alpha)$$

for $\alpha \in R_u$. We have

$$R_= = \bigcup_{u \in Y} (R_u)_=$$

which is regular.

Now let $x_v \in X_v$. Note that if $s_u x_v = t$ in S for some $s_u \in S_u$ then $t \in S_{uv}$.

Thus

$$\begin{aligned}R_{x_v} &= \bigcup_{u\in Y}\{(\alpha_u,\beta_{uv})\in R_u\times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha_u))\varphi_{v,uv}(x_v)=\nu_{uv}(\beta_{uv})\}\\
&= \bigcup_{u\in Y}\{(\alpha_u,\beta_{uv})\in R_u\times R_{uv} : (\nu_{uv}^{-1}(\varphi_{u,uv}(\nu_u(\alpha_u))),\beta_{uv})\in (R_{uv})_{\varphi_{v,uv}(x_v)}\}\\
&= \bigcup_{u\in Y}\{(\alpha_u,\beta_{uv})\in R_u\times R_{uv} : \text{ there exists } \gamma_{uv}\in R_{uv} \text{ such that}\\
&\qquad (\alpha_u,\gamma_{uv})\in L_{\varphi_{u,uv}} \text{ and } (\gamma_{uv},\beta_{uv})\in (R_{uv})_{\varphi_{v,uv}(x_v)}\},\end{aligned}$$

where

$$L_{\varphi_{u,uv}} = \{(\alpha,\beta)\in R_u\times R_{uv} : \varphi_{u,uv}(\nu(\alpha))=\nu(\beta)\}$$

is the language recognising $\varphi_{u,uv}$. As all our homomorphisms are regular we have that $L_{\varphi_{u,uv}}$ is regular, and as S_{uv} is graph automatic we have that $(R_{uv})_{\varphi_{v,uv}(x_v)}$ is regular. Thus R_{x_v} is regular, and so S is graph automatic with structure (X,Σ,R,ν). □

We now use this to give examples of situations where we have a semilattice of semigroups which is graph automatic, beginning with the case where our strong semilattice consists of isomorphic semigroups and all the homomorphisms are the identity.

Proposition 5.4.5. *Let Y be a finite semilattice and let $S = \bigcup_{u\in Y} S_u$ be a strong semilattice over Y where all the S_u are copies of a graph automatic semigroup T. Let each homomorphism $\varphi_{u,v} : S_u \to S_v$ be the identity homomorphism. Then S is graph automatic.*

Proof. Let T be a graph automatic semigroup with structure with uniqueness (X,Σ,R,ν). For each $u\in Y$ we take a copy of T indexed by u, namely $S_u = \{s_u : s\in T\}$. Let $S=\bigcup_{u\in Y} S_u$ be a strong semilattice of semigroups. As T is graph automatic, each S_u is graph automatic with structure (X_u,Σ_u,R_u,ν_u).

Consider $\varphi_{u,v} : S_u \to S_v$ defined by $\varphi_{u,v}(s_u) = s_v$. Then

$$\begin{aligned} L_{\varphi_{u,v}} =& \{(\alpha, \beta) \in R_u \times R_v : \varphi_{u,v}(\nu_u(\alpha)) = \nu_v(\beta)\} \\ =& \{(\alpha_u, \alpha_v) : \alpha \in R\} \end{aligned}$$

which is regular as we can easily construct an automaton which recognises if we have two copies of the same word but with different indices. Thus by Proposition 5.4.4 we have that S is graph automatic. ☐

We may also show that a strong semilattice of isomorphic semigroups with an idempotent is graph automatic, if all the homomorphisms are trivial.

Proposition 5.4.6. *Let Y be a finite semilattice and let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice over Y where all the S_u are copies of a graph automatic semigroup T containing a distinguished idempotent e. Let each homomorphism $\varphi_{u,v} : S_u \to S_v$ be defined by $\varphi_{u,v}(s_u) = e_v$ for each $s_u \in S_u$. Then S is graph automatic.*

Proof. Let T be a graph automatic semigroup with structure (X, Σ, R, ν) with uniqueness. For each $u \in Y$ we take a copy of T indexed by u, namely $S_u = \{s_u : s \in T\}$. Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups. As T is graph automatic, each S_u is graph automatic with structure $(X_u, \Sigma_u, R_u, \nu_u)$. Consider $\varphi_{u,v} : S_u \to S_v$ defined by $\varphi_{u,v}(s_u) = e_v$. Then

$$\begin{aligned} L_{\varphi_{u,v}} =& \{(\alpha, \beta) \in R_u \times R_v : \varphi_{u,v}(\nu_u(\alpha)) = \nu_v(\beta)\} \\ =& R_u \times \{\eta_v\}, \end{aligned}$$

where η_v is the unique word in R_v representing e_v. This is a regular language, and so by Proposition 5.4.4 S is graph automatic. ☐

We now consider whether the homomorphisms associated to a graph automatic strong semilattice of semigroups must be automatic. We begin by considering the case where all our constituent semigroups are monoids.

Proposition 5.4.7. *Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of monoids over a finite semilattice Y. If S is graph automatic then each homomorphism $\varphi_{u,v} : S_u \to S_v$ is automatic with respect to the graph automatic structures of S_u and S_v.*

Proof. Let $S = \bigcup_{u \in Y} S_u$ be graph automatic with structure (X, Σ, R, ν). By Corollary 5.4.3, each S_u is graph automatic with structure $(X_u, \Sigma, R_u, \nu_u)$, where $R_u = \nu^{-1} S_u$ and ν_u is the restriction of ν to S_u.

Consider the homomorphism $\varphi_{u,uv}$. Note that this is a monoid homomorphism, and so we must have that $\varphi_{u,uv}(1_u) = 1_{uv}$. Let

$$L_{\varphi_{u,uv}} = \{(\alpha, \beta) \in R_u \times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha)) = \nu_{uv}(\beta)\},$$

that is the language recognising $\varphi_{u,uv}$. We want to show that this is regular. As S is graph automatic we have that

$$\begin{aligned} R_{1_v} &= \bigcup_{u \in Y} \{(\alpha, \beta) \in R_u \times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha)) \varphi_{v,uv}(1_v) = \nu_{uv}(\beta)\} \\ &= \bigcup_{u \in Y} \{(\alpha, \beta) \in R_u \times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha)) 1_{uv} = \nu_{uv}(\beta)\} \\ &= \bigcup_{u \in Y} \{(\alpha, \beta) \in R_u \times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha)) = \nu_{uv}(\beta)\} \end{aligned}$$

is regular. Then as S_u is graph automatic we have that R_u is regular for each $u \in Y$, and so

$$\begin{aligned} R_{1_v} \cap (R_u \times R) &= \{(\alpha, \beta) \in R_u \times R_{uv} : \varphi_{u,uv}(\nu_u(\alpha)) = \nu_{uv}(\beta)\} \\ &= L_{\varphi_{u,uv}} \end{aligned}$$

is regular. ☐

We now use this to show that homomorphisms for any graph automatic strong semilattice of semigroups must be automatic.

Proposition 5.4.8. *Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups over a finite semilattice Y. If S is graph automatic then each homomorphism $\varphi_{u,v} : S_u \to S_v$ is automatic with respect to the graph automatic structures of S_u and S_v.*

Proof. Let $S = \bigcup_{u \in Y} S_u$ be graph automatic with structure (X, Σ, R, ν). We consider each of the semigroups S_u for $u \in Y$ and adjoin an identity to each in order to get S_u^1. We also extend each of the homomorphisms associated with our semilattice, with $\varphi_{u,v} : S_u \to S_v$ being extended to a homomorphism $\bar{\varphi}_{u,v} : S_u^1 \to S_v^1$, given by

$$\bar{\varphi}_{u,v}(s) = \begin{cases} \varphi_{u,v}(s), & s \in S \\ 1_v, & s = 1_u. \end{cases}$$

We now form a new semigroup $\bar{S} = \bigcup_{u \in Y} S_u^1$, which has multiplication defined by

$$st = \bar{\varphi}_{u,uv}(s)\bar{\varphi}_{v,uv}(t)$$

for $s \in S_u^1$ and $t \in S_v^1$. Thus we have formed a new semigroup which is a strong semilattice of monoids.

Now S is an ideal of $\bar{S}$, and S has finite Rees index in $\bar{S}$. Thus, by Proposition 3.3.2 we have that $\bar{S}$ is also graph automatic with structure $(X \cup \{1_u : u \in Y\}, \Sigma \cup C, \bar{R} = R \cup C, \bar{\nu})$, where $C = \{c_u : u \in Y\}$ is a set in one-to-one correspondence with the set $\{1_u : u \in Y\}$ and

$$\bar{\nu}(\alpha) = \begin{cases} \nu(\alpha), & \alpha \in R \\ 1_u, & \alpha = c_u. \end{cases}$$

Then by Proposition 5.4.7 each homomorphism $\bar{\varphi}_{u,v}$ is automatic with respect to the graph automatic structures of S_u^1 and S_v^1 and so

$$\begin{aligned} L_{\bar{\varphi}_{u,v}} &= \{(\alpha, \beta) \in \bar{R}_u \times \bar{R}_v : \bar{\varphi}_{u,v}(\bar{\nu}_u(\alpha)) = \bar{\nu}_v(\beta)\}\} \\ &= \{(\alpha, \beta) \in R_u \times R_v : \varphi_{u,v}(\nu_u(\alpha)) = \nu_v(\beta)\} \cup \{(c_u, c_v)\} \end{aligned}$$

is regular, and so

$$L_{\varphi_{u,v}} = L_{\bar{\varphi}_{u,v}} \setminus \{(c_u, c_v)\}$$

is regular. □

Combining these results gives the following theorem.

Theorem 5.4.9. *A strong semilattice of semigroups $S = \bigcup_{u \in Y} S_u$ is graph automatic if and only if each constituent semigroup is graph automatic and each homomorphism $\varphi_{u,v} : S_u \to S_v$ is automatic with respect to the graph automatic structures of S_u and S_v.*

Proof. This follows from Corollary 5.4.3, Proposition 5.4.8, and Proposition 5.4.4. □

Chapter 6

Unary Graph Automatic Semigroups

In this chapter we consider a special case of graph automatic semigroups, namely those whose alphabets contain a single letter. We call such semigroups *unary graph automatic*. We will examine the structure of the automata which recognise such semigroups, and then apply this to demonstrate some properties of unary graph automatic semigroups.

6.1 Definition and Examples

Definition 6.1.1. A semigroup S is *unary graph automatic* if it has a graph automatic structure $(X, \{a\}, R, \nu)$. When our alphabet consists of a single letter we will often write our structure as (X, a, R, ν).

Similarly to the general case, we may show that all finite semigroups have a unary graph automatic structure.

Proposition 6.1.2. *Any finite semigroup is unary graph automatic.*

Proof. Let $S = \{x_1, x_2, \dots, x_n\}$ be a finite semigroup. Let $R = \{a, a^2, \dots, a^n\}$.

Then we define $\nu : R \to S$ by $\nu(a^k) = x_k$. So

$$R_= = \{(a^i, a^i) : 1 \leq i \leq n\}$$

and

$$R_x = \{(a^i, a^j) \in R \times R : x_i x = x_j\}$$

for each $x \in S$ are all finite, thus regular. Hence (S, a, R, ν) is a unary graph automatic structure for S. □

Therefore, as in the general graph automatic case, we will primarily be interested in infinite semigroups.

It follows from the properties of general graph automatic semigroups that unary graph automatic semigroups have a structure with uniqueness. This means that for a unary graph automatic semigroup we may find a structure (X, a, R, ν) such that ν is a bijection $\nu : R \to S$. We use the following result from [11] to show that if we have an infinite semigroup we can always choose our language R to be the entirety of a^* and still maintain injectivity.

Proposition 6.1.3 ([11], Theorem 9)**.** *Let S be an infinite relational structure that admits a unary automatic presentation. Then S has an injective unary automatic presentation (a^*, ψ).*

Now graph automatic structures are special cases of FA-presentable structures, as shown in Subsection 2.2.2. Namely, a semigroup is graph automatic if and only if its Cayley graph is FA-presentable, and in particular a semigroup is unary graph automatic if and only if its Cayley graph has a unary automatic presentation. Hence we may apply Proposition 6.1.3 to unary graph automatic semigroups.

Corollary 6.1.4. *Let S be an infinite unary graph automatic semigroup. Then S has an injective unary graph automatic structure (X, a, a^*, ν).*

From this point on we may assume that all our unary graph automatic structures are structures with uniqueness, and that for any infinite unary graph automatic semigroups our structure is of the form (X, a, a^*, ν).

We now give some examples of unary graph automatic semigroups. We begin by showing that $\mathbb{N}$ has a unary graph automatic structure.

Example 6.1.5. Consider $(\mathbb{N}, +)$, generated by $\{1\}$. We show that this is unary graph automatic. Let $\nu : a^* \to \mathbb{N}$ be defined by $\nu(a^n) = n + 1$. Then

$$\begin{aligned}(a^*)_= =&\{(a^n, a^m) : \nu(a^n) = \nu(a^m)\}\\ =&(a, a)^*\end{aligned}$$

and

$$\begin{aligned}(a^*)_1 =&\{(a^n, a^m) : \nu(a^n) + 1 = \nu(a^m)\}\\ =&(a, a)^*(\$, a)\end{aligned}$$

are both regular, thus $(\{1\}, a, a^*, \nu)$ is a unary graph automatic structure for $\mathbb{N}$.

Next we see that $\mathbb{Z}$ is unary graph automatic.

Example 6.1.6. Consider $(\mathbb{Z}, +)$ with generating set $\{1, -1\}$. We show that $\mathbb{Z}$ is unary graph automatic.

Let $\nu : a^* \to \mathbb{Z}$ be defined by

$$\nu(a^n) = \begin{cases} \frac{n}{2}, & n \text{ even} \\ -\frac{n+1}{2}, & n \text{ odd}\ . \end{cases}$$

This is injective and so

$$(a^*)_= = (a, a)^*$$

is regular.

Then

$$(a^*)_1 = \{(a^n, a^m) : \nu(a^n) + 1 = \nu(a^m)\}$$

$$= (a^2, a^2)^*\{(\epsilon, a^2), (a^3, a)\} \cup \{(a, \epsilon)\}$$

and

$$\begin{aligned}(a^*)_{-1} &= \{(a^n, a^m) : \nu(a^n) - 1 = \nu(a^m)\} \\ &= (a^2, a^2)^*\{(a^2, \epsilon), (a, a^3)\} \cup \{(\epsilon, a)\}\end{aligned}$$

are regular, so $(\{1, -1\}, a, a^*, \nu)$ is a unary graph automatic structure for $\mathbb{Z}$.

Finally we show that the free product of two trivial semigroups is unary graph automatic.

Example 6.1.7. Let $S = \{s\}$ and $T = \{t\}$ be trivial semigroups. We show that $S * T$ is unary graph automatic. Let $\Sigma = \{a\}$ and $R = a^*$. Define a map $\nu \to S * T$ by

$$\nu(a^{4q+r}) \begin{cases} s(ts)^q, & r = 0 \\ s(ts)^q t, & r = 1 \\ t(st)^q, & r = 2 \\ t(st)^q s, & r = 3. \end{cases}$$

Note that this map is injective, so $R_=$ is regular. Then

$$\begin{aligned} R_s =& \{(a^{4q}, a^{4q}) : q \in \mathbb{N}\} \cup \{(a^{4q+1}, a^{4(q+1)}) : q \in \mathbb{N}\} \\ &\cup \{(a^{4q+2}, a^{4q+3}) : q \in \mathbb{N}\} \cup \{(a^{4q+3}, a^{4q+3}) : q \in \mathbb{N}\} \\ =& (a^4, a^4)^* \cup (a^4, a^4)^*\{(a, a^4), (a^2, a^3), (a^3, a^3)\} \end{aligned}$$

and

$$\begin{aligned} R_t =& \{(a^{4q}, a^{4q+1}) : q \in \mathbb{N}\} \cup \{(a^{4q+1}, a^{4q+1}) : q \in \mathbb{N}\} \\ &\cup \{(a^{4q+2}, a^{4q+2}) : q \in \mathbb{N}\} \cup \{(a^{4q+3}, a^{4(q+1)+2}) : q \in \mathbb{N}\} \\ =& (a^4, a^4)^*\{(\$, a), (a, a), (a^2, a^2), (a^3, a^6)\} \end{aligned}$$

so R_s and R_t are regular. Hence $(\{s,t\}, a, a^*, \nu)$ is a unary graph automatic structure for $S * T$.

We will see more examples of unary graph automatic semigroups in Sections 6.3 and 6.6.

6.2 Unary Automatic Semigroups and Unary FA-Presentable Semigroups

We now consider the relations between unary graph automatic semigroups and other unary structures. We begin by looking at unary automatic semigroups.

For an automatic semigroup to be represented by a single letter alphabet it must be generated by a single element. This means that the only unary automatic semigroups are monogenic semigroups, and in particular the only infinite unary automatic semigroup is the free monogenic semigroup. Now as any finite semigroup is unary graph automatic, we have examples of semigroups which are unary graph automatic but not unary automatic, namely any finite semigroup generated by more than one element.

Unary FA-presentable semigroups are discussed in [11]. In this paper the authors examine the properties of unary FA-presentable semigroups, showing that they are locally finite and must satisfy some Burnside identity. They go on to consider the Green's relations of unary FA-presentable semigroups, and examine which constructions preserve unary FA-presentability. In particular, we note that the authors also show that finitely generated semigroups are unary FA-presentable if and only if they are finite (Corollary 14, [11]). This means that the semigroups which are both unary FA-presentable and unary graph automatic are precisely finite semigroups. However, we have seen in Examples 6.1.5, 6.1.6 and 6.1.7 that there exist unary graph automatic semigroups which are not finite. Thus we have examples of graph automatic semigroups which are not unary FA-presentable.

Excluding $\mathbb{N}$, any infinite unary graph automatic semigroup will be neither unary automatic nor unary FA-presentable, thus Examples 6.1.6 and 6.1.7 give us semigroups which are unary graph automatic but neither automatic not FA-presentable.

6.3 Automata for Unary Graph Automatic Structures

We now consider the possible structures of the automata associated with infinite unary graph automatic semigroups. By Corollary 6.1.4, we may assume that we have an injective unary graph automatic structure (X, a, a^*, ν). As our structure is injective, the automaton which checks equality always accepts the language $(a, a)^*$ and so we need only examine the automata that accept the languages for multiplication by generators. We will refer to these as *acceptor automata*.

An acceptor automaton, $\mathcal{A}_x$, will accept words over the alphabet $\{(a, a), (a, \$), (\$, a)\}$. We take a deterministic automaton accepting our language, but for simplicity we ignore any sink states, meaning that our transition function is a partial function. So when we discuss the possible structures of our automata we refer only to those paths which can lead to an accept state. In practice this means that each state will have at most one transition labelled by each of (a, a), $(a, \$)$ and $(\$, a)$.

We now consider restrictions on the structure of our automata. Firstly, as we are accepting infinite languages our automata must contain at least one circuit or loop. Note that once we read $(a, \$)$ or $(\$, a)$ we can only read further letters of this form, so any circuit must be labelled with a single letter over our extended alphabet. We will refer to a circuit whose arrows are all labelled by (a, a) as an (a, a)-circuit, and similarly we have $(\$, a)$-circuits and $(a, \$)$-circuits. These circuits may have *offshoots*, by which we mean finite paths leading to accept states. If we have a $(\$, a)$-circuit or an $(a, \$)$-circuit we cannot have any offshoots on the circuit, due to the deterministic nature of our automata and the fact that once we read \$ in a component we must continue to do so. If we have an (a, a)-circuit, we may have offshoots labelled by either $(a, \$)$ or $(\$, a)$.

The following lemmas allow us to determine the possible structures for the acceptor automata for a unary graph automatic semigroup. We operate under the following assumptions:

- Our semigroup is infinite.
- We have a unary graph automatic structure (X, a, R, ν), where ν is injective and $R = a^*$.
- The automaton $\mathscr{A}_x$ recognises the language $(a^*)_x$.
- We form $\mathscr{A}_x$ by taking a deterministic automaton and removing any states and their associated transitions which can never lead to an accept state.

We are particularly interested in the forms and positions of circuits. We first see that we can immediately rule out one type of circuit.

Lemma 6.3.1. *An acceptor automaton for a unary graph automatic semigroup cannot have a* $(\$, a)$*-circuit.*

Proof. Suppose that $\mathscr{A}_x$ contains a $(\$, a)$-circuit of length p. There must be an accept state somewhere on the circuit. Let (a^i, a^j) be the first word accepted by a state on this circuit. Then for any $n \in \mathbb{N}$ we have that (a^i, a^{j+pn}) is also accepted by the automaton, as we may traverse the circuit multiple times and return to the accept state. Hence we have $\nu(a^j) = \nu(a^{j+np})$ for any $n \in \mathbb{N}_0$. This contradicts the injectivity of our structure. □

We next consider what happens if we have multiple circuits, and see that they cannot be placed successively, in the sense that a path to an accept state cannot go through one circuit and then into a second circuit.

Lemma 6.3.2. *An acceptor automaton for a unary graph automatic semigroup cannot have two successive circuits.*

Proof. Suppose that $\mathscr{A}_x$ has two successive circuits. As $\mathscr{A}_x$ is deterministic, these circuits must be of distinct types, else there would be a point where we leave the first circuit to enter the second where we have two edges leaving one state labelled with the same letter, which contradicts the determinism of our automaton. We cannot have a $(\$, a)$-circuit and we cannot read (a, a) after reading $(a, \$)$, thus the only way to have successive circuits is to have an (a, a)-circuit followed by an $(a, \$)$-circuit. We suppose that this is the case and let p be the length of the first circuit and q be the length of second circuit. We consider the form of words which are accepted by some accept state $\bar{s}$ on the $(a, \$)$-circuit. This state will accept words of the form $(a^{np+i}, a^{np+i})(a^{mq+j}, \$)$ for all $n, m \in \mathbb{N}$, where i is the length of the path from the start state to the point at which we leave the first circuit and j is the finite length of the path from the place where we leave the first circuit to $\bar{s}$ on the second circuit.

We now consider the effect of traversing the two circuits multiple times by varying the values of m and n, and show that it is possible for this automaton to accept two different words with the same first components. First let $n = 3q$ and $m = 2p$. Then the state $\bar{s}$ accepts the word $(a^{5pq+i+j}, a^{2pq+i})$. Now let $n = 2q$ and $m = 3p$. Then $\bar{s}$ also accepts the word $(a^{5pq+i+j}, a^{3pq+i})$. Now we have that $\nu(a^{2pq+i}) = \nu(a^{5pq+i+j})x = \nu(a^{3pq+i})$, a contradiction to the injectivity of our structure. Thus we cannot have two successive circuits. □

This still allows our automaton to contain multiple circuits, but only if they are not successive. We can have multiple circuits labelled $(a, \$)$ but the determinism of our automaton means that we can have at most one (a, a)-circuit, as if we had multiple (a, a)-circuits both would have to be preceded by a path labelled (a, a) and so at some point there would have to be a place where the initial path labelled (a, a) split into two paths labelled (a, a), contradicting the determinism of our automaton. Note that if we have multiple circuits we can alter the form of our automaton to get an equivalent automaton where all the circuits have the same length and are the same distance away from the start state, whilst maintaining the determinism of our automaton.

Lemma 6.3.3. *If an acceptor automaton for a unary graph automatic semigroup has distinct circuits then we can find an equivalent automaton such that all circuits have the same length p and each circuit is the same distance from the start state. Moreover, each accept state will accept words of a different remainder when their lengths are considered modulo p.*

Proof. Suppose that $\mathscr{A}_x$ has k circuits. As our automaton is deterministic and we cannot have a $(\$, a)$-circuit, we can have at most one (a, a)-circuit, and the rest must be $(a, \$)$-circuits. We number the circuits from 1 to k and let circuit i have length u_i for $1 \leq i \leq k$. Let $p = \mathrm{lcm}(u_1, u_2, \ldots, u_k)$. Then we may construct an equivalent automaton where all circuits have length p by the following process. Suppose that the first circuit consists of states $\bar{s}_1, \bar{s}_2, \ldots, \bar{s}_u$. Then instead of closing the circuit, we may extend it by introducing new states $\bar{s}'_1, \bar{s}'_2, \ldots, \bar{s}'_u$ between states $\bar{s}_u$ and $\bar{s}_1$, where state $\bar{s}'_i$ behaves the same as state $\bar{s}_i$, meaning it has the same offshoots and is an accept state precisely if $\bar{s}_i$ is an accept state. This is illustrated in Figure 6.1. We may repeat this process n_1 times in order to get a circuit of length $u_1 n_1$ which accepts the same words as our original circuit. In a similar way we extend the ith circuit by repeating it n_i times. Then by choosing each of the n_i appropriately we can extend all circuits to have length p, where p is a common multiple of all the u_i.

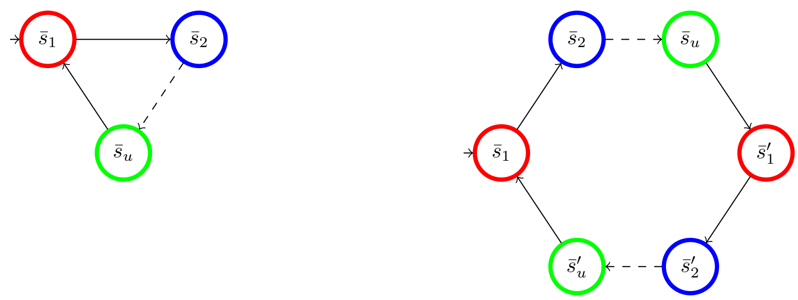


Figure 6.1: The process of extending the size of a circuit. States which are the same colour will have the same finite offshoots and will be the same type of state (accept or reject).

Note also that the paths prior to entering each circuit may be of different lengths. However, we can form an equivalent automaton by 'unravelling' part of the shorter circuit to ensure that the path length to both circuits is the same. Let states $\bar{t}_0, \bar{t}_1, \ldots, \bar{t}_k, \bar{s}_0$ be the sequence of states along the path from the start state to a circuit, with $\bar{s}_0$ being the state where we enter the circuit. Then if we wish to extend the finite path by $l = qp + r$ states for some $q \in \mathbb{N}$ and $0 \leq r < p$, and the circuit has states $\bar{s}_0, \bar{s}_2, \ldots, \bar{s}_{p-1}$, we introduce new states $\bar{s}_0^{(1)}, \ldots, \bar{s}_{p-1}^{(1)}, \ldots, \bar{s}_0^{(q)}, \ldots, \bar{s}_{p-1}^{(q)}, \bar{s}_1^{(q+1)}, \ldots, \bar{s}_r^{(q+1)}$ after state $\bar{t}_k$. Each state $\bar{s}_j^{(i)}$ behaves in the same way as $\bar{s}_j$, meaning it has the same offshoots and is an accept state precisely if $\bar{s}_j$ is an an accept state. However, instead of the transition from $\bar{s}_{p-1}^{(i)}$ to $\bar{s}_0^{(i)}$ which would close the circuit we have a transition with the same label from $\bar{s}_{p-1}^{(i)}$ to $\bar{s}_0^{(i+1)}$. Then from state $\bar{s}_r^{(q+1)}$ we will enter the circuit at state $\bar{s}_{r+1}$ if $r \neq p - 1$ and state $\bar{s}_0$ if $r = p - 1$. This process is shown in Figure 6.2. Thus if we have multiple circuits we may apply this process to all but the one with the longest path to the circuit in order to make the path lengths to all our circuits the same.

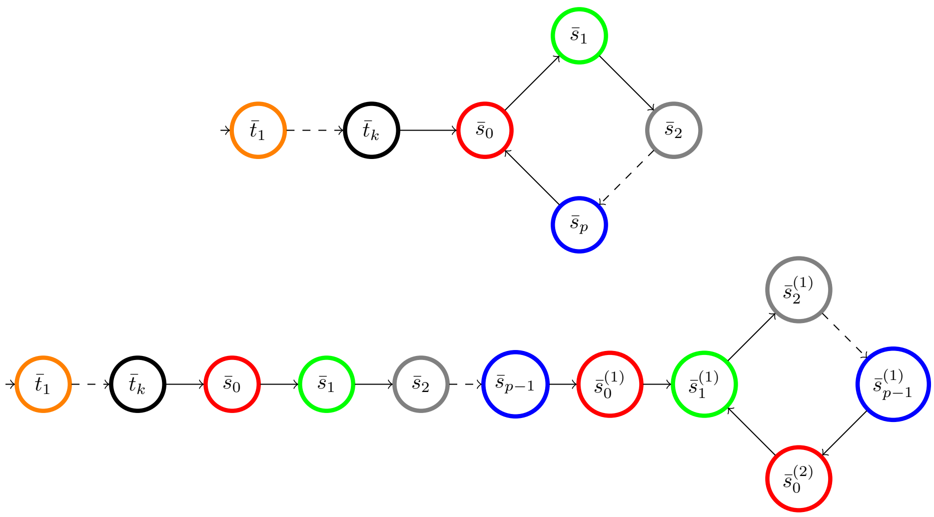


Figure 6.2: The process of extending the length of a path. States which are the same colour will have the same finite offshoots and will be the same type of state (accept or reject).

We now consider the first components of the words accepted by each circuit. Each state on a circuit accepts words of the form a^{c+pn+r} in the first component, where c is the length of the path before reaching a circuit and $0 \leq r < p$. Due to the fact that ν is a bijection, each such word must be accepted precisely once in the first component, as each element of the semigroup can be multiplied by x (and so each representative must be accepted by the automaton) resulting in precisely one element (and so a word cannot appear in the first component of two different words). Thus each state must accept words with a different remainder modulo p. □

We illustrate this process with an example of how it is applied to a specific automaton.

Example 6.3.4. We begin with the automaton in Figure 6.3. Note that this has the correct format to possibly be an acceptor automaton for some graph automatic semigroup.

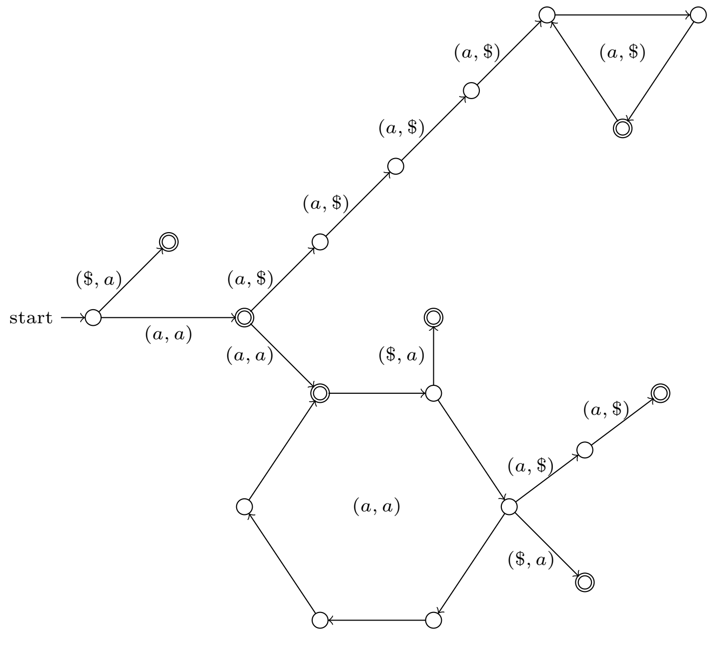


Figure 6.3: The original automaton

We wish to change this automaton into an equivalent automaton, where the paths to the circuits and the circuits themselves have the same length. Thus we wish to extend the upper circuit and the lower path. We begin by considering the circuit lengths. We wish both circuits to have length p, where p is a multiple of both current circuit lengths. Thus we may take $p = 6$, and so we only need to extend the $(a, \$)$-circuit. This is shown in Figure 6.4.

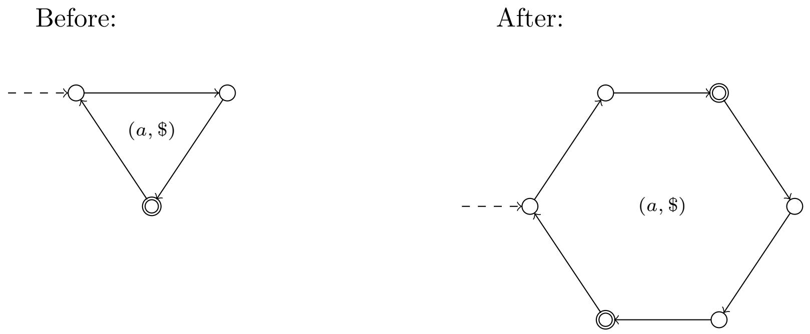


Figure 6.4: The $(a, \$)$-circuit before and after it is extended.

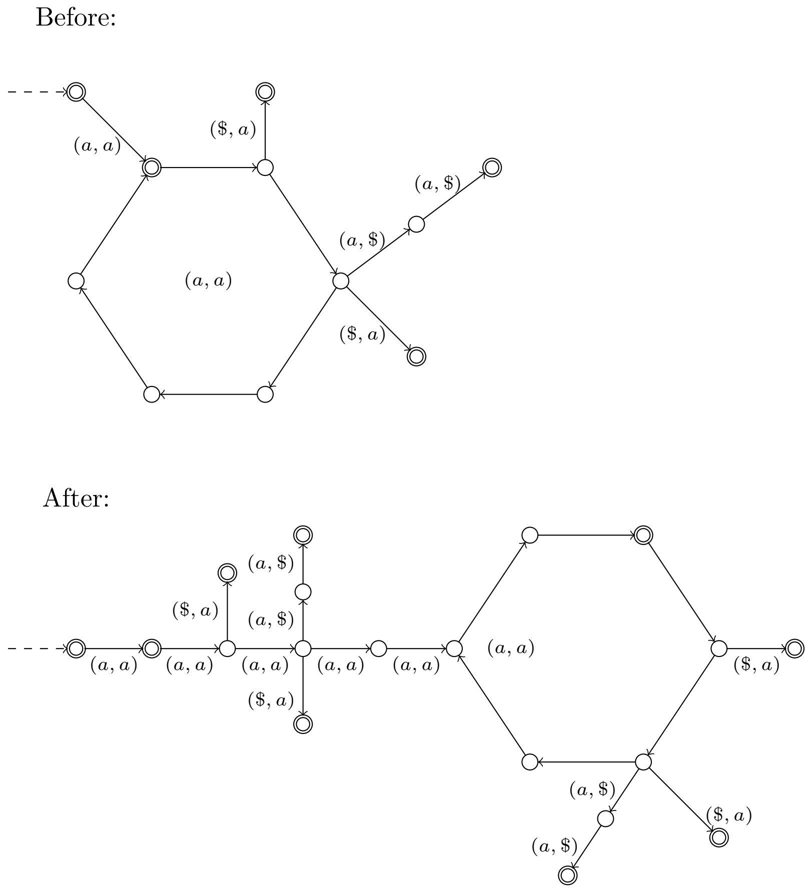


Figure 6.5: The path to the (a, a)-circuit before and after it is extended

Next we wish the paths before entering the circuits to be the same length. We take the shorter path, which leads to the (a, a)-circuit, and extend it to the length of the longer path. This extension is shown in Figure 6.5. Note that this results in the entry state to the circuit being changed. We may now combine these processes to get a new automaton, shown in Figure 6.6, which is equivalent to the original automaton.

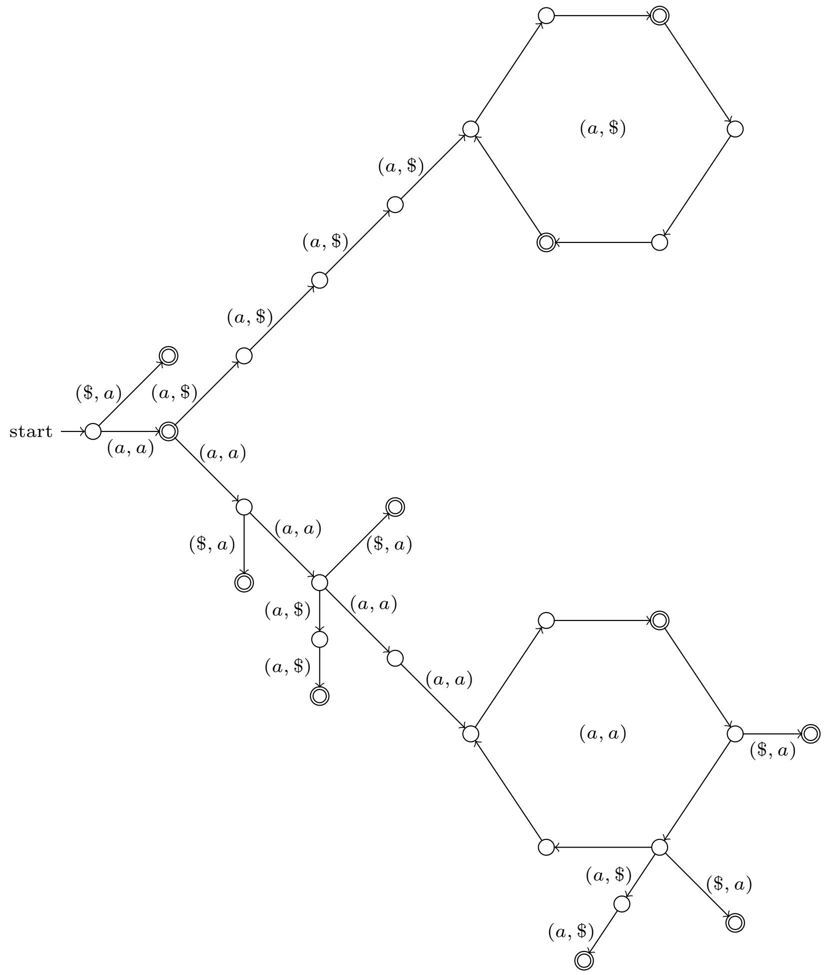


Figure 6.6: The new form of our automaton

Finally, we consider the behaviour of an acceptor automaton when the only circuit is an $(a, \$)$-circuit.

Lemma 6.3.5. *If an acceptor automaton for a unary graph automatic semi-*

group contains only one circuit, which is labelled $(a, \$)$*, then we can find an equivalent automaton where this circuit has length one (i.e. it is a loop labelled* $(a, \$)$*).*

Proof. Suppose $\mathscr{A}_x$ has only one circuit, which is labelled $(a, \$)$. This cannot have any offshoots by determinism of our automata. Now as every power of a must be accepted by some accept state, we must have that every state along our circuit is an accept state. This is equivalent to a circuit of length one. □

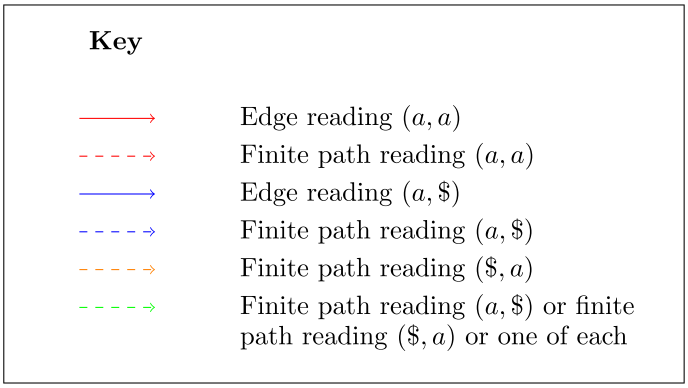
These lemmas now enable us to give a description of our acceptor automata.

Theorem 6.3.6. *Let* S *be an infinite semigroup with an injective unary graph automatic structure* (X, a, a^*, ν)*. Then any acceptor automaton for* S *is equivalent to one of the following, when we consider only the states and transitions which may lead to an accept state.*

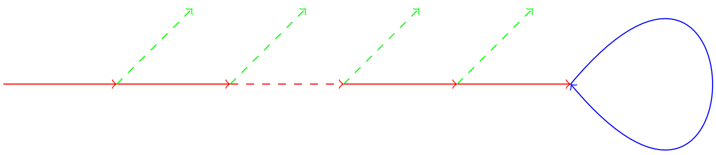
1. *A finite path labelled* (a, a)*, which may have finite offshoots labelled* $(a, \$)$ *or* $(\$, a)$*, then a finite path labelled* $(a, \$)$ *followed by a single loop of the form* $(a, \$)$*.*
2. *A finite path labelled* (a, a)*, followed by a single circuit of the form* (a, a)*. Both the path and the circuit may have finite offshoots labelled* $(a, \$)$ *or* $(\$, a)$*.*
3. *A finite path labelled* (a, a)*, which has finitely many branches labelled* $(a, \$)$ *leading to circuits labelled* $(a, \$)$ *and may end in a circuit labelled* (a, a)*. Both the* (a, a)*-path and the* (a, a)*-circuit may have finite offshoots labelled* $(a, \$)$ *or* $(\$, a)$*. Each circuit is the same distance from the start state, each circuit has the same length* q*, and each accept state will accept words in the first component of different remainders modulo* q*.*

The third of these structures is the most general form, with type 1 and 2 being specific cases of type 3. From this point on we will assume that any acceptor automaton for a unary graph automatic semigroup has one of these forms.

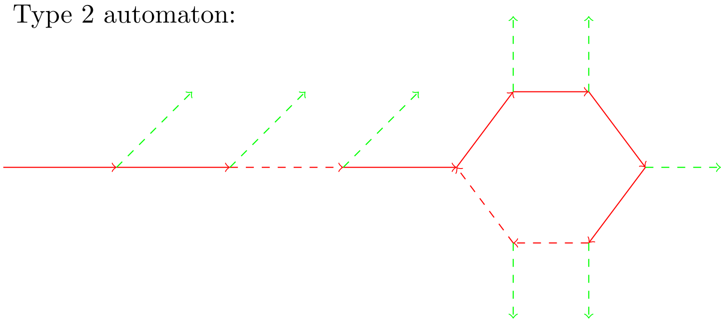
These automata are illustrated in Figure 6.7. Note that not every state will be an accept state, we merely illustrate the possible paths to reach accept states, and the accept states must be distributed in such a way that every power of a is accepted in the first component of precisely one word.



Type 1 automaton:



Type 2 automaton:



Type 3 automaton:

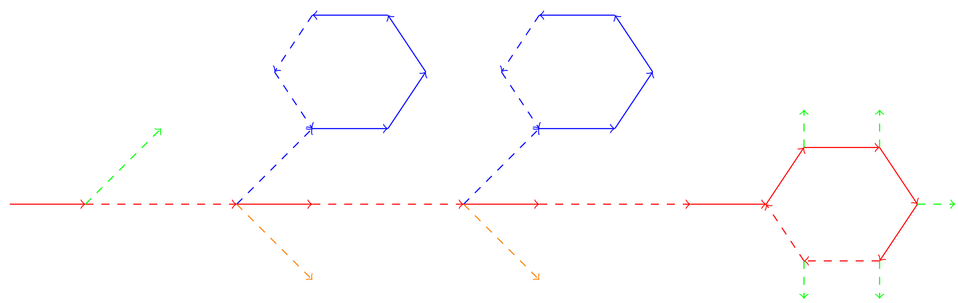


Figure 6.7: Forms of acceptor automata for unary graph automatic semigroups

6.3.1 Examples of Automata

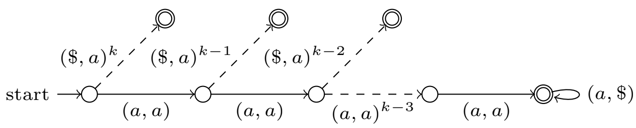
We shall now see examples of semigroups which have each type of automaton. We begin by noting that any infinite unary graph automatic semigroup with a zero will have an automaton of type 1.

Example 6.3.7. Let S be a unary graph automatic semigroup with a zero element, $z \in S$. Let S have injective unary graph automatic structure (X, a, a^*, ν) with $x \in X$. Then as z is a zero we have that

$$(a^*)_z = a^* \times a^k,$$

where $\nu(a^k) = z$. Therefore the automaton accepting $\mathscr{A}_z$ must be of type 1, and is illustrated in Figure 6.8. Note that each path to an accept state has length k.

Figure 6.8: Example of a type 1 automaton.



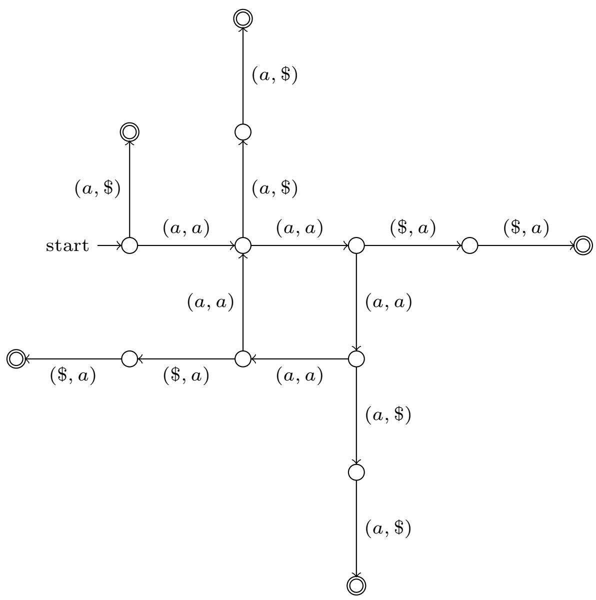
Next we return to one of our examples from Section 6.1 to provide us an example of an automaton of type 2.

Example 6.3.8. In Example 6.1.6 we saw that $\mathbb{Z}$ is a unary graph automatic semigroup with injective structure $(\{1, -1\}, a, a^*, \nu)$. Consider

$$(a^*)_1 = (a^2, a^2)^*\{(\epsilon, a^2), (a^3, a)\} \cup \{(a, \epsilon)\}.$$

The automaton $\mathscr{A}_1$ is a type 2 automaton, as illustrated in Figure 6.9.

Figure 6.9: Example of a type 2 automaton.



Finally we give another example of a unary graph automatic semigroup, in order to give an example of a type 3 automaton.

Example 6.3.9. Let S be the semigroup defined by the presentation

$$\langle x, y \mid xy = yx = x \rangle$$

and define $\nu : a^* \to S$ by

$$\nu(a^{2n+r}) = \begin{cases} x^{n+1}, & r = 0 \\ y^{n+1}, & r = 1. \end{cases}$$

Then as ν is a bijection we have that $(a^*)_=$ is regular, and we also have that

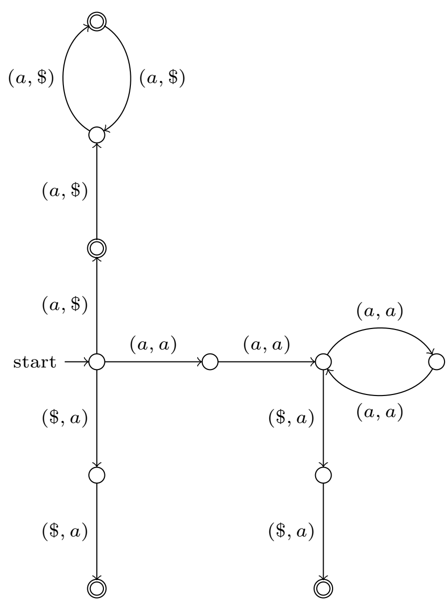
$$(a^*)_x = (a^2, a^2)^*(\epsilon, a^2) \cup ((a^2)^* a \times \{\epsilon\})$$

and

$$(a^*)_y = (a^2, a^2)^*(a, a^3) \cup (a^2, a^2)^*$$

are regular. Thus S is unary graph automatic with structure $(\{x, y\}, a, a^*, \nu)$. We now consider $\mathscr{A}_x$, and see that this is an example of an automaton of type 3, as shown in Figure 6.10.

Figure 6.10: Example of a type 3 automaton.



Thus we see that it is possible to get all three types of automaton as an acceptor automaton for some unary graph automatic semigroup.

6.4 Periodicity

A semigroup is *periodic* if every monogenic subsemigroup is finite. In this section we use the structure of our automata to show that infinite unary graph automatic semigroups are not periodic. We first note that we may express our semigroup in terms of normal forms, such that our set of normal forms is *prefix-closed*, that is if a word $x_1x_2\dots x_n$ belongs to our set of normal forms N, then $x_1x_2\dots x_k$ is also in N for any $1 \leq k \leq n$.

Lemma 6.4.1. *Let S be a semigroup generated by a finite set X. Then there is a set of unique normal forms $N \subseteq X^+$ for S such that N is prefix-closed.*

Proof. Let S be a semigroup which is finitely generated by a set X. To construct a set of prefix-closed normal forms for S, we begin by imposing an ordering on the elements of X. We then define the set of normal forms to be

$$N = \{w \in X^+ : \text{ if } v \in X^+ \text{ represents the same element of } S \text{ as } w, \text{ then } w <_s v\},$$

where $<_s$ represents the shortlex order. Suppose that N is not prefix-closed. Then there is some word $x_1x_2\dots x_n \in N$ such that $x_i \in X$ and $x_1x_2\dots x_m \notin N$ for some $m < n$. Then we must have another word $y_1y_2\dots y_l \in N$, with $y_i \in X$, which represents the same element of the semigroup as $x_1x_2\dots x_m$, with $y_1y_2\dots y_l <_s x_1x_2\dots x_m$. But then $y_1y_2\dots y_lx_{m+1}\dots x_n$ represents the same element of the semigroup as $x_1x_2\dots x_n$ and $y_1y_2\dots y_lx_{m+1}\dots x_n <_s x_1x_2\dots x_n$, and so $x_1x_2\dots x_n \notin N$, a contradiction. Thus N is prefix-closed. □

We now use this to show that the only periodic unary graph automatic semigroups are finite semigroups.

Theorem 6.4.2. *Infinite unary graph automatic semigroups are not periodic.*

Proof. Let S be a unary graph automatic semigroup with structure (X, a, a^*, ν). By Lemma 6.4.1, there is a set of unique normal forms, $N \subseteq X^*$, for the elements of S such that N is prefix-closed.

We consider the automata $\mathscr{A}_x$ for each $x \in X$. The automaton $\mathscr{A}_x$ contains circuits of length λ_x. Let λ be the lowest common multiple of the λ_x. Similarly, let γ_x be the maximum length of a path before reaching a circuit in $\mathscr{A}_x$ and let γ be the maximum of the γ_x. Now any word longer that $\lambda + \gamma$ must be accepted by a state on a circuit or an offshoot of a circuit of each $\mathscr{A}_x$.

Note also that if a word (a^n, a^m) is accepted by the automaton $\mathscr{A}_x$ then we have some bound b_x such that $m \leq n + b_x$. This is because we cannot have a $(\$, a)$-circuit, and so there is a bound on how much multiplication by x can increase the length of our representative. Let b be the maximum of the b_x. Then for any word (a^n, a^m) accepted by any of our acceptor automata we have $m \leq n + b$.

Consider $\mathscr{A}_x$. If this contains $(a, \$)$-circuits, then for each such circuit there is a fixed word a^{j_x} such that words of the form (a^k, a^{j_x}) are accepted by $\mathscr{A}_x$ for infinitely many k. There are at most finitely many such a^{j_x} associated with each $\mathscr{A}_x$, one for each $(a, \$)$-circuit. Let F_x be the set of all words accepted by any state before we reach a circuit in $\mathscr{A}_x$, together with the finitely many choices for a^{j_x}. If $\mathscr{A}_x$ does not contain an $(a, \$)$-circuit then F_x is merely the set of words accepted before reaching the circuit in $\mathscr{A}_x$. Let F be the union of the sets F_x. This set is finite, as each F_x is finite, and we let a^f be the longest word in this set.

As S is infinite, we have arbitrarily long products of generators representing distinct elements of our semigroup. We must also have elements represented by arbitrarily long words of a^*. Note that if an element is represented by a word of length $f + m$ then it must be a product of at least $\lceil m/b \rceil$ generators, as once we leave F we can only increase a word by at most b each time we multiply by a generator. We take a word a^p where $p > f + \lambda b$ and consider the element

$\nu(a^p) = w$. Then w has normal form

$$w = x_1 x_2 \dots x_n$$

such that all prefixes of w give different elements of the semigroup. There is a corresponding sequence of words

$$a^{p_1}, a^{p_2}, \dots, a^{p_n}$$

such that a^{p_l} represents $x_1 x_2 \dots x_l$ for $l = 1 \dots n$ and $a^p = a^{p_n}$. It follows that $(a^{p_i}, a^{p_{i+1}})$ is accepted by $A_{x_{i+1}}$ for $1 \leq i \leq n-1$.

Note that there is a point q_1 in this sequence such that from a^{q_1} onwards our words do not represent elements of F. There must be at least $(p-f)/b > \lambda$ elements left in the sequence. Consider a^{q_2}, where this is the first word in the sequence after a^{q_1} where $q_1 < q_2$. Continuing in this manner we get a subsequence

$$a^{q_1}, a^{q_2}, \dots, a^{q_k}$$

such that $a^{q_{i+1}}$ is the first word in the original sequence such that $q_i < q_{i+1}$. Note that $q_{i+1} \leq q_i + b$.

Now as w is longer than $f + \lambda b$, we must have at least λ increases in the sequence of words a^{p_i} after leaving the set F, and so we have that $k > \lambda$. Hence the sequence $a^{q_1}, a^{q_2}, \dots, a^{q_k}$ must contain two words a^{q_i} and a^{q_j} such that $q_i = q_j \bmod \lambda$. Then these two words are accepted by the same state of some automaton. Let $v = y_1 y_2 \dots y_k$ be the sequence of generators such that $\nu(a^{q_i})v = \nu(a^{q_j})$.

Applying v repeatedly gives an infinite sequence of elements

$$z, zv, zv^2, zv^3, \dots$$

represented by distinct words

$$a^{q_i}, a^{q_j}, a^{q_j+d}, a^{q_j+2d}, \ldots$$

where $d = q_j - q_i$. As our structure is injective, these must all be distinct elements of our semigroup, and so v is an element of infinite order. Thus S is not periodic. □

Note that infinite automatic semigroups are also never periodic. It is shown that automatic groups are not periodic in [18], and the same proof can be used for semigroups. This leads us to ask:

Question 6.4.3. Are infinite graph automatic semigroups ever periodic?

6.5 Proving Non-Unary Graph Automaticity

Unlike the general graph automatic case, we are able to show that certain semigroups are not unary graph automatic. We use the structure of our automata to determine a structure for certain types of unary graph automatic semigroups, which in turn we may use to find examples of semigroups which are not unary graph automatic. We first introduce some notation. For a semigroup S and sets $A, B \subseteq S$ we define the set

$$AB^{-1} = \{s \in S : sB \cap A \neq \emptyset\}.$$

This means that the set AB^{-1} is the set of all elements of S which are translated to an element of A by an element of B. We now use this to provide a structure for certain unary graph automatic semigroups.

Proposition 6.5.1. *Let S be an infinite unary graph automatic semigroup. Let $x \in X$ be an element of infinite order such that*

- *$\mathscr{A}_x$ is of type 2,*

- *$sx^i \neq s$ for any element $s \in S$ and any $i \in \mathbb{N}$, and*
- *for every finite set $F \subseteq S$, the set $F\langle x \rangle^{-1}$ is also finite.*

Then S can be written as Ax^* for some finite set $A \subseteq S$.

Proof. Let S be an infinite unary graph automatic semigroup with injective unary graph automatic structure (X, a, a^*, ν). Without loss of generality we may assume that $x \in X$.

We now consider the first components of the words accepted by each state of $\mathscr{A}_x$. Note that the injectivity of our structure ensures that each word in a^* will appear in the first component of precisely one word accepted by $\mathscr{A}_x$, and so we cannot have words with the same first component being accepted by different accept states. Thus we may use the accept states of $\mathscr{A}_x$ to partition our language.

Let $\bar{s}_1, \ldots, \bar{s}_j$, $\bar{t}_1, \ldots, \bar{t}_k$ be the accept states of $\mathscr{A}_x$, where the $\bar{s}_i$ are the states before reaching the circuit and the $\bar{t}_i$ are the states either on the circuit or on an offshoot of the circuit. We have a finite set F consisting of all the words in a^* which are the first component of a word accepted by the states $\bar{s}_i$, and finitely many sets $P_1, \cdots, P_k$ corresponding to the accept states $\bar{t}_1, \ldots, \bar{t}_k$ in the same way. In each P_i, the lengths of the words form an arithmetic progression.

Each arithmetic progression has common difference d, where d is the length of the circuit in $\mathscr{A}_x$. There is a natural ordering on each of the sets P_i, defined by the length of the words. Let $p_i \in P_i$ be the word in each of the sets P_i such that from p_i onwards no word is mapped by x to a word in F. This means that from this point onwards the words are always accepted by a state on the circuit of $\mathscr{A}_x$. Then from this point onwards we have $\nu(a^m)x = \nu(a^n)$ if and only if $\nu(a^{m+d})x = \nu(a^{n+d})$, and so above p_i each of our sets is mapped rigidly to another. Due to injectivity, each P_i has precisely one set P_j which is mapped to it. Hence we may follow a path between the sets, going from a^m to a^n if and only if $\nu(a^m)x = \nu(a^n)$. Note that there can be no closed paths, as a closed path would give us $\nu(a^i)x^k = \nu(a^i)$ for some $i, k \in \mathbb{N}$. Let p_i' be the first time we return to P_i after following this path through from p_i. The two are distinct, as there are no closed paths.

We must also have $p'_i > p_i$. If we have some $p'_j < p_j$ then this means we must have a path through the sets P_i which is always decreasing. Any such path must end in the set F. But then any element represented by a word from P_j can be mapped to the finite set F by repeated multiplication by x, as starting at any point above p_i will give the same decreasing pattern which must eventually end in F, and so $F\langle x\rangle^{-1}$ is not a finite set, a contradiction.

Now consider

$$A' = F \cup \bigcup_i \{a^n \in P_i : a^n \leq p'_i\}.$$

We can reach any word in any of our sets P_i by starting from an element in A' and following the path that comes from multiplying by x repeatedly. Hence we can reach any element of S by taking an element represented by a word in A' and multiplying repeatedly by x. Then if $A = \nu(A') = \{\nu(\alpha) : \alpha \in A\}$ we have $S = Ax^*$. □

This result will allow us to give several examples of graph automatic semigroups which are not unary graph automatic. We first note that if we have a right-cancellative element, our acceptor automaton must be of type 2.

Proposition 6.5.2. *An acceptor automata $\mathscr{A}_x$ for a right-cancellative element x must be of type 2.*

Proof. Let S be unary graph automatic with injective unary graph automatic structure (X, a, a^*, ν). Let $x \in X$ be right-cancellative and suppose that $\mathscr{A}_x$ is not of type 2. Then $\mathscr{A}_x$ has an $(a, \$)$-circuit and so accepts words (a^n, a^k) and (a^m, a^k) for some $n, m, k \in \mathbb{N}$ with $m \neq n$. So $\nu(a^n)x = \nu(a^k) = \nu(a^m)x$. Then as x is right-cancellative we have that $\nu(a^n) = \nu(a^m)$, a contradiction to the injectivity of our unary graph automatic structure. □

Note also that if our semigroup is cancellative then the second condition of Proposition 6.5.1 is also immediately satisfied. Thus for a cancellative semigroup we need only check the third condition.

Proposition 6.5.3. *The semigroup* $\mathbb{N}_0 \times \mathbb{N}_0$ *is not unary graph automatic.*

Proof. Suppose that $\mathbb{N}_0 \times \mathbb{N}_0$ is unary graph automatic. Note that $\mathbb{N}_0 \times \mathbb{N}_0$ is generated by the set $\{(0,1),(1,0),(0,0)\}$ and consider the automaton $\mathscr{A}_{(1,0)}$. Note also that $(1,0)$ has infinite order in $\mathbb{N}_0 \times \mathbb{N}_0$. Now $\mathbb{N}_0 \times \mathbb{N}_0$ is cancellative and so the first two conditions of Proposition 6.5.1 are satisfied. We check the third condition.

Consider some finite set $F = \{(n_1,m_1),(n_2,m_2)\dots(n_l,m_l)\} \subseteq \mathbb{N}_0 \times \mathbb{N}_0$. Now consider the set

$$\begin{aligned} F\langle(1,0)\rangle^{-1} &= \{(n,m) \in \mathbb{N}_0 \times \mathbb{N}_0 : (n,m)(1,0)^* \cap F \neq \emptyset\} \\ &= \{(n,m) \in \mathbb{N}_0 \times \mathbb{N}_0 : (n+k,m) \in F \text{ for some } k \in \mathbb{N}\}. \end{aligned}$$

Then if $(n,m) \in F\langle(1,0)\rangle^{-1}$ we must have that $m \in \{m_1, m_2, \dots, m_l\}$ and $n \leq \max\{n_1, n_2, \dots, n_l\}$. Thus there are only finitely many choices for n and m, and so $F\langle(1,0)\rangle^{-1}$ is finite.

Hence by Proposition 6.5.1 we can write $\mathbb{N}_0 \times \mathbb{N}_0 = A(1,0)^*$ for some finite set A. Let $A = \{(n_1,m_1),(n_2,m_2),\dots,(n_k,m_k)\}$. But if $(n,m) \in A(1,0)^*$ we must have $m \in \{m_1, m_2, \dots, m_k\}$. Thus $A(1,0)^*$ cannot contain every element of $\mathbb{N}_0 \times \mathbb{N}_0$, and so $\mathbb{N}_0 \times \mathbb{N}_0$ is not unary graph automatic. □

In fact, we may easily generalise this proof to show that $\mathbb{N}_0 \times \mathbb{N}_0 \times \cdots \times \mathbb{N}_0$ is not unary graph automatic for any finite number of copies of $\mathbb{N}_0$. We may show that if $\mathbb{N}_0 \times \mathbb{N}_0 \times \cdots \times \mathbb{N}_0$ was unary graph automatic then we could write it as $A(1,0,\dots,0)^*$ for some finite set A, which will restrict us to finitely many entries in all but the first component, giving a contradiction in the same way as in Proposition 6.5.3.

We next consider free semigroups, and see that they are not generally unary graph automatic.

Proposition 6.5.4. *The free semigroup* F_S *generated by the set* S *is unary* *graph automatic if and only if* $|S| = 1$.

Proof. Let F_S be the free semigroup generated by S. If $S = \{s\}$ then F_S is unary graph automatic as F_S is isomorphic to $\mathbb{N}$, which is unary graph automatic as we have seen in Example 6.1.5.

Now suppose that $|S| \geq 2$ and that F_S is unary graph automatic. Then we consider some element $s \in S$. This will have infinite order, and as free semigroups are cancellative we have that the first two conditions of Proposition 6.5.1 are satisfied. We now consider some finite set $Y \subset F_S$. Each element of F_S has a length, and as Y is finite there is a maximum length of an element in Y. Now each time we multiply any element of F_S by s we increase the length, and so the set

$$Y\langle s\rangle^{-1} = \{t \in F_S : ts^* \cap Y \neq \emptyset\}$$

must be finite.

Hence by Proposition 6.5.1 we can write $F_S = As^*$ for some element $s \in S$ and some finite set $A \subseteq F_S$. This means that all elements of F_S must end in s, a contradiction. Thus F_S is not unary graph automatic if $|S| > 1$. □

A similar method will also show that the free monoid on S is unary graph automatic if and only if $|S| = 1$.

Note that not all unary graph automatic semigroups can be written as Ax^*. In particular, groups cannot be written in this way. As all groups are cancellative, this means that all the acceptor automata for a group are of type 2. If we take a group G with an element of infinite order x, then cancellativity means that the condition $gx^i \neq g$ for any element $g \in G$ and any $i \in \mathbb{N}$ will also be satisfied. However, an infinite group will not be able to satisfy the final condition of Proposition 6.5.1. For example, we have seen in Example 6.1.6 that $\mathbb{Z}$ is unary graph automatic, but clearly we cannot write $\mathbb{Z} = Ax^*$ for any finite set A and element x. However, we do have that $\mathbb{Z}$ can be written as $A(x^* \cup (x^{-1})^*)$, where $A = \{1\}$. We now consider whether any unary graph automatic group can be written in such a form.

Note that if we can write a group G as $G = A(g^* \cup (g^{-1})^*$, this means that G contains $\langle g, g^{-1}\rangle$ as a subgroup of finite index. The *index* of a subgroup H in a group G, denoted $[G : H]$, is the number of *cosets* of H in G. A left coset of H in G is a set of the form $gH = \{gh : h \in H\}$, for some element $g \in G$. A right coset of H in G is a set of the form $Hg = \{hg : h \in H\}$, for some $g \in G$. Note that the number of left and right cosets of a subgroup is equal, and that cosets partition the group. If $[G : H] < \infty$ we say that H is a subgroup of finite index in G. A group is *virtually cyclic* if it has a cyclic subgroup of finite index.

In [6], the author shows that a group is unary graph automatic if and only if it is virtually cyclic. We provide an alternative proof that unary graph automatic groups contain a cyclic subgroup of finite index, using a similar method to Proposition 6.5.1 based on the forms of the acceptor automata.

Proposition 6.5.5. *Let G be an infinite unary graph automatic group. Then for any element of infinite order $g \in G$ we have that $G = A(g^* \cup (g^{-1})^*)$ for some finite set $A \subseteq G$.*

Proof. Let G be an infinite unary graph automatic group with unary graph automatic structure (X, a, a^*, ν). By Theorem 6.4.2, we know that we must have an element $g \in G$ of infinite order. Consider $\mathscr{A}_g$, which must be of type 2 by Proposition 6.5.2, and use this to partition a^* into a finite set F, plus finitely many arithmetic progressions $P_1, \ldots, P_k$, as in Proposition 6.5.1. Each arithmetic progression has a natural ordering based on the length of the words, and we may follow a path through the different sets in our partition by looking at the effect of multiplication by g. Let $p_i \in P_i$ be the point in each of the sets P_i such that from p_i onwards no word is mapped by g to a word in F, and let p_i' be the first time that we return to P_i on following the path from p_i. As G is a group it is cancellative, so we cannot have any closed paths meaning that $p_i \neq p_i'$. However, as the set $F\langle g\rangle^{-1}$ will be infinite, we cannot ensure that multiplying by g will give us an increasing path through these arithmetic progressions to allow us to reach all the elements of G. Thus we cannot ensure that we have $p_i' > p_i$.

However, we may use the automata $\mathscr{A}_{g^{-1}}$ to get arithmetic progressions $Q_1, \ldots, Q_k$, and note that if multiplying by g gives us a decreasing path through the progressions P_i then multiplying by g^{-1} must give us a corresponding increasing path through the progressions Q_i. Thus, we get the corresponding words q_i and q_i' based on the path through our progressions resulting from multiplying by g^{-1}, where $q_i < q_i'$ if and only if $p_i > p_i'$ for each $1 \leq i \leq k$.

Hence, similarly to Proposition 6.5.1, there is a finite set

$$A' = F \cup \bigcup_i \{a^n \in P_i : a^n \leq p_i'\} \cup \bigcup_i \{a^n \in Q_i : a^n \leq q_i'\}$$

such that if we start in this set we may find an infinite increasing path allowing us to reach any element by multiplying by either g or g^{-1}. Thus if $\nu(A') = A$, we have that $G = A(g^* \cup (g^{-1})^*)$. □

We also provide an alternative proof to the converse statement, showing directly that a virtually cyclic group is unary graph automatic by explicitly constructing a unary graph automatic structure.

Proposition 6.5.6. *Any group containing a cyclic subgroup of finite index is unary graph automatic.*

Proof. Let G be a group and let $H = \langle g \rangle$ be a subgroup of G, such that $[G : H] = q < \infty$. If H is finite then G is also finite, thus is unary graph automatic.

Now suppose that G is infinite. We can write any element of G as $g^n y$ for some $n \in \mathbb{Z}$ and $y \in Y$, where Y is a finite set $Y = \{y_0, y_1, \ldots, y_{q-1}\}$. Note that this means that G is finitely generated by $Y \cup \{g, g^{-1}\}$. We will show that there is a unary graph automatic structure for G. We define $\nu : G \to a^*$ by

$$\nu(a^{nq+r}) = \begin{cases} x^{\frac{n}{2}} y_r, & n \text{ even} \\ x^{-\frac{n+1}{2}} y_r, & n \text{ odd.} \end{cases}$$

This is a bijection, and so

$$R_= = (a,a)^*$$

is regular. We now consider R_x for some generator $x \in Y \cup \{g, g^{-1}\}$. We have that

$$R_x = \bigcup_{0 \le r \le q-1} K_r$$

where

$$K_r = \{(a^i, a^j) \in R_x : \nu(a^i) \in Hy_r\},$$

as the cosets of H partition the group and so give a corresponding partition of R. So each K_r is the contribution to R_x by the coset Hy_r for some $0 \le r \le q-1$. We will show that R_x is regular by showing that each of the K_r are regular.

Consider the effect of multiplying an element of Hy_r by x. We must have that $y_r x = g^k y_t$ for some $y_t \in Y$ and some $k \in \mathbb{Z}$. Suppose that $k \ge 0$. Then if n is even we have that

$$\nu(a^{nq+r})x = g^{\frac{n}{2}} y_r x = g^{\frac{1}{2}(n+2k)} y_t = \nu(a^{(n+2k)q+t})$$

and so $(a^{nq+r}, a^{(n+2k)q+t}) \in R_x$, and if n is odd and $n > 2k$ we have that

$$\nu(a^{nq+r})x = g^{-\frac{n+1}{2}} y_r x = g^{-\frac{1}{2}(n+1-2k)} y_t = \nu(a^{(n-2k)q+t})$$

and so $(a^{(n+2k)q+r}, a^{nq+t}) \in R_x$. Finally we have a finite set F_r of words recognising multiplication of elements represented by words a^{nq+r} where n is odd and $n \le 2k$. Thus the contribution to R_x by H_r is

$$K_r = (a^{2q}, a^{2q})^* \{(a^r, a^{2kq+t}), (a^{(1+2k)q+r}, a^{q+t})\} \cup F_r$$

and so, if k is positive, the contribution to R_g of Hy_r is regular.

We now consider the case where $k < 0$. In a similar way to the previous case we have that if n is even and $n > -2k$ we have that $(a^{(n-2k)q+r}, a^{nq+t}) \in R_x$, and if n is odd then we have that $(a^{nq+r}, a^{(n-2k)q+t}) \in R_x$. Finally we have a finite set F_r of words recognising multiplication of elements represented by words a^{nq+r} where n is even and $n \leq -2k$. Thus our contribution to R_x is

$$K_r = (a^{2q}, a^{2q})^* \{(a^{-2kq+r}, a^t), (a^{q+r}, a^{(1-2k)q+t})\} \cup F_r,$$

so when k is negative the contribution of Hy_r to R_x is also regular.

Now as the index of H in G is finite, there are only finitely many choices for r. So we have that

$$R_x = \bigcup_{0 \leq r \leq q-1} K_r$$

is a finite union of regular languages, thus R_x is regular.

Hence we have a unary graph automatic structure $(Y \cup \{g, g^{-1}\}, a, a^*, \nu)$ for G, thus G is unary graph automatic. □

6.6 Disjoint unions of the free monogenic semigroup

Disjoint unions of the free monogenic semigroup were studied in [1]. In this paper the authors show that disjoint unions of the free monogenic semigroup are always finitely presented and residually finite. In order to do this, the authors first introduce several results regarding how the elements from the different copies of the free monogenic semigroup must interact. Using these results, we show that all such semigroups are examples of unary graph automatic semigroups.

Let

$$S = \dot{\bigcup}_{x \in X} N_x$$

be a semigroup which is a disjoint union of finitely many free monogenic semigroups, with each N_x a copy of $\mathbb{N}$ generated by $x \in X$ for a finite set X. As in

[1], we define the sets

$$T(x, s, y) = \{t \in N_x : ts \in N_y\},$$

the set of elements in N_x that are sent to N_y upon right multiplication by s. Lemma 2.6 of [1] tells us that if such a set is infinite, it consists of an arithmetic progression plus finitely many other elements.

Lemma 6.6.1 (Lemma 2.6 of [1]). *If $T = T(x, s, y)$ is infinite then there exist sets $F = F(x, s, y)$ and $P = P(x, s, y)$ such that the following hold:*

1. $T = F \cup P$,

2. $P = \{x^{p+qt} : t \in \mathbb{N}_0\}$ *for some* $p = p(x, s, y), q = q(x, s, y) \in \mathbb{N}$ *and* $x^{p-q} \notin T$, *and*

3. $F \subseteq \{x, \ldots, x^{p-1}\}$ *is a finite set.*

We also require a further pair of lemmas which tells us about how our arithmetic progressions behave when multiplied by a generator. The first states that larger powers of one generator will be mapped to larger powers of another.

Lemma 6.6.2 (Lemma 2.3 of [1]). *If $x, y \in X$ and $s \in S$ are such that*

$$x^p s = y^u \text{ and } x^{p+q} s = y^v$$

for some $p, q, u, v \in \mathbb{N}$ then $u \leq v$.

The second shows that arithmetic progressions of one generator are mapped to arithmetic progressions of another generator.

Lemma 6.6.3 (Lemma 2.4 of [1]). *If*

$$x^p s = y^u \text{ and } x^{p+q} s = y^{u+v}$$

for some $x, y \in X, s \in S, p, q, u \in \mathbb{N}$ *and* $v \in \mathbb{N}_0$*, then*

$$x^{p+qt}s = y^{u+vt}$$

for all $t \in \mathbb{N}_0$*.*

We now use this to prove that S is unary graph automatic.

Theorem 6.6.4. *Every semigroup which is a disjoint union of finitely many copies of the free monogenic semigroup is unary graph automatic.*

Proof. Let $S = \dot{\bigcup}_{x \in X} N_x$ for $X = \{x_0, \ldots, x_{n-1}\}$. Let $R = a^*$ and define $\nu : a^* \to S$ by

$$\nu(a^{kn+i}) = x_i^k.$$

This is a bijection, so $R_=$ is regular. Let $s \in X$ and consider R_s. For each $x_i \in X$, we consider what happens when powers of x_i are multiplied by s.

We consider $T(x_i, s, x_j)$ for $x_i, x_j \in X$. This set is either finite, or stabilises into an arithmetic progression. If $T(x_i, s, x_j)$ is finite, then the corresponding contribution to R_s is finite, thus regular. Now if $T(x_i, s, x_j)$ is infinite then $T = F \dot{\cup} P$, for a finite set F and arithmetic progression P. We have a finite (hence regular) contribution from F, and so we need only consider the arithmetic progression $P(x_i, s, x_j)$. Let $(\alpha, \beta) \in R_s$ such that $\nu(\alpha) = x_i^p$ is the smallest power p such that $x_i^p \in P(x_i, s, x_j)$. Let $\nu(\beta) = x_j^u$. Then we also have $(\alpha', \beta') \in R_s$ with $\nu(\alpha') = x_i^{p+q}$ and $\nu(\beta') = x_j^{u+v}$, with $q \in \mathbb{N}$ and $v \in \mathbb{N}_0$ by Lemma 6.6.2. Thus by Lemma 6.6.3 we have

$$x_i^{p+qt}s = x_j^{u+vt}$$

for all $t \in \mathbb{N}_0$. Now $x_i^{p+qt} = \nu(a^{np+nqt+i})$ and $x_j^{u+vt} = \nu(a^{nu+vtn+j})$ and so this arithmetic progression is represented by the language

$$(a^{np+i}, a^{nu+j})(a^{nq}, a^{nv})^*.$$

As n, p, i, u and j are all fixed, $||a^{np+i}| - |a^{nu+j}||$ is finite, and so this language is regular by 1.1.1.

Now we repeat this process for each pair of generators from X, getting a regular contribution from each such pair. In this way we construct the whole of R_s. As X is finite, we have that R_s is a finite union of regular languages, thus is itself regular. Hence S is unary graph automatic. □

Note that not all unary graph automatic semigroups are of this form. We have already seen an example of such a semigroup, namely the free product of two trivial semigroups, in Example 6.1.7. However, this example is the disjoint union of two free monogenic semigroups (generated by st and ts), together with finitely many elements, namely s and t. Thus we ask if all unary graph automatic semigroups are the disjoint union of finitely many copies of the free monogenic semigroup with the addition of finitely many elements. The following example shows that this is not the case.

Example 6.6.5. Let S be the semigroup given by the presentation

$$\langle x, y \mid x^2 = y^2, xy = yx \rangle.$$

So $S = \{x^i, y, x^i y : i \in \mathbb{N}\}$. Define $\nu : a^* \to S$ by

$$\nu(a^{2i-r}) = \begin{cases} x^i, & r = 1 \\ x^i y, & r = 0. \end{cases}$$

This is a bijection, and so $R_=$ is regular. Then

$$\begin{aligned} R_x &= \{(a^k, a^l) : \nu(a^k)x = \nu(a^l)\} \\ &= \{(a^{2k-1}, a^{2k+1}) : k \in \mathbb{N}\} \cup \{(a^{2k}, a^{2k+2}) : k \in \mathbb{N}_0\} \end{aligned}$$

and

$$\begin{aligned} R_y &= \{(a^k, a^l) : \nu(a^k)y = \nu(a^l)\} \\ &= \{(a^{2k-1}, a^{2k}) : k \in \mathbb{N}\} \cup \{(a^{2k}, a^{2k+3}) : k \in \mathbb{N}_0\} \end{aligned}$$

are both regular. Hence S is unary graph automatic. We now show that S is not a disjoint union of finitely many free monogenic semigroups.

Suppose that S is a disjoint union of free monogenic semigroups, possibly with finitely many elements adjoined. We first show that S is not isomorphic to the free monogenic semigroup, nor is S the free monogenic semigroup with finitely many elements adjoined.

Suppose that S is isomorphic to the free monogenic semigroup. Then S is generated by a single element. We consider each of the possible generators. Note that any power of y can be rewritten as either a power of x or a power of x multiplied by y, thus when considering generators we need only consider those of the form x^i and $x^i y$. Our generator cannot be of the form x^i, as this will never give us an element of the form $x^j y$. If our generator has the form $x^i y$, then $(x^i y)^{2k} = x^{2ki+2k}$ and $(x^i y)^{2k+1} = x^{(2k+1)i+2k} y$ for any $k \in \mathbb{N}$. Thus we can never get an element of the form $x^{2ki+2k+1}$, in particular meaning we can never get an odd power of x, and so our generator cannot be of this form, meaning our semigroup is not isomorphic to the free monogenic semigroup. Additionally, each of these possibilities misses infinitely many elements, and so S is not isomorphic to the free monogenic semigroup with infinitely many elements adjoined.

Thus we must have at least two copies of the free monogenic semigroup. We consider the possible combinations of generators for the disjoint components.

If we have two copies generated by different powers of x, namely x^k and x^l, then we have that $(x^k)^l = (x^l)^k$, and so our monogenic semigroups are not disjoint. If we have a power of x, say x^k, together with y as our generators, then $(x^k)^2 = y^{2k}$ and so our monogenic semigroups are not disjoint. If we have an element $x^k y$, for some $k \in \mathbb{N}$, together with y as our generators, then $(x^k y)^2 = y^{2(k+1)}$ and so our monogenic semigroups are not disjoint. Finally, if we have $x^k y$ and $x^l y$, for some $k, l \in \mathbb{N}$ as generators then $(x^k y)^{l+1} = (x^l y)^{k+1}$, and again our monogenic semigroups are not disjoint. Note that we need not consider the case where y^k

Thus there is no possible combination of generators for which the monogenic semigroups produced are disjoint. Hence S is not a disjoint union of finitely many monogenic semigroups, nor is S a disjoint union of finitely many monogenic semigroups with finitely many elements adjoined.

So Theorem 6.6.4 and Example 6.6.5 together show that the class of unary graph automatic semigroups contains, but is not equal to, the class of semigroups which are disjoint unions of finitely many copies of the free monogenic semigroup.

This leads us to ask whether the properties exhibited in [1] for disjoint unions of finitely many copies of the free monogenic semigroup also hold for unary graph automatic semigroups in general. The authors show that such semigroups are always finitely presented, and so we ask:

Question 6.6.6. Are unary graph automatic semigroups always finitely presented?

A semigroup is *residually finite* if for any two distinct elements $s, t \in S$ there exists a homomorphism φ from S into a finite semigroup such that $\varphi(s) \neq \varphi(t)$. In [1] the authors show that disjoint unions of finitely many copies of the free monogenic semigroups are residually finite, and so we ask whether this is also the case for unary graph automatic semigroups.

Question 6.6.7. Are unary graph automatic semigroups residually finite?

6.7 Monogenic Subsemigroups

In this section we use the structure of our acceptor automata to show that any monogenic subsemigroup must be represented by a regular language. Note that this is not necessarily the case for general graph automatic semigroups, as shown in Example 3.1.3

Theorem 6.7.1. *Monogenic subsemigroups of unary graph automatic semigroups are regular.*

Proof. If S is a finite semigroup then any subsemigroup of S is also finite, thus is a regular subsemigroup. In particular this means that monogenic subsemigroups of finite semigroups are regular.

Now let S be an infinite unary graph automatic semigroup with structure (X, a, a^*, ν) with uniqueness. Let $T = \langle x \rangle$ be a monogenic subsemigroup of S. If T is finite then $\nu^{-1}T$ is also finite and so T is regular. We now consider the case where T is infinite. Consider R_x and the corresponding automaton $\mathscr{A}_x$. If this does not contain an (a, a)-circuit then then there are only finitely many solutions z to $sx = z$ for $s \in S$. This cannot be the case, as T is infinite and so we must have that $x^i x$ is distinct for each $i \in \mathbb{N}$. In particular, all elements of T are represented by different words, only finitely many of which will not be in the first component of a word accepted by a state which is an offshoot of the (a, a)-circuit.

We consider paths through the automaton, starting with some word $a_i \in a^*$ as the input on the first tape of $\mathscr{A}_x$. There is a unique word a_{i+1} such that $(a_i, a_{i+1}) \in \mathscr{A}_x$.

Let γ be the length of the path from the start state to the point where we enter the (a, a) circuit. We can find some power $x^j \in T$ such that for any $k \geq j$ we have that $|a_k| > \gamma$, thus the word representing x^j, and all subsequent powers of x will always be in the first component of a word accepted by a state which is an offshoot of the (a, a)-circuit. Now successive multiplication by x gives us an infinite sequence of words $a_j, a_{j+1}, a_{j+2}, \ldots$ representing successive powers of x, beginning with x^j. Let p be the length of the (a,a)-circuit, and let a_{j+k} for some $k \neq 0$ be the first word in this sequence such that $|a_j| = |a_{j+k}| \bmod p$. Now as $|a_j| = |a_{j+k}| \bmod p$, both (a_j, a_{j+1}) and (a_{j+k}, a_{j+k+1}) are accepted by the same state. This gives us that $|a_{j+1}| = |a_{j+k+1}| \bmod p$, and continuing in this way we get that $|a_{j+l}| = |a_{j+k+l}| \bmod p$ for $0 \leq l < k$. In this way we generate k arithmetic progressions, $a_{j+l}a^{dn}$ for $0 \leq l < k$, $d = |a_{j+k}| - |a_j|$ and $n \in \mathbb{N}_0$ which cover our subsemigroup T from the point x^j onwards. Thus our subsemigroup is represented by a subset of a^* consisting of a finite set plus finitely many arithmetic progressions. This is a regular subset of a^*, and so our semigroup T is a regular subsemigroup. □

Thus, unlike the general graph automatic case, we may always find a regular language representing a monogenic subsemigroup. We ask whether this result extends to subsemigroups in general.

Question 6.7.2. Are subsemigroups of unary graph automatic semigroups always regular?

6.8 Normal Forms

We have seen that certain types of unary graph automatic semigroups can be written as either Ax^* or $A(x^* \cup (x^{-1})^*)$ for a finite set A and an element x in Section 6.5. We have also seen that all disjoint unions of finitely many copies of the free monogenic semigroup are unary graph automatic, and it is clear that such semigroups can be written as $x_1^* \cup x_2^* \cup \cdots \cup x_n^*$. Thus we ask whether there is a set of uniform normal forms for the elements of any unary graph automatic semigroup.

We use our automata to find normal forms for the elements of a unary graph automatic semigroup.

Theorem 6.8.1. *Let S be a unary graph automatic semigroup. Then*

$$S = F \cup \left(\bigcup_{i=1}^{n} s_i c_i^* t_i \right),$$

where F is a finite set, $c_i \in S$, and $s_i, t_i \in S^1$.

Proof. If S is a finite semigroup then it is clear that we can write the elements of S in this form. In particular, take some $s \in S$ and let $F = S \setminus s^*$. Then $S = F \cup s^*$.

Now let S be an infinite unary graph automatic semigroup with structure (X, a, a^*, ν) with uniqueness. As in Theorem 6.4.2 we consider our words modulo λ, where λ is the lowest common multiple of the lengths of the circuits in the automata for our generators, $\mathscr{A}_x$ for $x \in X$. We let F be all elements which are accepted by either a finite path on some automaton, together with words a^j such that words of the form (a^k, a^j) are accepted by one of the $(a, \$)$-circuits of some $\mathscr{A}_x$. The rest of our elements can be partitioned into λ sets, based on the remainder modulo λ of the word representing the element. We call these partitions $P_1, P_2, \ldots, P_\lambda$, and note that there is a natural ordering on the elements in each set based on the length of their representative word from a^*.

We consider the infinite collection of elements zv^d for $d \in \mathbb{N}_0$, as constructed in Theorem 6.4.2. Let $v = x_1 x_2 \ldots x_k$. Then we have that each family of elements

$$z(x_1 x_2 \ldots x_k)^d$$

$$z x_1 (x_2 \ldots x_k x_1)^d$$

$$\vdots$$

$$z x_1 x_2 \ldots x_{k-1} (x_k x_1 x_2 \ldots x_{k-1})^d$$

for $d \in \mathbb{N}$ is represented by words whose length is a different remainder modulo λ. Thus in this way we reach elements in k of our partitions with elements of the form $s_i c_i^*$ for $1 \leq i \leq k$.

Now suppose that a partition P_i is only partially covered in this way, in particular let $a^{i_1}, a^{i_2}, \ldots$ be the representatives of our family of elements. Then as we follow the same path through our automata each time we go from a^{i_k} to $a^{i_{k+1}}$ we must increase the length of our representative by the same amount each time. Thus the elements must have been covered at regular intervals, so we have reached the elements represented by $a^{\lambda qd+i}$ for all $d \in \mathbb{N}$ and some fixed $1 < q \leq b$, where b is the maximum possible increase by a single automaton. Then if $z = z_0$ is represented by some element a^m we take $z_1 = \nu(a^{m+\lambda})$, $z_2 = \nu(a^{m+2\lambda})$, and so on, up to $z_{q-1} = \nu(a^{m+(q-1)\lambda})$ and use these elements in place of z to reach all elements in P_i by repeating the process above. This gives us additional representatives of the form $s_i c_i^*$ for $k < i \leq qk$.

Note that if P_i is mapped to P_j then it must be mapped rigidly, that is $\nu(a^i)x = \nu(a^j)$ if and only if $\nu(a^{i+\lambda})x = \nu(a^{j+\lambda})$, and so this method will cover all of $P_1, \ldots, P_k$. All our elements are expressed in a unique way, as at each stage we only cover new elements.

Now if $k \neq \lambda$ we consider the sets $P_{k+1}, \ldots, P_\lambda$ which we could not reach by starting with one of the z_i. If there is a sequence of generators $y_1, \ldots, y_l$ which takes us from an element $u \in P_i$ to $w \in P_j$ for some $1 \leq i \leq k$ and $j < k \leq \lambda$ then the entire set is mapped rigidly and so we can reach the whole of P_j from P_i by taking $sy_1 \ldots y_l$ for each $s \in P_i$.

We may now repeat this process until we have covered every set that it is possible to reach in this way, using any of P_1 to P_k as our starting point, and getting representatives of the form $s_i c_i^* t_i$ for at most $q\lambda$ values of i. If this process reaches all the sets P_i then we have our normal forms. Otherwise, the sets P_1 to P_l that we have covered plus F must form an ideal, as there is no way of moving from these to the remaining sets P_{l+1} to P_λ. We call this ideal I and consider the remaining elements, $S \setminus I$. Now as this is an infinite set and S is finitely generated we must be able to find a sequence of generators

$$x'_1, x'_2, \ldots, x'_n$$

and a corresponding sequence of words

$$a^{p_1}, a^{p_2}, \dots, a^{p_n}$$

such that $x'_1 x'_2 \dots x'_{k'}$ is represented by $a^{p_{k'}}$ and $(a^{p_i}, a^{p_{i+1}})$ is accepted by some $\mathscr{A}_x$ for $x \in X$, as in Theorem 6.4.2. We must be able to do this in such a way that we can construct an arbitrarily long sequence which avoids words from I, else $S \setminus I$, and hence S, cannot be finitely generated. Now we may proceed as in Theorem 6.4.2 to find a repeat modulo λ in the lengths of our representative words, and so find an infinite word of the form $z'v'$.

We now return to the beginning of this process, and use $z'v'$ to begin covering our remaining sets $P_{l+1}, \dots, P_\lambda$. If we cannot reach all of these in this way, then we get another ideal, and so can construct another infinite word. As we have at most λ sets to cover and each set can be covered with at most b different starting words, we need at most λb starting elements to cover all the sets P_i, and so this process will eventually have covered all such words.

Note that at each stage of this process we ensure that our representatives are unique as we only cover those elements which we have not yet reached, and so this gives us a set of normal forms for S. $\square$

Chapter 7

Constructions for Unary Graph Automatic Semigroups

In this chapter we will revisit several semigroup constructions in the context of unary graph automaticity. We will see that some constructions preserve unary graph automaticity as well as graph automaticity in general, whereas other constructions behave differently to the general case.

We first note that some of the results for general graph automatic semigroups immediately carry over into the unary case. In particular, we have that unary graph automaticity is preserved by regular subsemigroups.

Proposition 7.0.1. *A regular subsemigroup of a unary graph automatic semigroup is unary graph automatic.*

Proof. This immediately follows from Theorem 3.1.2, as the graph automatic structure of a regular subsemigroup uses the same alphabet as the graph automatic structure for the original semigroup. Thus if we have a regular subsemigroup of a unary graph automatic semigroup, this subsemigroup is also unary graph automatic. □

Thus any of our results in the general graph automatic case which were obtained by demonstrating that a subsemigroup is regular will immediately also hold for unary graph automatic semigroups.

7.1 Zero Unions

We begin by considering one of the simplest semigroup constructions, zero unions. In Proposition 5.1.1, it was shown that the zero union of two graph automatic semigroups is graph automatic if and only if both semigroups are also graph automatic. We have the analogous result for the unary case.

Theorem 7.1.1. *The zero union of two semigroups is unary graph automatic if and only if the two semigroups themselves are unary graph automatic.*

Proof. Let S and T be unary graph automatic semigroups. If both S and T are finite then $S \cup_0 T$ is finite and thus is unary graph automatic.

We next consider the case where S is finite and T is infinite. If $S = \{s_1, s_2, \ldots, s_k\}$ and T has structure $(X, a, L = a^*, \nu)$, then we define $\mu : c^* \to S \cup_0 T$ by

$$\mu(c^n) = \begin{cases} 0, & n = 0 \\ s_n, & 1 \leq n \leq k \\ \nu(a^{n-(k+1)}), & n > k \end{cases}$$

which is injective, so $(c^*)_=$ is regular. We also define a homomorphism $\varphi : a^* \times a^* \to c^* \times c^*$ by

$$\varphi(a^m, a^n) = (c^m, c^n).$$

Now $S \cup_0 T$ is generated by $S \cup X \cup \{0\}$ and we have that

$$(c^*)_0 = c^* \times \{\epsilon\},$$

which is regular. Then for $s \in S$ we have

$$(c^*)_s = \{(c^i, c^j) : 1 \leq i, j \leq k \text{ and } s_i s = s_j\} \cup (\{c^i : i > k\} \times \{\epsilon\}) \cup \{(\epsilon, \epsilon)\}.$$

The first set is finite, thus regular, and the second set is clearly regular. Hence $(c^*)_s$ is a regular language. Finally for $x \in X$ we have that

$$\begin{aligned}(c^*)_x &= \{(c^i, c^j) : (a^{i-(k+1)}, a^{j-(k+1)}) \in L_x\} \cup (\{c^i : 0 \leq i \leq k\} \times \{\epsilon\}) \\ &= (c^{k+1}, c^{k+1})\varphi(L_x) \cup (\{c^i : 0 \leq i \leq k\} \times \{\epsilon\}).\end{aligned}$$

This is regular, as homomorphisms of regular languages are regular, and so $S \cup_0 T$ is unary graph automatic. Similarly, if S is infinite and T is finite then $S \cup_0 T$ can be shown to be unary graph automatic in the same way.

Finally we consider the case where S and T are both infinite, with structures $(X, a, L = a^*, \nu_1)$ and $(Y, b, K = b^*, \nu_2)$ respectively. We define homomorphisms $\varphi_1 : a^* \to c^*$ and $\varphi_2 : b^* \to c^*$ by

$$\varphi_1(a^n) = c^{2n}$$

and

$$\varphi_2(b^n) = c^{2n}.$$

Then

$$c^* = c\varphi_1(a^*) \cup c^2\varphi_2(b^*) \cup \epsilon,$$

and we define $\nu : c^* \to S \cup_0 T$ by

$$\nu(c^n) = \begin{cases} 0, & n = 0 \\ \nu_1(a^{(n-1)/2}), & n \text{ odd} \\ \nu_2(b^{(n-2)/2}), & n \text{ even and } n > 0 \end{cases}$$

and this is a bijective map. Let $c^* = R$. By injectivity, $R_=$ is regular. Now

$S \cup_0 T$ is generated by $X \cup Y \cup \{0\}$ and we have that

$$R_0 = R \times \{\epsilon\},$$

which is regular. We now extend our homomorphisms to our two-tape languages, defining $\bar{\varphi}_1 : a^* \times a^* \to c^* \times c^*$ and $\bar{\varphi}_2 : b^* \times b^* \to c^* \times c^*$ by

$$\bar{\varphi}_1(a^m, a^n) = (c^{2m}, c^{2n})$$

and

$$\bar{\varphi}_2(b^m, b^n) = (c^{2m}, c^{2n}).$$

Then for $x \in X$ we have that

$$R_x = (c, c)\bar{\varphi}_1(L_x) \cup ((R \setminus \varphi_1(L)) \times \{\epsilon\}\})$$

and for $y \in Y$ we have that

$$R_y = (c^2, c^2)\bar{\varphi}_2(K_y) \cup ((R \setminus \varphi_2(K)) \times \{\epsilon\}).$$

Now, as homomorphisms of regular languages are regular, R_x and R_y are regular and $S \cup_0 T$ is unary graph automatic.

Conversely, suppose that $S \cup_0 T$ is unary graph automatic. Then, in the same way as in Proposition 5.1.1, we have that S and T are both regular subsemigroups of $S \cup_0 T$, and so S and T are both unary graph automatic by Proposition 7.0.1. □

7.2 Ordinal Sums

We now consider ordinal sums. In Proposition 5.2.1 we saw that ordinal sums preserve graph automaticity. We now show that the analogous result holds in the unary case.

Proposition 7.2.1. *The ordinal sum of two semigroups is unary graph automatic if and only if the two semigroups themselves are unary graph automatic.*

Proof. Let S and T be unary graph automatic semigroups. Consider their ordinal sum U, with ordering $S > T$. If both S and T are finite, then so is their ordinal sum, hence U is unary graph automatic.

We now consider the case where we have one finite and one infinite semigroup. First suppose that S is infinite with injective graph automatic structure $(X, a, L = a^*, \nu)$, and $T = \{t_1, t_2, \ldots, t_k\}$ is finite. Define $\mu : c^* \to U$ by

$$\mu(c^n) = \begin{cases} t_{n+1}, & 0 \leq n \leq k-1 \\ \nu(a^{n-k}), & n \geq k \end{cases}$$

which is injective, thus $(c^*)_=$ is regular. Define a homomorphism $\varphi : a^* \times a^* \to c^* \times c^*$ by

$$\varphi(a^m, a^n) = (c^m, c^n).$$

Then for $x \in X$ we have

$$\begin{aligned}(c^*)_x &= \{(c^i, c^j) : (a^{i-k}, a^{j-k}) \in L_x\} \cup \{(c^i, c^i) : 0 \leq i \leq k-1\} \\ &= (c^k, c^k)\varphi(L_x) \cup \{(c^i, c^i) : 0 \leq i \leq k-1\}\end{aligned}$$

and for $t_m \in T$ we have

$$(c^*)_{t_m} = (\{c^i : i > k\} \times \{c^{m-1}\}) \cup \{(c^i, c^j) : 0 \leq i \leq k-1 \text{ and } t_{i+1}t_m = t_{j+1}\}.$$

These are both regular, thus U is unary graph automatic.

Now suppose that $S = \{s_1, s_2, \ldots, s_k\}$ is finite, and T is graph automatic with injective graph automatic structure $(Y, b, K = b^*, \mu)$. Define $\nu : c^* \to U$ by

$$\nu(c^n) = \begin{cases} s_{n+1}, & 0 \leq n \leq k-1 \\ \mu(a^{n-k}), & n \geq k \end{cases}$$

which is injective, thus $(c^*)_=$ is regular. Define a homomorphism $\varphi : b^* \times b^* \to c^* \times c^*$ by

$$\varphi(b^m, b^n) = (c^m, c^n).$$

Then for $s_m \in S$ we have

$$(c^*)_{s_m} = \{(c^i, c^j) : 0 \leq i \leq k-1 \text{ and } s_{i+1}s_m = s_{j+1}\} \cup \{(c^i, c^i) : i \geq k\}$$

and for $y \in Y$ such that $\nu^{-1}(y) = \eta$ we have

$$\begin{aligned}(c^*)_y &= (\{c^i : 0 \leq i \leq k-1\} \times \{\eta\}) \cup \{(c^i, c^j) : (b^{i-k}, b^{j-k}) \in K_y\} \\ &= (\{c^i : 0 \leq i \leq k-1\} \times \{\eta\}) \cup (c^k, c^k)\varphi(K_y).\end{aligned}$$

These are both regular languages, and so U is unary graph automatic.

Finally we consider the case where S and T are both infinite unary graph automatic semigroups with injective structures $(X, a, L = a^*, \nu_1)$ and $(Y, b, K = b^*, \nu_2)$ respectively. Let $R = c^*$ for some c. Then define homomorphisms $\varphi_1 : L \to R$ and $\varphi_2 : K \to R$ by

$$\varphi_1(a^n) = c^{2n}$$

and

$$\varphi_2(b^n) = c^{2n}.$$

Then

$$R = \varphi_1(a^*) \cup c\varphi_2(b^*).$$

Now define $\nu : R \to U$ by

$$\nu(c^{2q+r}) = \begin{cases} \nu(a^q), & r = 0 \\ \mu(b^q), & r = 1 \end{cases}$$

This is injective, so $R_=$ is regular. We extend our homomorphisms to our two-tape languages, defining $\bar{\varphi}_1 : a^* \times a^* \to c^* \times c^*$ and $\bar{\varphi}_2 : b^* \times b^* \to c^* \times c^*$

by

$$\bar{\varphi}_1(a^m, a^n) = (c^{2m}, c^{2n})$$

and

$$\bar{\varphi}_2(b^m, b^n) = (c^{2m}, c^{2n}).$$

Now for $x \in X$ we have that

$$R_x = \bar{\varphi}_1(L_x) \cup (c, c)\bar{\varphi}_2(K_=),$$

and for $y \in Y$ we have that

$$R_y = (\varphi_1 L \times \{\eta\}) \cup (c, c)\bar{\varphi}_2(K_y),$$

where $\eta \in R$ is the unique word such that $\nu(\eta) = y$. Both of these are regular, thus we have a unary graph automatic structure $(X \cup Y, c, R = c^*, \nu)$ for U.

Conversely, if $S > T$ is unary graph automatic then both S and T are regular subsemigroups, in the same way as in Proposition 5.2.1. Thus they are both unary graph automatic by Proposition 7.0.1. □

This allows us to show that unary graph automaticity is preserved by adjoining identities and zeros, as in the general case.

Proposition 7.2.2. *A semigroup S is unary graph automatic if and only if S^1 is unary graph automatic.*

Proof. Suppose that S is unary graph automatic. Then S^1 is the ordinal sum of $\{1\}$ and S, with ordering $\{1\} > S$, so by Proposition 7.2.1 we have that S^1 is unary graph automatic.

Conversely, suppose S^1 is unary graph automatic. Then S is a subsemigroup of finite Rees index and as in the proof of Theorem 3.3.1 it was shown that subsemigroups of finite Rees index of graph automatic semigroups are regular, we have that S is unary graph automatic by Proposition 7.0.1. □

Proposition 7.2.3. *A semigroup S is unary graph automatic if and only if S^0 is unary graph automatic.*

Proof. Suppose that S is unary graph automatic. Then S^0 is the ordinal sum of $\{0\}$ and S with ordering $S > \{0\}$, so by Proposition 7.2.1 we have that S^0 is unary graph automatic.

Conversely, suppose that S^0 is unary graph automatic. Then S is a subsemigroup of finite Rees index and so we have that S is regular subsemigroup of S^0, as in the proof of Theorem 3.3.1 it was shown that subsemigroups of finite Rees index of graph automatic semigroups are regular, and so S unary graph automatic by Proposition 7.0.1. □

7.3 Semidirect and Direct Products

We have already seen from Proposition 6.5.3 that, unlike the general graph automatic case, it is possible to take the direct product of two unary graph automatic semigroups but have the product not be graph automatic. Hence, as direct products are a special case of semidirect products, unary graph automaticity is not preserved in general by semidirect products. As this example involves two infinite semigroups, we consider whether semidirect products preserved unary graph automaticity when one of our semigroups is finite.

As in the general graph automatic case in Theorem 4.2.2, we see that in the case where we have a left action and S is finite then unary graph automaticity is also preserved by the semidirect product $T \ltimes_\tau S$.

Theorem 7.3.1. *Let S and T be unary graph automatic semigroups, where S is finite. If the semidirect product $T \ltimes_\tau S$ is finitely generated then it is unary graph automatic.*

Proof. If S and T are both finite then $T \ltimes_\tau S$ is also finite, thus is unary graph automatic.

We now consider the case where T is infinite. Let $S = \{s_0, \ldots, s_{r-1}\}$ be a finite semigroup with structure $(S, a, R = a^*, \nu)$, where $\nu(a^i) = s_{i \bmod r}$. Let T

be an infinite unary graph automatic semigroup with structure $(X, b, K = b^*, \mu)$. Suppose that $T \ltimes_\tau S$ is finitely generated by a set Y.

Let $L = c^*$ and define $\psi : L \to T \ltimes_\tau S$ by

$$\psi(c^n) = (\mu(b^{\lfloor n/r \rfloor}), \nu(a^n)).$$

This is injective, so $L_=$ is regular.

We also define homomorphisms $\varphi_1 : a^* \times a^* \to c^* \times c^*$ and $\varphi_2 : b^* \times b^* \to c^* \times c^*$ by

$$\varphi_1(a^i, a^j) = (c^i, c^j)$$

and

$$\varphi_2(b^i, b^j) = (c^{ir}, c^{jr}).$$

Now let $(t, s) \in Y$. Then

$$\begin{aligned} L_{(t,s)} =& \{(c^n, c^m) : \mu(b^{\lfloor n/r \rfloor})({}^{\nu(a^n)}t) = \mu(b^{\lfloor m/r \rfloor}) \text{ and } \nu(a^n)s = \nu(a^m)\} \\ =& \{(c^n, c^m) : (b^{\lfloor n/r \rfloor}, b^{\lfloor m/r \rfloor}) \in K_{{}^{\nu(a^n)}t} \text{ and } (a^n, a^m) \in R_s\} \\ =& \bigcup_{u \in S} (\{(c^i, c^j) : 0 \leq i, j \leq r-1\}\varphi_2(K_{{}^ut}) \cap \varphi_1(R_s \cap (\nu^{-1}(u) \times a^*))). \end{aligned}$$

Thus $L_{(s,t)}$ is regular, and so $T \ltimes_\tau S$ is unary graph automatic with structure (Y, c, c^*, ψ). □

Next we consider the case where we have a right action. In Theorem 4.2.3 we saw that graph automaticity was preserved by a semidirect product using a right action as long as the relevant homomorphisms were automatic. However, Proposition 6.5.3 shows that this cannot be the case for unary semidirect products in general, as the direct product is the semidirect product $S \rtimes_\tau T$ where τ is trivial. Thus we consider whether a similar result holds when we restrict ourselves to the case where one of our semigroups is finite.

Theorem 7.3.2. *Let S and T be unary graph automatic semigroups. If*

- *$S \rtimes_\tau T$ is finitely generated by a set Y,*
- *at least one of S and T is finite, and*
- *$\tau(s)$ is automatic with respect to the graph automatic structure of T for every $s \in S$ such that $(s,t) \in Y$ for some $t \in T$,*

then the semidirect product $S \rtimes_\tau T$ is unary graph automatic.

Proof. If S and T are both finite then $S \rtimes_\tau T$ is also finite, thus is unary graph automatic.

Now suppose that $S = \{s_0, s_1, \ldots, s_{r-1}\}$ is a finite semigroup with structure $(S, a, R = a^*, \nu)$, where $\nu(a^i) = s_{i \bmod r}$. Let T be an infinite unary graph automatic semigroup with structure $(X, b, K = b^*, \mu)$. Suppose that $S \rtimes_\tau T$ is finitely generated by a set Y.

Let $L = c^*$ and define $\psi : L \to S \rtimes_\tau T$ by

$$\psi(c^n) = (\nu(a^n), \mu(b^{\lfloor n/r \rfloor})).$$

This is injective, so $L_=$ is regular.

We define homomorphisms $\varphi_1 : a^* \times a^* \to c^* \times c^*$ and $\varphi_2 : b^* \times b^* \to c^* \times c^*$ by

$$\varphi_1(a^i, a^j) = (c^i, c^j)$$

and

$$\varphi_2(b^i, b^j) = (c^{ir}, c^{jr}).$$

Then

$$\begin{aligned} L_{(s,t)} =& \{(c^n, c^m) : \nu(a^n)s = \nu(a^m) \text{ and } \mu(b^{\lfloor n/r \rfloor})^s t = \mu(b^{\lfloor m/r \rfloor}\} \\ =& \{(c^n, c^m) : (a^n, a^m) \in R_s \text{ and there exists } k \in \mathbb{N}_0 \text{ such that} \\ & (b^{\lfloor n/r \rfloor}, b^k) \in E_s \text{ and } (b^k, b^{\lfloor m/r \rfloor}) \in K_t\} \\ =& \{(c^{l_1}, c^{l_2}) : 0 \leq l_1, l_2 \leq r-1\}\varphi_2(\{(b^i, b^j) : (b^i, b^k) \in E_s \text{ and} \\ & (b^k, b^j) \in K_t\}) \cap \varphi_1(R_s) \end{aligned}$$

where $E_s = \{(b^i, b^j) : \mu(b^i)^s = \mu(b^j)\}$ is the language recognising the endomorphism $\tau(s)$, which is regular by assumption. Thus $L_{(s,t)}$ is regular, and so (Y, c, c^*, ψ) is a unary graph automatic structure for $S \rtimes_\tau T$.

Finally we consider the case where T is finite. Let S be a unary graph automatic semigroup with structure $(Z, a, R = a^*, \nu)$, and let $T = \{t_0, \ldots t_{r-1}\}$. As T is finite it is unary graph automatic, with structure $(T, b, K = b^*, \mu)$, where $\mu(b^i) = t_{i \bmod r}$. Suppose that $S \rtimes_\tau T$ is finitely generated by a set Y. Let $L = c^*$ and define $\psi : L \to S \times T$ by

$$\psi(c^n) = (\nu(a^{\lfloor n/r \rfloor}), \mu(b^n)).$$

This is injective, so $L_=$ is regular. We also define homomorphisms $\theta_1 : a^* \times a^* \to c^* \times c^*$ and $\theta_2 : b^* \times b^* \to c^* \times c^*$ by

$$\theta_1(a^i, a^j) = (c^{ir}, c^{jr})$$

and

$$\theta_2(b^i, b^j) = (c^i, c^j).$$

Then

$$\begin{aligned} L_{(s,t)} =& \{(c^n, c^m) : \nu(a^{\lfloor n/r \rfloor})s = \nu(a^{\lfloor m/r \rfloor}) \text{ and } \mu(b^n)^s t = \mu(b^m)\} \\ =& \{(c^n, c^m) : (a^{\lfloor n/r \rfloor}, a^{\lfloor m/r \rfloor}) \in R_s \text{ and there exists } k \in \mathbb{N}_0 \text{ such that} \\ & (b^n, b^k) \in E_s \text{ and } (b^k, b^m) \in K_t\} \\ =& \{(c^{l_1}, c^{l_2}) : 0 \leq l_1, l_2 \leq r-1\}\theta_2(\{(b^i, b^j) : (b^i, b^k) \in E_s \text{ and} \\ & (b^k, b^j) \in K_t\}) \cap \theta_1(R_s) \end{aligned}$$

where $E_s = \{(b^i, b^j) : \mu(b^i)^s = \mu(b^j)\}$ is the language recognising the endomorphism $\tau(s)$, which is regular. Thus $L_{(s,t)}$ is regular, and so (Y, c, c^*, ψ) is a unary graph automatic structure for $S \rtimes_\tau T$. □

This allows us to show that taking the direct product of a finite semigroup with a unary graph automatic semigroup preserves unary graph automaticity.

Corollary 7.3.3. *If the direct product of a unary graph automatic semigroup and a finite semigroup is finitely generated then it is unary graph automatic.*

Proof. Let S and T be unary graph automatic. The direct product $S \times T$ is the semidirect product $S \rtimes_\tau T$, where τ is trivial. This means that all our actions $\tau(s)$ for $s \in S$ are the identity, and so are clearly automatic. Thus by Theorem 7.3.2, if one of S or T is finite the direct product is unary graph automatic. $\square$

We have seen that this is not necessarily the case when both our semigroups are infinite. In fact, we conjecture that the direct product of two infinite semigroups is never unary graph automatic.

Conjecture 7.3.4. The direct product of two unary graph automatic semigroups is unary graph automatic if and only if at least one semigroup is finite.

In the case of groups, we can show that this holds.

Theorem 7.3.5. *The direct product of two unary graph automatic groups is unary graph automatic if and only if at least one of the groups is finite.*

Proof. Let G and H be unary graph automatic groups, and suppose that G and H are both infinite. Let $G \times H$ be unary graph automatic. Then there is some element of infinite order $(g, h) \in G \times H$ and some finite set $A = \{(g_1, h_1), \ldots, (g_n, h_n)\} \subset G \times H$ such that

$$G \times H = A((g, h)^* \cup (g^{-1}, h^{-1})^*),$$

by Proposition 6.5.5.

As H is infinite, there exists an infinite collection of distinct elements

$$v_1, v_2, v_3, \cdots \in H.$$

Consider the infinite collection of elements $(u, v_i) \in G \times H$ for $i \geq 1$ and some fixed $u \in G$. Then each such element can be written as

$$(u, v_i) = (g_{k_i}, h_{k_i})(g, h)^{p_i}$$

for some $(g_{k_i}, h_{k_i}) \in A$. As A is finite, we must have some distinct v_i and v_j for which $(g_{k_i}, h_{k_i}) = (g_{k_j}, h_{k_j})$. Then, as the first components of $(u, v_i) = (g_{k_i}, h_{k_i})(g, h)^{p_i}$ and $(u, v_j) = (g_{k_j}, h_{k_j})(g, h)^{p_j}$ are equal, we must have that $u = g_{k_i} g^{p_i} = g_{k_j} g^{p_j}$ and so $p_i = p_j$, as $g_{k_i} = g_{k_j}$ and G is infinite so g is an element of infinte order. Thus we have that

$$v_i = h_{k_i} h^{p_i} = h_{k_j} h^{p_j} = v_j,$$

contradicting our assumption that $v_i \neq v_j$. Thus G and H cannot both be infinite.

The converse follows from Corollary 7.3.3. □

Alternatively, we may prove the forward direction of this theorem using the classification of unary graph automatic groups discussed in Section 6.5.

Proposition 7.3.6. *The direct product of two infinite groups is never unary graph automatic.*

Proof. Let G and H be unary graph automatic groups. Suppose that $G \times H$ is unary graph automatic. Then, by Proposition 6.5.5, we have that $G \times H$ contains a cyclic subgroup of finite index, S. Let $S = \langle (g, h) \rangle$, and note that $\langle g \rangle$ must be a subgroup of finite index in G and $\langle h \rangle$ is a subgroup of finite index in H. Then as $[G \times H : S] < \infty$ we must also have that

$$[(G \times H) \cap (\langle g \rangle \times \langle h \rangle) : S \cap (\langle g \rangle \times \langle h \rangle)] = [\langle g \rangle \times \langle h \rangle : S] < \infty.$$

Thus we have that the direct product of two infinite cyclic subgroups contains an infinite cyclic subgroup of finite index, a contradiction. □

In Section 6.5 we saw that certain graph automatic semigroups could be written in the form $S = Ax^*$, where A is a finite set and x is an element of infinite order. We have already seen that not all semigroups can be written this way, as groups do not have this form. We now use Corollary 7.3.3 to give a non-group example which cannot be expressed in this form.

Example 7.3.7. Let $Z = \{z_1, \ldots, z_k\}$ be a right-zero semigroup. Consider $S = \mathbb{N} \times Z$, where $\mathbb{N}$ is generated by x. Note that S is generated by the set $\{(x, z_i) : 1 \leq i \leq k\}$. Then as $\mathbb{N}$ is unary graph automatic and Z is finite we have that S is unary graph automatic by Corollary 7.3.3. Consider (x^n, z_i) for some $n \in \mathbb{N}$ and $1 \leq i \leq k$, and suppose that we can write S as $A(x^n, z_i)^*$ for some finite set A. But for any $(x^m, z_j) \in A$ we have that $(x^m, z_j)(x^n, z_i) = (x^{m+n}, z_i)$ and so $A(x^n, z_i)^*$ will only ever have finitely many elements which do not have z_i in the second component. Therefore this is not equal to S for any choice of z_i.

We have also seen that every disjoint union of finitely many free monogenic semigroups is unary graph automatic, in Section 6.6. Every such semigroup $S = \bigcup_{x \in X} N_x$ can clearly be expressed as $S = x_1^* \cup \cdots \cup x_k^*$ for some $x_1, \ldots, x_k \in X$. We again use Corollary 7.3.3 to show that this is not the case for every unary graph automatic semigroup.

Example 7.3.8. Let $C_2 = \{y, y^2\}$ be the cyclic group of order two and consider $S = \mathbb{N} \times C_2$, where $\mathbb{N}$ is generated by x. We have that S is generated by the set $\{(x, y), (x, y^2)\}$. Then as $\mathbb{N}$ is unary graph automatic and C is finite we have that S is unary graph automatic by Corollary 7.3.3. Suppose that $S = (x^{n_1}, y^{m_1})^* \cup (x^{n_2}, y^{m_2})^* \cup \cdots \cup (x^{n_k}, y^{m_k})^*$ for some $n_1, n_2, \ldots, n_k \in \mathbb{N}$ and $m_1, m_2, \ldots, m_k \in \{1, 2\}$. Then as $(x^{n_i}, y^{m_i})^2 = ((x^{n_1})^2, y^2)$ for both $m_i = 1$ and $m_i = 2$, we can never get all of the elements of the form $(x^{2^p n_1}, y)$ for $p \in \mathbb{N}$, a contradiction. Thus S cannot be written as $(x^{n_1}, y^{m_1})^* \cup (x^{n_2}, y^{m_2})^* \cup \cdots \cup (x^{n_k}, y^{m_k})^*$.

7.4 Free Products

We have already seen that free semigroups are generally not unary graph automatic. We consider free products of unary graph automatic semigroups, and see that they only preserve unary graph automaticity if we take the free product of two trivial semigroups.

Theorem 7.4.1. *The free product of two semigroups is unary graph automatic if and only if both semigroups are trivial.*

Proof. Example 6.1.7 shows that the free product of two trivial semigroups is unary graph automatic.

To prove the converse, let S and T be finitely generated semigroups, generated by X and Y respectively. Suppose that $S*T$ is unary graph automatic with injective structure $(X \cup Y, a, L = a^*, \nu)$. We now consider the subsemigroup U consisting of all elements which end in an element of S. The preimage of U is given by

$$L_S = \bigcup_{x \in X} L_x^{(2)}$$

where $L_w^{(2)}$ signifies the second component of L_w for $w \in S * T$. This is regular, and so we have a unary graph automatic structure $(X, a, L_S, \nu|_{L_S})$ for U.

Consider $ts \in U$, for some $s \in S$ and $t \in T$. We will now show that the conditions of Proposition 6.5.1 are satisfied, and so there is a finite set A such that $U = A(ts)^*$. First note that ts has infinite order in U. As every element of U ends in an element of S we must have that $u_1 ts = u_2 ts$ implies that $u_1 = u_2$, for any $u_1, u_2 \in U$. Thus ts is right-cancellative in U, so the automaton $\mathscr{A}_x$ is of type 2 by Proposition 6.5.2, satisfying the first condition of Proposition 6.5.1. Every time we multiply an element of U by ts we increase the length of our element. Thus we cannot have any $u \in U$ such that $uts = u$ and the second condition is satisfied. Finally, if we have a finite set $F \subset U$ then the set

$$F\langle ts \rangle^{-1} = \{u \in U : u(st)^* \cap F \neq \emptyset\}$$

is finite, as there is a bound on the length of the elements of F, and each multiplication by ts increases the length of an element. Thus the third condition is satisfied.

Hence by Proposition 6.5.1 we can write $U = A(ts)^*$ for some finite set $A \subset U$. Similarly, we may define the subgroup V consisting of all elements ending in an element of T and find a unary graph automatic structure $(Y, a, L_T, \nu|_{L_T})$, such that we can write $V = B(st)^*$ for some finite set $B \subset V$.

Finally, as $S * T = U \cup V$ we have that $S * T = A(ts)^* \cup B(st)^*$. Thus all but finitely many elements of the semigroup end in either s or t. As the free product of two semigroups must have infinitely many elements ending in every element of S and every element of T, this is only the case if S and T are both trivial. □

7.5 Rees Matrix Semigroups

In Section 5.3 we saw that finitely generated Rees matrix semigroups over graph automatic semigroups are also graph automatic. We show that the same result holds when we consider unary graph automatic semigroups.

Theorem 7.5.1. *Let $S = M[T; I, J; P]$ be a Rees matrix semigroup, where T is a unary graph automatic semigroup, and I and J are finite. If S is finitely generated then it is unary graph automatic.*

Proof. Let T be a unary graph automatic semigroup with structure $(X, a, R = a^*, \nu)$. Let $|I| = k_I, |J| = k_J$ and $k = k_I k_J$. Define $\mu : b^* = L \to S$ by

$$\mu(b^m) = (m \bmod k_I, \nu(a^{\lfloor m/k \rfloor}), \lfloor (m \bmod k)/k_I \rfloor).$$

Then as we increase m, we first vary the first component, while keeping the second and third components fixed. After reaching $k_I - 1$, the first component will then cycle again, but this time with the final component increased by 1. This continues until we have cycled through all possible combinations in the first and third components while keeping the middle component fixed. Once all such combinations have been achieved, the middle component will change and the process begins again. Thus each element is represented by precisely one word. Hence μ is a bijection, and so $L_=$ is regular. Let Y be a finite generating set for S and let $(i, y, j) \in Y$. Then

$$\begin{aligned} L_{(i,y,j)} =& \{(b^m, b^n) : m \bmod k_I = n \bmod k_I, (a^{\lfloor m/k \rfloor}, a^{\lfloor n/k \rfloor}) \in R_{p_{li}y}, \\ & \text{and } j = \lfloor (n \bmod k)/k_I \rfloor\}, \end{aligned}$$

where $l = \lfloor (m \bmod k)/k_I \rfloor$.

We define a homomorphism $\psi : a^* \to b^*$ by

$$\psi(a^n) = b^{nk}$$

and so

$$\begin{aligned} L_{(i,y,j)} =& (b, b)^* \{(\$, b^{k_I})^*, (b^{k_I}, \$)^*\} \\ & \cap \bigcup_{l \in J} (\{(b^p, b^q) : 0 \leq p, q < k\} \psi(R_{p_{li}y}) \cap ((b^k)^* \{b^{k_I l + r} : 0 \leq r < k_I\} \times b^*)) \\ & \cap (b^* \times (b^k)^* \{b^{j k_I + r} : 0 \leq r < k_I\}). \end{aligned}$$

The first of these sets checks that n and m are equal modulo k_I, the second checks that $(a^{\lfloor m/k \rfloor}, a^{\lfloor n/k \rfloor}) \in R_{p_{li}y}$ for each choice of $l = \lfloor (m \bmod k)/k_I \rfloor$, and the third checks that $j = \lfloor (n \bmod k)/k_I \rfloor$.

As T is unary graph automatic, $R_{p_{li}y}$ is regular, and so $L_{(i,y,j)}$ is regular. Thus S is unary graph automatic with structure (Y, b, b^*, μ). □

Note that this is similar to the unary FA-presentable case, where unary FA-presentability is preserved by taking finite-by-finite Rees matrix constructions over a unary FA-presentable semigroup, and also by taking finite-by-countable Rees matrix semigroups over a finite semigroup, as shown in [11]. Note that to preserve graph automaticity we require both I and J to be finite in order for the Rees matrix semigroup to be finitely generated, by Proposition 5.3.1.

7.6 Semilattices of Semigroups

In Section 5.4, we saw that strong semilattices of graph automatic semigroups are graph automatic if and only if each of the homomorphisms is automatic. We now consider strong semilattices of unary graph automatic semigroups, and show that this construction also preserves unary graph automaticity.

Theorem 7.6.1. *Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups over a finite semilattice Y. Then S is unary graph automatic if and only if each S_u is unary graph automatic and each homomorphism $\varphi_{u,v} : S_u \to S_v$ is automatic with respect to the graph automatic structures for S_u and S_v.*

Proof. Let $S = \bigcup_{u \in Y} S_u$ be a strong semilattice of semigroups over the finite semilattice $Y = \{0, 1, \ldots, n-1\}$, such that each of the S_u is graph automatic. Thus each S_u has graph automatic structure (X_u, a_u, a_u^*, ν_u). Let $X = \bigcup_{u \in Y} X_u$. Then X is a generating set for S and is finite as each of the X_u is finite.

Let $R = b^*$, and define $\nu : b^* \to S$ by

$$\nu(b^{qn+r}) = \nu_r(a_r^q).$$

Then

$$\begin{aligned} R_= &= \{(b^{pn+u}, b^{qn+v}) : \nu_u(a_u^p) = \nu_v(a_v^q)\} \\ &= (b, b)^* \end{aligned}$$

as $\nu_u(a_u^p) = \nu_v(a_v^q)$ if and only if $u = v$ and $p = q$, due to the uniqueness of the original structures.

We define a homomorphism $\psi : a_u^* \times a_{uv}^* \to b^* \times b^*$ by

$$\psi(a_u^k, a_{uv}^l) = (b^{nk}, b^{nl}).$$

Now let $x_v \in X_v$. Note that if $s_u x_u = t$ in S for some $s_u \in S_u$ then $t \in S_{uv}$. Thus

$$\begin{aligned} R_{x_v} &= \bigcup_{u\in Y} (\{b^{pn+u}, b^{qn+w}) \in b^* \times b^* : \nu_u(a_u^p)x_v = \nu_w(a_w^q) \text{ and } w = uv\} \\ &= \bigcup_{u\in Y} \{(b^{pn+u}, b^{qn+uv}) \in b^* \times b^* : \varphi_{u,uv}(\nu_u(a_u^p))\varphi_{u,uv}(x_v) = \nu_{uv}(a_{uv}^q)\} \\ &= \bigcup_{u\in Y} (b^u, b^{uv})\psi\{(a_u^p, a_{uv}^q) : (\varphi_{uv}(a_u^p), a_{uv}^q) \in (a_{uv}^*)_t\} \\ &= \bigcup_{u\in Y} (b^u, b^{uv})\psi\{(a_u^p, a_{uv}^q) : (a_{uv}^d, a_{uv}^q) \in (a_{uv}^*)_t \text{ and } (a_u^p, a_{uv}^d) \in L\} \end{aligned}$$

where $t = \varphi_{u,uv}(x_v)$ is fixed, and

$$L = \{(a_u^k, a_{uv}^l) : \varphi_{u,uv}(\nu_u(a_u^k)) = \nu_{uv}(a_{uv}^l)\}$$

is the language recognising $\varphi_{u,uv}$. Then as L and $(a_{uv}^*)_t$ are regular, R_{x_v} is regular, and so S is unary graph automatic.

Conversely, suppose that S is graph automatic. Then each S_u is graph automatic by Corollary 5.4.3, and in particular S_u is shown to be graph automatic by showing that it is a regular subsemigroup. Hence by Proposition 7.0.1, we have that S_u is unary graph automatic for each S_u in Y. In addition, we have that each homomorphism $\varphi_{u,v}$ is regular with respect to the graph automatic structure of S_u and S_v as in Proposition 5.4.8. □

7.7 Bruck-Reilly Extensions

In Section 4.4 we saw that Bruck-Reilly extensions preserve graph automaticity if we have certain conditions on the homomorphism. We now show that the same does not hold for Bruck-Reilly extensions of unary graph automatic monoids.

Proposition 7.7.1. *The Bruck-Reilly extension of a monoid is never unary graph automatic.*

Proof. Consider the Bruck-Reilly extension $T = \mathrm{BR}(M, \theta)$ of a monoid M. Suppose that T is unary graph automatic. Then by Theorem 6.8.1, we have that

$$T = F \cup \left(\bigcup_{i=1}^{n} s_i u_i^* t_i \right),$$

where F is a finite set, $u_i \in T$, and $s_i, t_i \in T^1$. Now any element of T can also be written as $c^i m b^j$ for some $m \in M$, and $i, j \in \mathbb{N}_0$, where b and c are elements not in M. Thus given an element of T in the form $su^k t$ for some $s, t \in T^1$, $u \in T$ and $k \in \mathbb{N}_0$ we may replace each of s, t and u by an element of the form $c^i m b^j$. Thus all but finitely many elements of T can be written as

$$(c^{i_1} m_1 b^{j_1})(c^{i_2} m_2 b^{j_2})^k (c^{i_3} m_3 b^{j_3}) = c^{i_4} m_4 b^{j_4}$$

for some $m_i \in M$. Now if $i_2 > j_2$ then we can get arbitrarily large powers of c but our power of b will be fixed, and conversely if $i_2 < j_2$ we will get arbitrarily large powers of b but will fix the power of c. Finally, if $i_2 = j_2$ then our power of b and c will remain fixed. Thus it is not possible to get every possible combination of powers of b and c by starting with only finitely many elements. □

In particular, this allows us to show that the bicyclic monoid is not unary graph automatic.

Example 7.7.2. The bicyclic monoid is the Bruck-Reilly extension

$$B = \mathrm{BR}(\{1\}, \theta)$$

of the trivial monoid $\{1\}$ under the trivial homomorphism, thus is not unary graph automatic, by Theorem 7.7.1.

Chapter 8

Conclusion and Overview of Open Questions

In this thesis we have provided an introduction to the theory of graph automatic semigroups. We now recap some of the open problems and questions that we have encountered throughout this thesis, as an indication of the directions which further work may take.

Throughout this thesis we have demonstrated that various constructions are graph automatic. However, we do not have an effective way of determining whether a given semigroup is not graph automatic. Currently, the only way of saying that a semigroup is not graph automatic is if it is not finitely generated or does not have decidable word problem. In Chapter 6, we found that unary graph automatic semigroups must possess certain properties, enabling us to show that some semigroups were not unary graph automatic. In order to do this we considered the form of the acceptor automata for our semigroups. A similar approach for general graph automatic semigroups would be much more complicated, as the forms of the possible acceptor automata are much more complex. Nevertheless, having an effective way of showing that a semigroup is not graph automatic would be both interesting in its own right and also useful in helping to answer some of the other open problems regarding our semigroup constructions. Thus we ask:

Question 8.0.1. How can we show that a semigroup is not graph automatic?

In Chapter 3, we saw that it is possible to have a non-regular subsemigroup of a graph automatic semigroup, but that left ideals of graph automatic semigroups are always regular. We ask whether this is the case for right ideals also, and whether a right ideal of a graph automatic semigroup is necessarily graph automatic. Of course, if the answer to the first question is yes then the answer to the second must be yes also.

Question 8.0.2. Are finitely generated right ideals of graph automatic semigroups always regular subsemigroups? Are finitely generated right ideals of graph automatic semigroups necessarily graph automatic?

In fact we wish to go further, and ask whether all finitely generated subsemigroups of graph automatic semigroups are graph automatic. If the answer to this question was yes then it would immediately allow us to answer some of the open questions regarding constructions. If the answer was no, then this might still provide insight into how to answer some of the open questions regarding constructions, possibly by giving us examples which allow us to show that some of the constructions do not preserve graph automaticity.

Question 8.0.3. Are finitely generated subsemigroups of graph automatic semigroups necessarily graph automatic?

In Chapters 4 and 5, we considered whether various semigroup constructions preserved graph automaticity. We now highlight some of the major open problems regarding these constructions.

In Section 4.1, we saw that taking the free product of two graph automatic semigroups gives a graph automatic semigroup. We ask whether the converse holds. This is of particular interest when we consider it in comparison to automatic and FA-presentable semigroups. We have that a free product of two semigroups is automatic if and only if both semigroups are automatic, but a free product is FA-presentable if and only if both semigroups are trivial. As these results differ so much, it would be of interest to be able to fully compare them to the graph automatic case.

Question 8.0.4. If a free product of two semigroups is graph automatic, are both factors graph automatic?

Also of interest is the analogous question for direct products, that is if we have a direct product of two semigroups which is graph automatic, are both semigroups graph automatic? In Section 4.3 we saw that we could answer this question positively when our graph automatic semigroup was the direct product of a finite semigroup and a monoid. The next steps would perhaps be to answer this question for the direct product of a finite semigroup with an infinite semigroup, or for the direct product of two infinite monoids, with the aim of answering the question in general.

Question 8.0.5. If a direct product of two monoids is graph automatic, are both monoids graph automatic? If a direct product of a finite semigroup and an infinite semigroup is graph automatic, are both semigroups graph automatic? If a direct product of two arbitrary semigroups is graph automatic, are both semigroups graph automatic?

In Section 5.3, we consider Rees matrix semigroups. We saw that finitely generated Rees matrix semigroups and finitely generated Rees matrix semigroups with zero are both graph automatic if their base semigroup is graph automatic. We ask whether the converse also holds. A first step might be to show that if a Rees matrix semigroup over a group is graph automatic then the group is graph automatic.

Question 8.0.6. If a Rees matrix semigroup over a group is graph automatic, is the group graph automatic? If a Rees matrix semigroup over a semigroup is graph automatic, is the semigroup graph automatic?

In Chapters 6 and 7 we examined those semigroups which have a graph automatic structure over a single letter alphabet. We now consider some of the open problems for unary graph automatic semigroups.

In Section 6.6 we saw that the class of unary graph automatic semigroups contains the class of semigroups which are disjoint unions of finitely many free monogenic semigroups. In [1], the authors show that such semigroups are finitely presented and residually finite. We ask whether all unary graph automatic semigroups also possess these properties. In Section 6.8, we found a way of expressing a unary graph automatic semigroup by a uniform set of normal forms. It is possible that these normal forms may allow us to answer these questions, similarly to how normal forms were used to demonstrate these properties in [1].

Question 8.0.7. Are unary graph automatic semigroups always finitely presented? Are they residually finite?

In Section 6.5 we saw that unary graph automatic groups are precisely those groups with a cyclic subgroup of finite index. We ask whether we may, perhaps in a similar way, find a necessary and sufficient condition for a semigroup to be unary graph automatic.

Question 8.0.8. Can we classify all unary graph automatic semigroups?

Bibliography

[1] N. Abu-Ghazalh and N. Ruškuc. On disjoint unions of finitely many copies of the free monogenic semigroup. *Semigroup Forum*, 87:243–256, 2013.

[2] I. Andrade, L. Descalço, and Manuel A. Martins. Automatic structures for semigroup constructions. *Semigroup Forum*, 76:239–255, 2008.

[3] Isabel M. Araújo, Mário J.J. Branco, Vitor H. Fernandes, Gracinda M.S. Gomes, and N. Ruškuc. On generators and relations for unions of semigroups. *Semigroup Forum*, 63:49–62, 2001.

[4] H. Ayik and N. Ruškuc. Generators and relations of Rees matrix semigroups. *Proceedings of the Edinburgh Mathematical Society*, 42:481–495, 1999.

[5] G. Baumslag, S. M. Gersten, M. Shapiro, and H. Short. Automatic groups and amalgams. *Journal of Pure and Applied Algebra*, 76:229 – 316, 1991.

[6] A. Blumensath. Automatic structures. Diploma thesis, RWTH Aachen, 1999.

[7] Achim Blumensath and Erich Gradel. Finite presentations of infinite structures: Automata and interpretations. *Theory of Computing Systems*, 37:641–674, 2004.

[8] Alan J. Cain. Automatic semigroups and Bruck-Reilly extensions. *Acta Mathematica Hungaria*, 126:1 – 15, 2010.

[9] Alan J. Cain, Graham Oliver, Nik Ruškuc, and Richard M. Thomas. Automatic presentations for semigroups. *Information and Computation*, 207:1156 – 1168, 2009.

[10] Alan J. Cain, Graham Oliver, Nik Ruškuc, and Richard M. Thomas. Automatic presentations and semigroup constructions. *Theory of Computing Systems*, 47:568 – 592, 2010.

[11] Alan J. Cain, Nik Ruškuc, and Richard M. Thomas. Unary fa-presentable semigroups. *International Journal of Algebra and Computation*, 22, 2012.

[12] C. M. Campbell, E. F. Robertson, N. Ruškuc, and R. M. Thomas. Direct products of automatic semigroups. *Journal of the Australian Mathematical Society*, 69:19 – 24, 2000.

[13] C. M. Campbell, E. F. Robertson, N. Ruškuc, and R. M. Thomas. Automatic completely-simple semigroups. *Acta Mathematica Hungaria*, 95:201 – 215, 2002.

[14] Colin M. Campbell, Edmund F. Robertson, Nikola Ruškuc, and Richard M. Thomas. Automatic semigroups. *Theoretical Computer Science*, 250:365 – 391, 2001.

[15] A. H. Clifford and G. B. Preston. *The Algebraic Theory of Semigroups*, volume I. American Mathematical Society, 1961.

[16] L. Descalço and N. Ruškuc. On automatic Rees matrix semigroups. *Communications in Algebra*, 30:1207–1226, 2002.

[17] Luis Descalço. *Automatic Semigroups: Constructions and Subsemigroups.* PhD thesis, University of St Andrews, 2002.

[18] David B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Patterson, and W. P. Thurston. *Word Processing in Groups.* Jones and Bartlett Publishers, 1992.

[19] P. A. Grillet. *Semigroups: An Introduction to the Structure Theory.* Marcel Dekker, 1995.

[20] Michael Hoffman and Richard M. Thomas. Automaticity and commutative semigroups. *Glasgow Mathematical Journal*, 44:167 – 176, 2002.

[21] Michael Hoffmann, Nik Ruškuc, and Richard M. Thomas. Automatic semigroups with subsemigroups of Rees index. *International Journal of Algebra and Computation*, 12:463–476, 2002.

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

[23] John M. Howie. *Automata and Languages.* Oxford University Press, 1991.

[24] John M. Howie. *Fundamentals of Semigroup Theory.* Oxford University Press, 1995.

[25] Olga Kharlampovich, Bakhadyr Khoussainov, and Alexi Miasnikov. From automatic structures to automatic groups. *Groups, Geometry, and Dynamics*, 8:157–198, 2014.

[26] Bakhadyr Khoussainov and Anil Nerode. Automatic presentations of structures. In Daniel Leivant, editor, *Logic and Computational Complexity*, volume 960 of *Lecture Notes in Computer Science*, pages 367–392. Springer Berlin Heidelberg, 1995.

[27] Mark V. Lawson. *Finite Automata.* CRC Press LLC, 2004.

[28] Graham P. Oliver and Richard M. Thomas. Automatic presentations for finitely generated groups. In Volker Diekert and Bruno Durand, editors, *STACS 2005*, volume 3404 of *Lecture Notes in Computer Science*, pages 693–704. Springer Berlin Heidelberg, 2005.

[29] E. F. Robertson, N. Ruškuc, and J. Wiegold. Generators and relations of direct products of semigroups. *Transactions of the American Mathematical Society*, 350:2665–2685, 1998.

[30] Pedro V. Silva and Benjamin Steinberg. A geometric characterisation of automatic monoids. *The Quarterly Journal of Mathematics*, 55:333–356, 2004.

[31] Michael Sipser. *Introduction to the Theory of Computation.* Thomson Course Technology, second edition edition, 2006.