Formal Languages and the Word Problem in Groups

Thesis submitted for the degree of
Doctor of Philosophy
at the University of Leicester

Duncan W. Parkes

Department of Mathematics
and Computer Science
University of Leicester
LE1 7RH

September 2000

Contents

Acknowledgements

I am indebted to my supervisor Rick Thomas for all his help and support, and for countless suggestions and improvements to every part of this thesis. I would also like to thank Volodya Shavrukov, for many helpful discussions and suggestions, particularly in relation to Chapter 7. I am grateful to my examiners, John Fountain and Robert Marsh, for suggesting corrections and improvements, and to Friedrich Otto for his suggestions relating to [23], an earlier version of some of the results presented here. Finally, I would like to thank my family and friends, and, in particular, my wife Ruth, for all their support and encouragement.

I am grateful to the University of Leicester for funding this work.

Formal Languages and the Word Problem in Groups

Duncan W. Parkes

Abstract

We consider some interactions between the theory of groups and the theory of formal languages.

For any group G and generating set X we shall be primarily concerned with three sets of words over X: the word problem, the reduced word problem, and the irreducible word problem. We explain the relationships between these three sets of words and give necessary and sufficient conditions for a language to be the word problem (or the reduced word problem) of a group.

We prove that the groups which have context-free reduced word problem with respect to some finite monoid generating set are exactly the context-free groups, thus proving a conjecture of Haring-Smith. We also show that, if a group G has finite irreducible word problem with respect to a monoid generating set X, then the reduced word problem of G with respect to X is simple. In addition, we show that the reduced word problem is recursive (or recursively enumerable) precisely when the word problem is recursive.

The irreducible word problem corresponds to the set of words on the left hand side of a special rewriting system which is confluent on the equivalence class containing the identity. We show that the class of groups which have monoid presentations by means of finite special $[\lambda]$-confluent string-rewriting systems strictly contains the class of plain groups (the groups which are free products of a finitely generated free group and finitely many finite groups), and that any group which has an infinite cyclic central subgroup can be presented by such a string-rewriting system if and only if it is the direct product of an infinite cyclic group and a finite cyclic group.

Chapter 1

Introduction

Given a group G with a generating set X, we define the word problem of G with respect to X to be the set of words in X^* which are equal to the identity in G. This definition gives a natural link between group theory and the theory of formal languages, and presents us with the question of what relationship exists between the complexity of the word problem as a formal language and the algebraic structure of the group.

With any family of languages $\mathcal{F}$, we can associate the class of groups which have some finite generating set with respect to which the word problem is in $\mathcal{F}$: an interesting task is to try to find algebraic descriptions of these classes of groups. While, in general, the fact that the word problem lies in $\mathcal{F}$ may depend on our choice of finite generating set, it is well known that this is not the case if $\mathcal{F}$ is closed under inverse homomorphism: in this case, if the word problem with respect to any finite generating set lies in $\mathcal{F}$, then the word problem with respect to every finite generating set lies in $\mathcal{F}$ (see [14] for example).

There have been several such characterizations of the groups having word problem in a particular class of languages. One early characterization, by Anisimov [1], is that the class of groups which have regular word problem is the class of finite groups. In [21] the class of accessible groups with context-free word problem was shown to coincide with the virtually free groups (accessibility is a technical condition which we shall not define, and which can be removed thanks to the result of [8], that all finitely presented groups are accessible). In fact, if the word problem of a group is context-free, then it is deterministic context-free [22].

One further family of languages that has been considered is the one-counter languages. Herbst shows in [12] that a group has one-counter word problem if and only if it is a finite extension of a cyclic group (or, to put this another way, a group has a one-counter word problem if and only if it is either finite or is a finite extension of an infinite cyclic group). As in the case of the context-free languages, if the word problem of a group is a one-counter language, then it is necessarily a deterministic one-counter language. Herbst also shows that the class of groups with word problem in $\mathcal{C}$, where $\mathcal{C}$ is a cone in between the regular languages and the context-free languages, must be either the finite or the one-counter groups. While there are, of course, interesting families of languages which are not cones, these results do suggest that the one-counter groups are also of special interest.

In [10] Haring-Smith defines the reduced word problem of a group to be the subset of the word problem consisting of those non-empty words which have no non-empty proper prefix equal to the identity, and proves that a group has simple reduced word problem, with respect to some generating set which contains inverses, if and only if it is a free product of finitely many finite groups and a finitely generated free group (where the free factor may be trivial); he calls this class of groups the *plain* groups. In the same paper he made the following suggestion as to what happens when the reduced word problem is strict deterministic:

Conjecture 1.0.1 ([10]) *A finitely generated group G has a presentation whose reduced word problem is a strict deterministic language if and only if G is a finite extension of a plain group.*

Another subset of the word problem introduced in [10] is the irreducible word problem, the set of all non-empty words w in the word problem such that no non-empty proper subword of w represents the identity. The irreducible word problem corresponds to the set of words on the left hand side of a special rewriting system which is confluent on the equivalence class containing the identity, and we shall use this to cast some light on an intriguing question raised by Madlener and Otto in [20] about the groups with word problem describable by such rewriting systems.

In Chapter 2 we give some definitions from formal language theory, and fix notation: in general, formal language definitions may be found in [11] or [15]. In the first few sections of Chapter 3 we give some definitions from combinatorial group theory. We then go on to describe in more detail some of the ways in which groups have been classified using formal languages and string-rewriting systems.

We start Chapter 4 by spelling out the relationships between the word problem and the reduced word problem, and then between the word problem and the irreducible word problem. In Sections 4.2 and 4.3 we give sets of conditions which characterize when a language is respectively the word problem, and the reduced word problem, of a group. In the last section of this chapter we look at how the solvability of the word problem and the reduced word problem of a group are related.

Chapter 5 is devoted to syntactic monoids. We start off by pointing out that there is no hope of a characterization of any class of languages which is closed under inverse homomorphism and which strictly contains the regular languages by means of syntactic monoids alone, even amongst those languages whose syntactic monoids are groups. In Section 5.2 we go on to consider word problems of syntactic monoids.

In Chapter 6 we consider the reduced word problem in more detail. Firstly, in Section 6.1 we prove the conjecture of Haring-Smith that a group has strict deterministic reduced word problem with respect to some generating set if and only if it is a finite extension of a plain group, by proving that both classes coincide with the context-free groups (Theorem 6.1.1). In the second section we consider simple reduced word problems with respect to monoid generating sets, and prove that if a group has finite irreducible word problem with respect to some monoid generating set then it has simple reduced word problem with respect to that generating set (Theorem 6.2.2), a generalization of one direction of Theorem 3.7.1.

We then move on to the irreducible word problem. In Chapter 7 we consider groups which have finite irreducible word problem with respect to some finite monoid generating set. In the first section we prove the observation (implicit in [20]) that a group may be presented by a finite special $[\lambda]$-confluent string-rewriting system if and only if it has finite irreducible word problem with respect to some monoid generating set. In Section 7.2 we exhibit a monoid generating set for the group $C_\infty \times C_n$ with respect to which it has finite irreducible word problem: thus the class of groups which may be presented by finite special $[\lambda]$-confluent string-rewriting systems strictly contains the class of plain groups. In Section 7.3 we show that the groups of the form $C_\infty \times C_n$ are the only ones which which have such a presentation and which have an infinite cyclic central subgroup. In the last section of this chapter we give some further examples of groups which may be presented by finite special $[\lambda]$-confluent string-rewriting systems.

Finally, in Chapter 8, we consider groups which have infinite irreducible word problem for any generating set. In the first section we note that, if the irreducible word problem of a group with respect to some generating set is regular, then it must be finite (Proposition 8.1.1). In the last two sections we consider groups which have context-free and one-counter irreducible word problem with respect to some generating set.

Chapter 2

Formal Languages

2.1 Languages

Let Σ be a finite set or *alphabet*. The set of all finite words (or strings) over Σ (including the empty word λ) is denoted Σ^*. The subsets of Σ^* are known as *languages* over Σ.

If L is a language then L^*, the *Kleene closure* of L, is the submonoid of Σ^* generated by L, i.e. the set all words which can be made by concatenating together a finite (possibly empty) sequence of words from L. Similarly, L^+ is the subsemigroup generated by L, i.e. the set all words which can be made by concatenating together a finite non-empty sequence of words from L.

The set $\{w_1w_2 : w_1 \in L_1, w_2 \in L_2\}$ will be denoted L_1L_2; if L_1 or L_2 is a singleton, we may omit the braces and write, for example, L_1x instead of $L_1\{x\}$. We shall denote the complement in Σ^* of the language L by L^c.

If v and w are words over an alphabet Σ then we shall use the expression $v \equiv w$ to mean that v and w are identical as strings of symbols. If $w \equiv uv$, where $u, v \in \Sigma^*$, then u is said to be a *prefix* of w, and v is said to be a *suffix* of w.

Let L be a language. The *prefix closure* of L is the set of all prefixes of words in L. The language L is said to be *prefix-closed* if is is equal to its prefix closure. The terms *suffix closure* and *suffix-closed* are defined similarly.

Given a language L, the set of words in L which have no non-empty proper prefix in L is denoted $\textsc{min}(L)$. A language L is said to be *prefix-free* if $L = \textsc{min}(L)$.

We shall write $|w|$ for the length of the word w, and $|w|_x$ for the number of occurrences in w of the symbol x.

2.2 Regular Expressions

The *regular expressions* R over an alphabet Σ, and the languages $L(R)$ they denote, are defined recursively as follows:

- $\emptyset$ is a regular expression with $L(\emptyset) = \emptyset$;
- λ is a regular expression with $L(\lambda) = \{\lambda\}$;
- For each $a \in \Sigma$, there is a regular expression a with $L(a) = \{a\}$;
- If R is a regular expression then R^* is a regular expression denoting $L(R)^*$;

- If R_1 and R_2 are regular expressions then $(R_1 \cup R_2)$ is a regular expression denoting $L(R_1) \cup L(R_2)$ and $(R_1 R_2)$ is a regular expression denoting $L(R_1)L(R_2)$.

A language is said to be *regular* if it is denoted by some regular expression. In addition, we use R^+ to stand for RR^*, so that $L(R^+) = L(R)^+$. The class of regular languages will be denoted $\mathcal{R}$.

2.3 Finite Automata

An alternative characterization of the regular languages is provided by finite automata.

A *(non-deterministic) finite automaton* M is a quintuple $(Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of *states*, Σ is another finite set (the *input alphabet*), the *transition relation* δ is a subset of $Q \times (\Sigma \cup \{\lambda\}) \times Q$, the *start state* q_0 is a special element of Q, and the set F of *accept states* is a subset of Q.

The transition relation δ may be extended inductively to a subset δ^* of $Q \times \Sigma^* \times Q$ in the following way.

- Let (q, λ, q) be in δ^* for each $q \in Q$.
- If $(q_1, x, q_2) \in \delta$ then let (q_1, x, q_2) be in δ^*.
- If $(q_1, w, q_2) \in \delta^*$ and $(q_2, x, q_3) \in \delta$ then let $(q_1, wx, q_3) \in \delta^*$.

We say that M *accepts* a word $w \in \Sigma^*$ if $(q_0, w, q_f) \in \delta^*$ for some $q_f \in F$.

The set of words from Σ^* which are accepted by M is denoted $L(M)$, and is known as the *language accepted by M*.

A finite automaton is said to be *deterministic* if for each pair $(q, x) \in Q \times X$ there is at most one state $q' \in Q$ such that either $(q, x, q') \in \delta$ or $(q, \lambda, q') \in \delta$. In fact, any language which can be accepted by a non-deterministic finite automaton can be accepted by a deterministic finite automaton:

Theorem 2.3.1 *If $L \subseteq \Sigma^*$ then the following are equivalent:*

- *L can be accepted by a non-deterministic finite automaton;*
- *L can be accepted by a deterministic finite automaton; and*
- *L is denoted by some regular expression.*

The following lemma is a useful tool for showing that a language is not regular.

Lemma 2.3.2 (The Pumping Lemma for Regular Languages) *Let L be a regular language over an alphabet Σ, and let $w \in L$. Then there exists a constant N such that if $|w| > N$ then $w \equiv u_1 v u_2$ with $|u_1 v| < N$, $v \not\equiv \lambda$ and $u_1 v^i u_2 \in L$ for all $i \geqslant 0$.*

2.4 Pushdown Automata

We now extend the concept of a finite automaton by adding a stack: the resulting machine is known as a pushdown automaton.

A *pushdown automaton* (PDA) M is a septuple $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, where Q is a finite set of *states*, Σ is a finite set called the *input alphabet*, Γ is another finite set called the *stack alphabet*, the *transition relation* δ is finite subset of $Q \times (\Sigma \cup \{\lambda\}) \times \Gamma \times Q \times \Gamma^*$, the *start state* q_0 is a special element of Q, the *start symbol* Z_0 is a special element of Γ, and $F \subseteq Q$ is the set of *accept states*.

We call the set $Q \times \Sigma^* \times \Gamma^*$ the set of *configurations* of M, and write $(q_1, xw, Z\gamma) \rightsquigarrow (q_2, w, \beta\gamma)$ if $(q_1, x, Z, q_2, \beta) \in \delta$. We write $\overset{*}{\rightsquigarrow}$ for the reflexive transitive closure of $\rightsquigarrow$. If $(q_0, w, Z_0) \overset{*}{\rightsquigarrow} (q_f, \lambda, \beta)$ for some accept state q_f and some β, then we say that M *accepts* w. The set of words accepted by M is denoted $L(M)$, and a language is said to be *context-free* if it is accepted by some pushdown automaton. We shall denote the class of context-free languages $\mathcal{CF}$.

The method of acceptance described here is known as *acceptance by final state*. It is also possible to define *acceptance by empty stack*, where M is said to accept a word $w \in \Sigma^*$ if $(q_0, w, Z_0) \overset{*}{\rightsquigarrow} (q, \lambda, \lambda)$ for some $q \in Q$ (there is no need here for the set of accept states). The classes of languages which are accepted by pushdown automata by final state and by empty stack are the same (see, for example, [15]).

The following result is a version of Ogden's Lemma, which, in the same manner as the Pumping Lemma in the case of regular languages, can be used to show that certain languages are not context-free.

Lemma 2.4.1 *Let $L \subseteq \Sigma^*$ be a context-free language. Then there exists a* *constant N such that if $w \in L$, and N or more letters in w are marked, then $w \equiv u_1v_1u_2v_2u_3$ with v_1v_2 containing at least one marked position, $v_1u_2v_2$ containing at most N marked positions, and $u_1{v_1}^iu_2{v_2}^iu_3 \in L$ for all $i \geqslant 0$.*

If M is a pushdown automaton such that, for any configuration, there is at most one possible move (in particular, δ must be a partial function from $Q \times (\Sigma \cup \{\lambda\}) \times \Gamma$ to $Q \times \Gamma^*$), then M is said to be a *deterministic* pushdown automaton (DPDA), and $L(M)$ is then said to be a *deterministic context-free* language. The class of deterministic context-free languages will be denoted $\mathcal{DCF}$.

For deterministic pushdown automata, acceptance by final state and acceptance by empty stack are no longer equivalent. The languages which are accepted by deterministic pushdown automata by empty stack are exactly those which are accepted by final state and which are prefix-free. These languages are known as the *strict deterministic* languages.

2.5 One-Counter Automata

Let M be a pushdown automaton. The start symbol Z_0 is said to be a *bottom marker* for the stack if it appears once at the very bottom of the stack, and nowhere else. In fact, Z_0 is a bottom marker if δ has the following properties:

$$(q, x, Z_0, r, \gamma) \in \delta \Rightarrow \gamma \in (\Gamma \setminus \{Z_0\})^* Z_0;$$

and,

$$Z \neq Z_0 \text{ and } (q, x, Z, r, \gamma) \in \delta \Rightarrow \gamma \in (\Gamma \setminus \{Z_0\})^*.$$

A pushdown automaton (accepting by final state) is said to be *one-counter* if the stack alphabet contains a bottom marker and just one other symbol. The languages accepted by such automata are known as the *one-counter* languages (proofs of properties of the one-counter languages may be found in [6]). The *deterministic one-counter* languages are those languages accepted by one-counter automata which are deterministic. We shall denote the class of one-counter languages $\mathcal{OC}$, and the class of deterministic one-counter languages $\mathcal{DOC}$.

2.6 Grammars

Another characterization of the regular languages and the context-free languages is provided by grammars.

An *(unrestricted) grammar* $\mathcal{G} = (V, \Sigma, P, S)$ consists of a finite set V of *non-terminal* symbols, a finite set Σ of *terminal* symbols, a finite set $P \subset ((V \cup \Sigma)^* \setminus \Sigma^*) \times (V \cup \Sigma)^*$ of *productions*, and a special symbol $S \in V$ called the *start symbol*, where the sets V and Σ are required to be disjoint.

Let $\beta_1, \beta_2 \in (V \cup \Sigma)^*$. We write $\beta_1\alpha_1\beta_2 \rightsquigarrow \beta_1\alpha_2\beta_2$ if $\alpha_1 \to \alpha_2$ is in P, and then extend $\rightsquigarrow$ by reflexive transitive closure to $\overset{*}{\rightsquigarrow}$. The language $L(\mathcal{G})$ generated by $\mathcal{G}$ is $\{w \in \Sigma^* : S \overset{*}{\rightsquigarrow} w\}$. A language is *recursively enumerable* if and only if it can be generated by an unrestricted grammar. We shall denote the class of recursively enumerable languages $\mathcal{RE}$. A language is *recursive* if and only if it is recursively enumerable and its complement is recursively enumerable. (These definitions are equivalent to other machine-based definitions.)

The grammar $\mathcal{G}$ is said to be *left-linear* if every rule is of the form $A \to Bw$ or $A \to w$, where A and B are non-terminals and w is a string of terminals. The languages which can be generated by such grammars are exactly the regular languages.

If the elements of P are of the form $A \to \alpha$, where A is a non-terminal, and α is a (possibly empty) string of terminals and non-terminals, then the grammar $\mathcal{G}$ is said to be *context-free.* A language is context-free if and only if it can be generated by a context-free grammar.

A context-free grammar in said to be in *Greibach normal form* if every production is of the form $A \to a\alpha$, where A is a non-terminal, a is a terminal, and α is a (possibly empty) string of non-terminals, or is the production $S \to \lambda$. Every context-free language can be generated by a grammar in Greibach normal form.

A grammar in Greibach normal form is said to be *simple* if, whenever $A \to a\alpha$ and $A \to a\beta$ are in P, then we must have that $\alpha \equiv \beta$, and, if $S \to \lambda$ is in P, then it is the only production. A language which is generated by a simple grammar is said to be *simple.* The simple languages are exactly those which are accepted by empty stack by deterministic pushdown automata with only one state, and are thus a subclass of the strict deterministic languages (the deterministic context-free languages which are prefix-free).

Lastly, a grammar is said to be *context-sensitive* if, for any rule

$\alpha \to \beta$ in P, we must have $|\beta| \geqslant |\alpha|$; the languages which can be generated by such grammars are said to be *context-sensitive*. The context-sensitive languages will be denoted $\mathcal{CS}$.

2.7 Closure Properties

Let F be a family of languages. Then $\mathcal{F}$ is said to be *closed under homomorphism* if

$$L \subseteq \Sigma^*, L \in \mathcal{F}, \phi : \Sigma^* \to \mathrm{T}^* \text{ a monoid homomorphism} \Rightarrow L\phi \in \mathcal{F}.$$

Similarly, $\mathcal{F}$ is said to be *closed under inverse homomorphism* if

$$L \subseteq \mathrm{T}^*, L \in \mathcal{F}, \phi : \Sigma^* \to \mathrm{T}^* \text{ a monoid homomorphism} \Rightarrow L\phi^{-1} \in \mathcal{F}.$$

We say that $\mathcal{F}$ is *closed under intersection with regular languages* if

$$L, L' \subseteq \Sigma^*, L \in \mathcal{F}, L' \in \mathcal{R} \Rightarrow L \cap L' \in \mathcal{F}.$$

A family of languages which is closed under homomorphism, inverse homomorphism and intersection with regular languages is known as a *cone* (in [15], a cone is known as a *full trio*).

Theorem 2.7.1 (See [6], for example) *The classes of regular languages, context-free languages, one-counter languages, and recursively enumerable languages are all cones.*

We say that $\mathcal{F}$ is *closed under union* if

$$L, L' \subseteq \Sigma^*, L, L' \in \mathcal{F} \Rightarrow L \cup L' \in \mathcal{F},$$

and that $\mathcal{F}$ is *closed under Kleene star* if

$$L \in \mathcal{F} \Rightarrow L^* \in \mathcal{F}.$$

A family of languages $\mathcal{F}$ is said to be *closed under the operator* MIN if

$$L \in \mathcal{F} \Rightarrow \textsc{min}(L) \in \mathcal{F}.$$

We shall make use of the following result in our proof of Theorem 6.1.1 (see [15] for a proof):

Lemma 2.7.2 (See [6], for example) *The deterministic context-free languages and the deterministic one-counter languages are closed under the operator* MIN.

We contrast this result with the situation for the context-free languages:

Example 2.7.3 *There is a context-free language L such that* MIN(L) *is not context-free.*

Proof. Let $\Sigma = \{a, A, b, B\}$ and let L be the language

$$\{\alpha \in \Sigma^* : |\alpha|_a = |\alpha|_A \text{ or } |\alpha|_b = |\alpha|_B\}.$$

Then L is certainly context-free.

Let $P = \textsc{min}(L)$ and let J be the regular language denoted by the regular expression $a^+b^+A^+B^+$. If P were context-free, then $K = P \cap J$ would be context-free, by Theorem 2.7.1. Now

$$a^ib^jA^kB^l \in K \Leftrightarrow j = l > 0 \text{ and } i > k > 0,$$

so that $K = \{a^{u+k}b^jA^kB^j : u, j, k > 0\}$.

We now use Ogden's lemma (Lemma 2.4.1) to prove that K is not context-free. If K were context-free, then there would exist a constant N such that, if we pick any word w in K and mark N or more positions in w, then we can write w as $u_1v_1u_2v_2u_3$ in such a way that v_1v_2 contains at least one marked position, $v_1u_2v_2$ contains at most N marked positions, and $u_1{v_1}^r u_2{v_2}^r u_3 \in K$ for all $r \geqslant 0$.

Consider the word $w \equiv a^{N+1}b^NA^NB^N$ in K with all the instances of b and A marked. With u_1, v_1, u_2, v_2 and u_3 as above, let $w_r \equiv u_1{v_1}^r u_2{v_2}^r u_3$ for $r \geqslant 0$. In order that the w_r are in K we must have that each of v_1 and v_2 consists of a repetition of a single letter (i.e. each is of the form a^i, b^i, A^i or B^i). If v_2 is of the form B^i, then v_1 must be b^i (else $|w_2|_b \neq |w_2|_B$) and then $v_1u_2v_2$ contains more than N marked positions. If v_2 is of the form A^i, then v_1 must be a^i (else $|w_r|_a \leqslant |w_r|_A$ for some $r > 1$), and $v_1u_2v_2$ contains more than N marked positions. If v_2 is of the form b^i, we do not have $|w_2|_b = |w_2|_B$, and, if v_2 is of the form a^i, then v_1v_2 does not contain a marked position. In all cases we have a contradiction to the conclusion of Ogden's lemma, and therefore K cannot be context-free. $\square$

A *generalized sequential machine* (GSM for short) is defined to be a sextuple $M = (Q, \Sigma, \Delta, \delta, q_0, F)$ where Q is a finite set of *states*, Σ is the *input alphabet*, Δ is the *output alphabet*, δ is a function from $Q \times (\Sigma \cup \{\lambda\})$ to the set of finite subsets of $Q \times \Delta^*$, q_0 is the *start state*, and F is the set of *accept states*.

As for finite automata, δ may be extended to a function δ^* with domain $Q \times \Sigma^*$ by inductively defining $\delta^*(q, \lambda)$ to be $\{(q, \lambda)\}$ and, for $u \in \Sigma^*$ and $a \in \Sigma$, defining $\delta^*(q, ua)$ to be the set of all (p, vw) such that, for some p', we have $(p', v) \in \delta^*(q, u)$ and $(p, w) \in \delta^*(p', a)$.

For $u \in \Sigma^*$, let $M(u)$ be the set of words $w \in \Delta^*$ such that $(f, w) \in \delta^*(q_0, u)$ for some accept state f, and let $M(L) = \bigcup\{M(u) : u \in L\}$. We now have a mapping from the set of languages over Σ to the set of languages over Δ; this is the *GSM-mapping* defined by M.

Proposition 2.7.4 *Any class of languages which is a cone is closed under GSM-mappings.*

There is a useful table showing which classes of languages are closed under what operations at [15, pages 280–281].

2.8 Insertions and Deletions

Let L be a language over an alphabet Σ. Then $\textsc{ins}(L)$ is defined to be the set of words which, when inserted at any point into a word from L, result in another word from L, i.e.

$$\text{INS}(L) = \{w \in \Sigma^* : uv \in L \Rightarrow uwv \in L\}.$$

Let $\text{SUB}(L)$ be the set of subwords of L, so that

$$\text{SUB}(L) = \{w \in \Sigma^* : uwv \in L \text{ for some } u, v \in \Sigma^*\}.$$

Then $\text{DEL}(L)$ is defined to be the set of words from $\text{SUB}(L)$ which, when deleted from any word in L, always result in another word from L, i.e.

$$\text{DEL}(L) = \{w \in \text{SUB}(L) : uwv \in L \Rightarrow uv \in L\}.$$

The subsets $\text{INS}(L)$ and $\text{DEL}(L)$ are defined and studied in [17].

For two languages L_1 and L_2 over the alphabet Σ, the *dipolar deletion* $L_1 \rightleftharpoons L_2$ is defined by the equation:

$$L_1 \rightleftharpoons L_2 = \{x \in \Sigma^* : u \equiv \alpha x \beta, v \equiv \alpha\beta, u \in L_1, v \in L_2\}.$$

The following result from [17] gives the relationships between $\text{INS}(L)$, $\text{DEL}(L)$ and $L_1 \rightleftharpoons L_2$.

Proposition 2.8.1 *Let L be a language over Σ. Then*

i. $\text{INS}(L) = (L^c \rightleftharpoons L)^c$, *and*

ii. $\text{DEL}(L) = (L \rightleftharpoons L^c)^c \cap \text{SUB}(L)$.

If $L \subseteq \Sigma^*$ is a language, then L is said to be *insertion closed* if, whenever u, v and w are words in Σ^* such that $v \in L$ and $uw \in L$, then $uvw \in L$.

We will need the following result from [17]:

Proposition 2.8.2 *Let Σ be a finite alphabet and $L \subseteq \Sigma^*$. Then:*

i. there is a unique smallest insertion closed language K in Σ^* containing L;

ii. if L is context free, then K is context-free.

The language K in Proposition 2.8.2 is known as the *insertion closure* of L in Σ^*. Note that it is not true that the insertion closure of a regular language must be regular.

Let $L \subseteq \Sigma^*$ be an insertion closed language. Then the *insertion base* of L is the set of non-trivial words from L which are not of the form uvw, with uw and v non-trivial elements of L.

2.9 Syntactic Monoids

The *syntactic congruence* $\sim_L$ of a language $L \subseteq \Sigma^*$ is the coarsest congruence on Σ^* such that L is a union of congruence classes. We shall denote the congruence class of a word w under the syntactic congruence by $[w]$. Proofs of properties of the syntactic congruence (and syntactic monoids) may be found in [16].

The following is an alternative characterization of the syntactic congruence.

Proposition 2.9.1 *Let L be a language over Σ. The syntactic congru-*

ence $\sim_L$ is given by

$$(w_1 \sim_L w_2) \Leftrightarrow \forall u, v \in \Sigma^*(uw_1v \in L \Leftrightarrow uw_2v \in L).$$

The *syntactic monoid* M_L of L is the quotient of the free monoid Σ^* by $\sim_L$, and the *syntactic morphism* η_L is the canonical homomorphism from Σ^* onto M_L, i.e. $\eta_L : w \mapsto [w]$. A monoid is said to be *syntactic* if it is the syntactic monoid of some language.

Since any congruence on Σ^* which has L as a union of congruence classes must also have L^c as a union of congruence classes, the following proposition is clear.

Proposition 2.9.2 *The syntactic monoid of a language $L \subseteq \Sigma^*$ is equal to the syntactic monoid of its complement L^c.*

The syntactic monoid M_L is, in a sense, the smallest monoid M onto which there is a homomorphism such that the images of L and L^c are disjoint. The following definition allows one to formalize the sense in which it is the smallest.

Definition 2.9.3 *Let M_1 and M_2 be monoids. Then M_1 is said to* divide *M_2 if M_1 is a homomorphic image of some submonoid of M_2.*

We then have the following:

Proposition 2.9.4 *Let $L \subseteq \Sigma^*$ be a language, let M be a monoid and let $\phi : \Sigma^* \to M$ be such that $L = A\phi^{-1}$ for some $A \subseteq M$. Then M_L divides M.*

It is useful to have a word to describe the situation where a subset of a monoid is the image of some language which has that monoid as its syntactic monoid.

Definition 2.9.5 *Let M be a monoid. A subset A of M is said to be* disjunctive*, or* syntactic*, if there is no nontrivial congruence on M such that A is a union of congruence classes.*

In particular, the image of a language in its syntactic monoid is disjunctive. In fact, if you have a language $L \subseteq \Sigma^*$, a monoid M, and a surjective homomorphism $\phi : \Sigma^* \to M$ such that $L = A\phi^{-1}$ for some disjunctive subset A of M then M is isomorphic to the syntactic monoid of L.

We shall need one further result, a proof of which may be found in, for example, [16].

Theorem 2.9.6 *A language is regular if and only if it has finite syntactic monoid.*

Quite a lot of work has been done to classify subclasses of the regular languages by means of syntactic monoids, but we shall not look at this here. A survey of what is known in this area can be found in [25].

2.10 String-Rewriting Systems

Given a finite alphabet Σ, a *string-rewriting system* R over Σ is a set of rules $u \to v$, where $u, v \in \Sigma^*$; we shall only be interested in finite string-rewriting systems here. The *domain* $\mathrm{dom}(R)$ of R is the set of all $u \in \Sigma^*$

such that there exists a rule $u \to v$ in R for some $v \in \Sigma^*$; similarly, the *range* $\mathrm{ran}(R)$ of R is the set of all $v \in \Sigma^*$ such that there exists a rule $u \to v$ in R for some $u \in \Sigma^*$.

We define the *reduction relation* $\overset{*}{\Rightarrow}_R$ to be the reflexive transitive closure of $\Rightarrow_R$, where $w_1uw_2 \Rightarrow_R w_1vw_2$ if $w_1, w_2 \in \Sigma^*$ and $u \to v \in R$. We shall say u *reduces* to v under R if $u \overset{*}{\Rightarrow}_R v$. The transitive symmetric closure of $\overset{*}{\Rightarrow}_R$ will be written as $\overset{*}{\Leftrightarrow}_R$, and is called the *Thue congruence* of R. The congruence class of a word w is then denoted by $[w]$.

A rule $u \to v$ is said to be *length-reducing* if $|u| > |v|$, and a string-rewriting system R is said to be *length-reducing* if every rule in R is length-reducing. A length-reducing string-rewriting system R is said to be *monadic* if $\mathrm{ran}(R) \subseteq \Sigma \cup \{\lambda\}$, and *special* if $\mathrm{ran}(R) = \{\lambda\}$.

A string-rewriting system R over Σ is said to be *confluent* ($[\lambda]$-*confluent*) if, whenever $u, w_1, w_2 \in \Sigma^*$ $(u, w_1, w_2 \in [\lambda])$ with $u \overset{*}{\Rightarrow}_R w_1$ and $u \overset{*}{\Rightarrow}_R w_2$, then there exists $v \in \Sigma^*$ $(v \in [\lambda])$ such that $w_1 \overset{*}{\Rightarrow}_R v$ and $w_2 \overset{*}{\Rightarrow}_R v$.

Chapter 3

Classifications of Groups

3.1 Generating Sets

A set X, where each $x \in X$ represents an element of a group G, is said to be a *monoid generating set* for G if every element of G is represented by a word from X^*. Let X^{-1} be a new set of symbols $\{x^{-1} : x \in X\}$, where x^{-1} represents the inverse of x. Then X is said to be a *group generating set* for G if $X \cup X^{-1}$ is a monoid generating set for G.

3.2 Presentations

A *group presentation* $\langle X : R \rangle$ for a group G consists of a group generating set X for G, and a set R of words over $X \cup X^{-1}$. If we let Y denote the set $X \cup X^{-1}$ then every element of G can be written as a word in Y^* and G is isomorphic to $Y^*/\approx$, where $\approx$ is the congruence on Y^* generated by all pairs of the form (w, λ) with $w \in R$, together with all pairs of the form (xx^{-1}, λ) and $(x^{-1}x, \lambda)$ with $x \in X$.

The *free group* with generating set X is the group with presentation $\langle X : \ \rangle$ (the trivial group can be thought of as the free group generated by the empty set). A *virtually free* group is a finite extension of a free group, i.e. a group with a free subgroup of finite index.

Let G_1 and G_2 be groups with presentations $\langle X_1 : R_1 \rangle$ and $\langle X_2 : R_2 \rangle$ (where X_1 and X_2 are disjoint). Then the *free product* $G_1 * G_2$ is the group with presentation $\langle X_1 \cup X_2 : R_1 \cup R_2 \rangle$.

Given groups $G = \langle X : R \rangle$ and $H = \langle Y : S \rangle$, where X and Y are disjoint sets, subgroups $A = \langle \{u_i : i \in I\} \rangle \leqslant G$ and $B = \langle \{v_i : i \in I\} \rangle \leqslant H$, where each u_i is an element of X^* and each v_i is an element of Y^*, and an isomorphism $\phi : A \to B$ such that $u_i\phi = v_i$ for $i \in I$, *the free product of G and H with A and B amalgamated* is the group with presentation $\langle X \cup Y : R \cup S \cup \{u_i {v_i}^{-1} : i \in I\} \rangle$. We shall often identify A and B, and we then write $G *_A H$ for this group.

3.3 Cayley Graphs

When we refer to the *Cayley graph* of a group we shall mean the right Cayley graph, that is, for a group G with monoid generating set X, the graph Γ whose vertices are the elements of G, and which has an edge labelled by

$x \in X$ from vertex g_1 to vertex g_2 exactly when $g_1 x = g_2$. For each vertex g of Γ, any word $w \in X^*$ labels a path starting at g. The path labelled by the word w (or sometimes the word itself) is said to be *closed* if it starts and ends at the same vertex (closed paths are also known as *loops*) and *simple* if no non-empty proper subword of w labels a loop.

It is noted in [10] that a non-empty word w labels a simple loop in the Cayley graph of G with respect to X if and only if it is in the irreducible word problem of G with respect to X.

3.4 Word Problems

Given a monoid generating set X for a group G, the *word problem*, $W_X^m(G)$, of G with respect to X is the set of all words from X^* which are equal to the identity in G. The word problem of G with respect to a group generating set X, denoted $W_X^g(G)$, is $W_{X \cup X^{-1}}^m(G)$. The set of non-empty words from the word problem which have no non-empty proper prefix in the word problem is called the *reduced word problem* of G with respect to X, and is denoted by $R_X^m(G)$, if X is a monoid generating set, or by $R_X^g(G)$, if X is a group generating set. Lastly, the set of non-empty words from the word problem which have no non-empty proper subword which is in the word problem is called the *irreducible word problem* and is denoted $I_X^m(G)$, or $I_X^g(G)$, depending on whether we are considering X as a monoid or as a group generating set.

We shall sometimes talk about $W_X^m(M)$ where M is a monoid which is not a group. It should be noted that in this case knowing how to decide whether or not a word is in $W_X^m(M)$ does not give a solution to the full word problem for M.

Another way of thinking of the word problem is the following. Let X be a finite alphabet, and let $\phi_X : X^* \to G$ be a surjective homomorphism. Then X is a finite monoid generating set for G, and the word problem of G with respect to X is the kernel of ϕ_X.

3.5 $\mathcal{F}$-Subsets

Let $\mathcal{F}$ be a class of languages which is closed under inverse homomorphism, and let M be a finitely generated monoid. A subset A of M is said to be an *$\mathcal{F}$-subset* if for any alphabet X and surjective homomorphism $\phi : X^* \to M$, we have $A\phi^{-1} \in \mathcal{F}$. The set of $\mathcal{F}$-subsets of the monoid M will be denoted by $\mathcal{F}(M)$.

The independence of this concept with respect to generating set and surjective homomorphism is provided by the following result.

Lemma 3.5.1 ([26]) *Let M be a finitely generated monoid, Σ and T finite alphabets, $\phi : \Sigma^* \to M$ a homomorphism, $\psi : \mathrm{T}^* \to M$ a surjective homomorphism. Then there is a homomorphism $\chi : \Sigma^* \to \mathrm{T}^*$ such that $\chi\psi = \phi$.*

Let A be an $\mathcal{F}$-subset of a monoid M, so that there is a monoid generating set X and a surjective homomorphism $\phi : X^* \to M$, with $A\phi^{-1} \in \mathcal{F}$. If Y and $\psi : Y^* \to M$ are respectively another alphabet and surjective homomorphism then, by Lemma 3.5.1, there is a homomorphism χ such that $\chi\phi = \psi$. We then have $A\psi^{-1} = A\phi^{-1}\chi^{-1}$, so that $A\psi^{-1}$ is an inverse image of $A\phi^{-1}$, and thus $A\psi^{-1} \in \mathcal{F}$ by the closure of $\mathcal{F}$ under inverse homomorphism.

Herbst writes extensively on $\mathcal{F}$-subsets in [12] and [13]. We shall use the following results from these papers.

Theorem 3.5.2 ([13]) *Let G be a finitely generated group, and let A be a finite, non-empty, disjunctive subset of G such that $A \in \mathcal{RE}(G)$ (respectively, $A \in \mathcal{CS}(G)$). Then every finite subset of G is recursively enumerable (context-sensitive).*

Theorem 3.5.3 ([13]) *Let M be a finitely generated cancellative monoid, and let T be a finite non-empty context-free subset of M. Then every finite subset of M is context-free.*

If $\mathcal{F}$ is closed under inverse homomorphism and the word problem of G with respect to a finite monoid generating set X is in $\mathcal{F}$ then $\{1\}$ is an $\mathcal{F}$-subset of G. Thus $W^m_Y(G) \in \mathcal{F}$ for any finite monoid generating set Y. In other words:

Proposition 3.5.4 *Let X and Y be finite monoid generating sets for a group G, and let $\mathcal{F}$ be a class of languages which is closed under inverse homomorphisms. If $W^m_X(G) \in \mathcal{F}$ then $W^m_Y(G) \in \mathcal{F}$.*

In the light of this result, if $\mathcal{F}$ is a class of languages which is closed under inverse homomorphism, and the word problem of a group G with respect to some finite monoid generating set X lies in $\mathcal{F}$, then we may say that the word problem of G is in $\mathcal{F}$ ($W(G) \in \mathcal{F}$) without reference to any particular generating set, and that G is an $\mathcal{F}$-group.

3.6 The Word Problem

With any family of languages $\mathcal{F}$, which is closed under inverse homomorphism, we can associate the class of groups whose word problem is in $\mathcal{F}$: an interesting task is to try to find algebraic descriptions of these classes of groups.

There have been several such characterizations of the groups having word problem in a particular class of languages. One early such was by Anisimov in [1]

Theorem 3.6.1 ([1]) *Let G be a finitely generated group. Then $W(G)$ is regular if and only if G is finite.*

Anisimov also proved several closure properties for the class of groups which have context-free word problem, including Proposition 3.5.4 in the case of the context-free languages, but he was unable to give an algebraic description of this class of groups.

In [21] the class of groups which have context-free word problem and which are accessible (a technical condition which we shall not define) was shown to coincide with the virtually free groups (the accessibility condition here can be removed thanks to the result of [8], that all finitely presented groups are accessible). In fact, if the word problem of a group is context-free, then it is deterministic context-free (see [22]). Putting these results together gives the following theorem:

Theorem 3.6.2 ([21, 22]) *Let G be a finitely generated group. The following are equivalent:*

i. $W(G) \in \mathcal{CF}$ *(i.e. G is a context-free group);*

ii. $W(G) \in \mathcal{DCF}$*;*

iii. *G is virtually free.*

One further family of languages that has been considered is the class of one-counter languages. Herbst showed in [12] that a group has one-counter word problem if and only if it is a finite extension of a cyclic group (or, to put this another way, a group has a one-counter word problem if and only if it is either finite or is a finite extension of an infinite cyclic group). As in the case of the context-free languages, if the word problem of a group is a one-counter language, then it is necessarily a deterministic one-counter language.

Theorem 3.6.3 ([12]) *Let G be a finitely generated group. The following are equivalent:*

i. $W(G) \in \mathcal{OC}$ *(i.e. G is a one-counter group);*

ii. $W(G) \in \mathcal{DOC}$*;*

iii. *G is a finite extension of a cyclic group.*

Several more equivalent conditions are given in [12] and [14].

Herbst also shows that the class of groups with word problem in $\mathcal{C}$, where $\mathcal{C}$ is a cone in between the regular languages and the context-free languages, must be either the finite or the one-counter groups. While there are, of course, interesting families of languages which are not cones, these results do suggest that the one-counter groups are also of special interest.

3.7 The Reduced Word Problem

The concepts of reduced word problem and irreducible word problem were introduced by Haring-Smith in [10], where he proved the following results:

Theorem 3.7.1 *Let G be a group, and let X a finite group generating set for G. Then $R_X^g(G)$ is a simple language if and only if $I_X^g(G)$ is finite.*

Theorem 3.7.2 *Let G be a finitely generated group. The following are equivalent:*

i. There is a finite group generating set X for G such that $R_X^g(G)$ is a simple language;

ii. G is the free product of a finitely generated free group and finitely many finite groups.

On account of the previous theorem, Haring-Smith named the class of groups which are the free product of a finitely generated free group and finitely many finite groups the *plain* groups. Note that, by Theorem 3.7.1, condition (i) in Theorem 3.7.2 is equivalent to there being a finite group generating set X for G such that $I_X^g(G)$ is finite.

3.8 String-rewriting systems

In this section we shall give a short survey of some of the things that are known about the presentation of groups by finite string-rewriting systems. Much of this material is derived from [20] which gives a general account of the subject.

A string-rewriting system R over Σ is said to *present* the monoid

$$\langle \Sigma : \{u = v : u \to v \text{ in } R\}\rangle,$$

which is isomorphic to $\Sigma^* / \overset{*}{\Leftrightarrow}_R$. We are interested here in the case where the monoid presented by R is a group.

Several interesting results have been proved giving algebraic characterizations of the groups which can be presented by certain classes of finite string-rewriting system; we will only mention a few of them here. The first such result was proved by Cochet in [7]:

Theorem 3.8.1 *A finitely generated group G has a presentation by a finite special confluent string-rewriting system if and only if it is a free product of finitely many (finite or infinite) cyclic groups.*

A string-rewriting system R over a set X is said to *provide inverses of* *length one* for every letter if, for each $x \in X$, there exists $y \in X$ such that $xy \Rightarrow^*_R \lambda$ and $yx \Rightarrow^*_R \lambda$.

In [20] Madlener and Otto prove the following:

Theorem 3.8.2 *A finitely generated group can be presented by a finite special* $[\lambda]$*-confluent string-rewriting system which provides inverses of length one if and only if it is a plain group.*

But the question as to which groups may be presented by finite special string-rewriting systems which do not necessarily provide inverses of length one is left open:

Question 3.8.3 *Which groups may be presented by finite* $[\lambda]$*-confluent special string-rewriting systems?*

We shall return to this question in Chapter 7.

There is at present no known algebraic characterization of the class of groups which may be presented by finite monadic confluent string-rewriting systems. In [9], Gilman makes the following conjecture:

Conjecture 3.8.4 ([9]) *A finitely generated group can be presented by a finite monadic confluent string-rewriting system if and only if it is plain.*

This has been shown to be true for finite monadic confluent string-rewriting systems which provide inverses of length one [4], and for finite two-monadic confluent string-rewriting systems [5] (a monadic string-rewriting system R is said to be *two-monadic* if $\mathrm{dom}(R) \subseteq \Sigma^2$), but remains open in general.

The following result, from [19], provides an easy way to show that some groups can not be presented by finite monadic confluent string-rewriting systems.

Theorem 3.8.5 *Let G be a group which can be presented by a finite monadic confluent string-rewriting system. Then each finitely generated abelian subgroup of G is either finite or infinite cyclic.*

Autebert, Boasson and Sénizergues have shown in [3] that the groups presented by finite monadic $[\lambda]$-confluent string-rewriting systems which provide inverses of length one are exactly the context-free groups, and it is shown in [20] that this continues to be true if the condition on inverses is dropped. This provides yet another equivalent condition for Theorem 3.6.2:

Theorem 3.8.6 ([20]) *Let G be a finitely generated group. Then G has a presentation by a finite monadic $[\lambda]$-confluent string-rewriting system if and only if G is is a context-free group.*

Chapter 4

Characterizations

4.1 Relationships

The relationship between the reduced word problem and the word problem of a group is summed up in the following pair of results:

Proposition 4.1.1 *The word problem of a group with respect to a monoid generating set is the Kleene closure of its reduced word problem with respect to that generating set.*

Proof. Let G be a group, X a finite monoid generating set for G, $R = R_X^m(G)$ and $W = W_X^m(G)$. Let w be an element of R^*. Since w is a product of elements from R, and $R \subseteq W$, we have that w is in W. So $R^* \subseteq W$.

Conversely, suppose that W is not contained in R^*, and let w be a word of minimal length in $W \setminus R^*$. If $w \equiv \lambda$ then clearly $w \in R^*$; so we may assume that there is a non-empty prefix u of w which is in R. Then $w \equiv uv$

where $v \in W$ and $|v| < |w|$. By the minimality of w, $v \in R^*$, and hence $w \in R^*$, a contradiction. □

Proposition 4.1.2 *If W is the word problem of a group with respect to a monoid generating set X and R is the reduced word problem with respect to X, then $R = \textsc{min}(W) \cap X^+$.*

Proof. Since $R \subseteq W$ and R is prefix-free, we certainly have that R is a subset of $\textsc{min}(W)$; since, by definition, R does not contain the empty word, R is a subset of $\textsc{min}(W) \cap X^+$. On the other hand, if $\alpha \in \textsc{min}(W)$ and $\alpha \not\equiv \lambda$ then α is a non-empty word in W such that no proper prefix of α lies in W, so that $\alpha \in R$ by definition. □

The following two results give the relationship between the word problem and the irreducible word problem.

Proposition 4.1.3 *If G is a group and X is a monoid generating set for G, then $W_X^m(G)$ is the insertion closure of $I_X^m(G) \cup \{\lambda\}$ in X^*.*

Proof. Let $W = W_X^m(G)$ and $I = I_X^m(G)$. Let K be the insertion closure of $I \cup \{\lambda\}$ in X^*.

If u, v and w are words such that $v \in W$ and $uw \in W$, then v and uw represent the identity of G, and so uvw represents the identity of G, giving that $uvw \in W$. So W is an insertion closed language containing $I \cup \{\lambda\}$, and therefore $K \subseteq W$ by definition of K.

Suppose that W is not contained in K, and let w be a word of minimal length in $W \setminus K$. If $w \in I$ or $w = \lambda$, then $w \in K$ by definition; so we may assume that $w \equiv usv$ for some words u and v with $|uv| > 0$ and some non-empty word $s \in I$. Since $uv \in W$ and $|uv| < |w|$, we have that $uv \in K$. Since $s \in I \subseteq K$ and K is insertion closed, we have that $w \equiv usv \in K$, a contradiction. □

Proposition 4.1.4 *If G is a group and X is a monoid generating set for G, then $I_X^m(G)$ is the insertion base of $W_X^m(G)$ in X^*.*

Proof. By Proposition 4.1.3, the word problem is insertion closed. The result follows easily from the definitions. □

4.2 The Word Problem

We now give a pair of conditions which characterize when a language W is the word problem of a group:

Proposition 4.2.1 *Let $W \subseteq \Sigma^*$. Then W is the word problem of a group if and only if it satisfies the following conditions:*

(W1) *if $\alpha \in \Sigma^*$ then there exists $\beta \in \Sigma^*$ such that $\alpha\beta \in W$;*

(W2) *if $\alpha \in W$ and $u\alpha v \in W$ then $uv \in W$.*

Proof. We first show that, if W is the word problem of a group, then W satisfies (W1) and (W2).

Let $\alpha \in \Sigma^*$ and suppose that α represents $g \in G$. Let β be a word representing g^{-1}; then $\alpha\beta$ represents gg^{-1}, and so $\alpha\beta \in W$. Therefore W satisfies (W1).

If $\alpha \in W$ and $u\alpha v \in W$ then $\alpha = u\alpha v = 1$ in G, so that $uv = 1$ in G, and so $uv \in W$. Therefore W satisfies (W2).

Conversely, suppose that W satisfies (W1) and (W2); we want to show that W is the word problem of a group.

Suppose that $\alpha \in W$ and that $u\alpha v \notin W$. By (W1), there exists $\beta \in \Sigma^*$ with $u\alpha v\beta \in W$. Since $\alpha \in W$ and $u\alpha v\beta \in W$, we have that $uv\beta \in W$ by (W2). If $uv \in W$, then we have $uv \in W$ and $uv\beta \equiv \lambda uv\beta \in W$, so that $\lambda\beta \equiv \beta \in W$ by (W2). Then, since $\beta \in W$ and $u\alpha v\beta\lambda \in W$, we have $u\alpha v\lambda \equiv u\alpha v \in W$ by (W2), a contradiction. So we have deduced:

(W3) if $\alpha \in W$ and $u\alpha v \notin W$ then $uv \notin W$.

Now let $\sim$ denote the syntactic congruence of W, i.e. $\sim$ is the congruence on Σ^* defined by:

$$(\alpha_1 \sim \alpha_2) \Leftrightarrow (u\alpha_1 v \in W \Leftrightarrow u\alpha_2 v \in W \ \forall u, v \in \Sigma^*).$$

Let M be the syntactic monoid $\Sigma^*/\sim$, so that we have the natural homomorphism $\phi : \Sigma^* \to M$. Recall that by standard properties of syntactic monoids, W is a union of congruence classes of $\sim$. Since if $\alpha_1 \in W$ and $\alpha_2 \in X^*$ such that $\alpha_1 \sim \alpha_2$ (noting that W is non-empty by (W1)), then

$$\begin{aligned}(\alpha_1 \sim \alpha_2) &\Rightarrow (u\alpha_1 v \in W \Leftrightarrow u\alpha_2 v \in W)\\ &\Rightarrow (\lambda\alpha_1\lambda \equiv \alpha_1 \in W \Leftrightarrow \lambda\alpha_2\lambda \equiv \alpha_2 \in W)\end{aligned}$$

and we must have $\alpha_2 \in W$.

In addition, in our situation, we have:

$$\begin{aligned} \alpha_1, \alpha_2 \in W \quad &\Rightarrow \quad \left(\begin{array}{rcl} u\alpha_1 v \in W & \Leftrightarrow & uv \in W \\ & \Leftrightarrow & u\alpha_2 v \in W \end{array} \right) \quad \text{by (W2) and (W3)} \\ &\Rightarrow \quad \alpha_1 \sim \alpha_2. \end{aligned}$$

Thus W consists of a single conguence class of $\sim$.

We have shown that W is equal to a congruence class of $\sim$. By (W2), if we pick $\alpha \in W$, we have that $\lambda\alpha\lambda \in W$, so that $\lambda\lambda \equiv \lambda \in W$. Thus $W = \{\alpha \in \Sigma^* : \alpha \sim \lambda\}$ and then $W = 1\phi^{-1}$.

If $m \in M$, choose $\alpha \in \Sigma^*$ with $\alpha\phi = m$, and then (using (W1)) choose $\beta \in \Sigma^*$ such that $\alpha\beta \in W$. If $n = \beta\phi$, then $mn = (\alpha\phi)(\beta\phi) = (\alpha\beta)\phi = 1$. So every element of M has a right inverse, which is a sufficient condition for the monoid M to be a group. Since $W = 1\phi^{-1}$, we have that W is the word problem of the group M as required. $\square$

4.3 The Reduced Word Problem

We may now give a set of conditions which characterize when a language R is the reduced word problem of a group:

Proposition 4.3.1 *Let $R \subseteq \Sigma^*$. Then R is the reduced word problem of a group if and only if it satisfies the following conditions:*

(R1) *if $\alpha \in \Sigma^*$ then there exists $\beta \in \Sigma^*$ such that $\alpha\beta \in R^*$;*

(R2) *if* $\alpha \in R$ *and* $u\alpha v \in R^*$ *then* $uv \in R^*$*;*

(R3) $\lambda \notin R$*;*

(R4) *if* $\alpha \in R$ *then there is no* $\beta \in R$ *such that* $\alpha \equiv \beta\gamma$ *and* $\gamma \not\equiv \lambda$*.*

Proof. Suppose that R is the reduced word problem for a group G. By Proposition 4.1.1, $W = R^*$ is the word problem of G. Given this, the fact that R satisfies (R1) and (R2) follows immediately from Proposition 4.2.1. The properties (R3) and (R4) follow immediately from the definition of R.

We now want to show that a language which satisfies (R1)–(R4) must be the reduced word problem of a group. Assume that R satisfies (R1)–(R4).

If $\alpha \equiv \alpha_1 \ldots \alpha_n \in R^*$, where each α_i is in R, then (R2) gives

$$\begin{array}{ccccccccc} u\alpha v \in R^* & \Rightarrow & u\alpha_1 \ldots \alpha_n v \in R^* & \Rightarrow & u\alpha_2 \ldots \alpha_n v \in R^* & & \\ & \Rightarrow & \cdots & \Rightarrow & u\alpha_n v \in R^* & \Rightarrow & uv \in R^*. \end{array}$$

Since (R1) is just (W1) with W replaced by R^*, we now know that R^* satisfies (W1) and (W2), and is therefore the word problem of a group G, by Proposition 4.2.1. Let $W = R^* = W^m_\Sigma(G)$; then we are finished if we can show that $R = \textsc{min}(W) \cap \Sigma^+$.

Suppose that $\alpha \in R$, with $\alpha \equiv \beta\gamma$, $\beta \in W$, $\gamma \not\equiv \lambda$, and $\beta \not\equiv \lambda$. Since $W = R^*$, we have $\beta \equiv \beta_1 \ldots \beta_m$ with $\beta_i \in R$. So $\alpha \equiv \beta_1\gamma'$, with $\beta_1 \in R$ and $\gamma' \not\equiv \lambda$, contradicting (R4). Noting that the empty word is not in R, we have that $R \subseteq \textsc{min}(W) \cap \Sigma^+$.

Conversely, let $\alpha \in \textsc{min}(W) \cap \Sigma^+$. Since $\alpha \in W = R^*$, we have $\alpha \equiv \alpha_1 \ldots \alpha_n$ with $\alpha_i \in R$. Since $\alpha_1 \in R \subseteq W$, and $\alpha \in \textsc{min}(W)$, we know that $\alpha \equiv \alpha_1$, and thus that $\alpha \in R$. Thus $\textsc{min}(W) \cap \Sigma^+ \subseteq R$, and so $R = \textsc{min}(W) \cap \Sigma^+$ is the reduced word problem of G. $\square$

4.4 Solvability

One would expect that solvability of the word problem for a group is closely linked to the solvability of the reduced word problem. This is indeed the case, and we have:

Theorem 4.4.1 *If W is the word problem of a group with respect to a monoid generating set X and if R is the reduced word problem with respect to X, then the following are equivalent:*

(S1) *R is recursive;*

(S2) *R is recursively enumerable;*

(S3) *W is recursive.*

Proof. The fact that (S1) implies (S2) is clear; it remains to show that (S2) implies (S3) and that (S3) implies (S1).

Let us assume that R is recursively enumerable. Let $\alpha \in \Sigma^*$. If no prefix of α lies in R, choose β of minimal length such that $\alpha\beta \in W$; we see that we must have $\alpha\beta \in R$. So

$$\begin{aligned} \alpha \in \Sigma^* \quad \Rightarrow \quad & \text{either there is a prefix of } \alpha \text{ in } R \\ & \text{or there is a non-empty word } \beta \text{ such that } \alpha\beta \in R \end{aligned} \tag{4.1}$$

We have a procedure $\mathcal{P}$ which, when given as input a word $\gamma \in \Sigma^*$, will terminate if and only if $\gamma \in R$. We will outline an algorithm $\mathcal{A}$ which, given a word α in Σ^*, will terminate with "yes" if $\alpha \in W$ and with "no" otherwise.

(A1) We set $\mathcal{P}$ going on every non-empty prefix of α. If $\mathcal{P}$ terminates on some such prefix η, say $\alpha \equiv \eta\theta$, then η represents the identity, so that $\alpha \in W$ if and only if $\theta \in W$. We delete η and restart $\mathcal{A}$ completely with θ in place of α. Note that, if $\alpha \equiv \eta$ and $\theta \equiv \lambda$, then $\alpha \in W$ and we have finished.

(A2) Whilst doing (A1), we enumerate words of the form $\alpha\beta$ which have α as a proper prefix. For each such word $\alpha\beta$, we run $\mathcal{P}$ on that word. If $\mathcal{P}$ terminates on such a word $\alpha\beta$, we know that $\alpha\beta \in R$ with β non-empty, and so $\alpha \notin W$. Note that, if we find a prefix η of α in R via (A1), then these procedures in (A2) are all abandoned when we restart $\mathcal{A}$ with θ in place of α.

By (4.1), our algorithm $\mathcal{A}$ either terminates with the empty word whilst performing (A1), in which case $\alpha \in W$, or else we find, in (A2), a word $uv \in R$, such that u and v are non-empty, and our original α is wu, for some $w \in W$; but then $\alpha = u \notin W$. So W is recursive.

Lastly, suppose that W is recursive. Given a word α, we test α and all its proper non-empty prefixes for membership of W. We have that $\alpha \in R$ if and only if $\alpha \in W$ and no proper non-empty prefix of α lies in W. So R is recursive as required. $\square$

It is interesting to note that there are no groups where the reduced word problem is recursively enumerable but not recursive, and so any language satisfying the conditions of Proposition 4.3.1 that is recursively enumerable is necessarily recursive.

Chapter 5

Syntactic Monoids

5.1 Characterizations of Languages

As has been noted in [24], it is not possible to give a characterization of the context-free languages using syntactic monoids in the same way as has been done for the regular languages, since languages which are very different in terms of position in the Chomsky Hierarchy can have the same syntactic monoid. For example, the context-free languages are not closed under complementation, and, by Proposition 2.9.2, a language always has the same syntactic monoid as its complement; therefore there are monoids which are the syntactic monoids of languages which are context-free and of languages further up the hierarchy. In fact, there is little hope of such a characterization for any class of languages which is closed under inverse homomorphism and which strictly contains the regular languages.

Theorem 5.1.1 *Let G be a finitely generated non-periodic group and let $\mathcal{F}$ be a family of languages which is closed under inverse homomorphisms and intersections with regular languages such that there exists $K \subseteq \{a\}^*$ with $K \notin \mathcal{F}$. Then $G = M_L$ for some $L \notin \mathcal{F}$.*

Proof. Let a be an element of G of infinite order. Let X be a group generating set for G containing a, let $\Sigma = X \cup X^{-1}$, and let $\phi : \Sigma^* \to G$ be the natural homomorphism.

Let K be a subset of $\{a\}^*$ such that $K \notin \mathcal{F}$, and let $I = \{i : a^i \in K\}$. If $K \cup \{\lambda\} \in \mathcal{F}$, then $(K \cup \{\lambda\}) \cap \{a\}^+ = K \in \mathcal{F}$, a contradiction. So we may assume without loss of generality that $\lambda \in K$, and thus that $0 \in I$. Let $S = \{a^i : i \in I\} \subseteq G$ (since $0 \in K$, we must have $1 \in S$), and let $L = S\phi^{-1}$.

If $L \in \mathcal{F}$, then $K = L \cap \{a\}^* \in \mathcal{F}$, a contradiction; so $L \notin \mathcal{F}$.

In order to show that $G = M_L$ we must show that there is no non-trivial congruence on G such that S is a union of congruence classes. Let $\sim$ be a non-trivial congruence on G. Suppose that a^i and a^j are in S, with $i > j$, and that $a^j \sim a^i$. Then $a^{j-i} \sim 1 \in S$, and $a^{j-i} \in S$ with $j - i < 0$, a contradiction. $\square$

In the light of the problems with using syntactic monoids to classify languages above the regular languages in the Chomsky Hierarchy, Sakarovitch suggests in [27] the framework of *syntactic pointed monoids*, effectively, that languages should be classified by the structure of the syntactic monoid and the image of the language in that monoid.

It would therefore be useful to have methods of finding out whether or not a particular subset of a monoid is disjunctive. Since the set of congruences on a group is in bijective correspondence with the set of normal subgroups we have that, in the case of groups, a subset A of a group G is disjunctive if there is no non-injective homomorphism ϕ from G onto a group G' such that A is the full inverse image of a subset B of G'.

It the subset that we are considering is a subgroup, then we have the following result.

Proposition 5.1.2 *Let G be a group, and H a subgroup of G. Then H is a disjunctive subset of G if and only if it contains no non-trivial normal subgroup of G.*

Proof. Assume H is not a disjunctive subset of G. Then there is some homomorphism ϕ from G onto G', say, such that H is the inverse image of some subset $B \subseteq G'$. Since the identity element of G is in H, and H is the inverse image of some subset of G', H must contain $Ker(\phi)$, a non-trivial normal subgroup of G.

Conversely, assume H contains a non-trivial normal subgroup N of G. Consider the natural homomorphism, ψ, from G onto G/N. By the First Isomorphism Theorem, ψ gives a bijective correspondence between the set of subgroups of G containing N and the set of subgroups of G/N. Hence there is a subgroup $H\psi$ of G/N such that $(H\psi)\psi^{-1} = H$. $\square$

An immediate corollary of Proposition 5.1.2 is the following well-known observation.

Proposition 5.1.3 *A group is the syntactic monoid of its word problem.*

Proof. Let G be a group. Then $\{1\}$ is a subgroup of G which contains no non-trivial normal subgroup of G. By the previous result, $\{1\}$ is a disjunctive subset of G, and thus G is the syntactic monoid of its word problem. $\square$

This gives us another proof of the following result of [2].

Corollary 5.1.4 *The groups with regular word problems are exactly the finite groups.*

Proof. A finite group G is the syntactic monoid of its word problem which must therefore be regular, by Theorem 2.9.6. If G has regular word problem then it is the syntactic monoid of a regular language and hence is finite, again, by Theorem 2.9.6. $\square$

5.2 Word Problems

We note that, given a language $L \subseteq X^*$, the word problem of the syntactic monoid M_L of L with respect to the the generating set X is the set of words over X which are equal to the identity, i.e. the congruence class $[\lambda]$ of the empty word under the syntactic congruence.

The following lemma allows us to assume, when considering the syntactic monoid of a language L from a class of languages which is closed under inverse homomorphism, that our alphabet contains inverses.

Lemma 5.2.1 *Let $\mathcal{F}$ is a class of languages which is closed under inverse homomorphism, and let $L \subseteq X^*$ be a language in $\mathcal{F}$. If M_L, the syntactic monoid of L, is a group G, then G is also the syntactic monoid of a language L' over the generating set $\Sigma = X \cup X^{-1}$, with $L' \in \mathcal{F}$.*

Proof. Let S be the image of L in G under η_L, the syntactic morphism (note that $L = S\eta_L{}^{-1}$). Let ϕ be the monoid homomorphism from Σ^* to G, which maps each $x \in X$ to $x\eta_L$, and each $x^{-1} \in X^{-1}$ to $(x\eta_L)^{-1}$. Since η_L is surjective, ϕ must also be surjective.

Let K be the inverse image of S under ϕ. Since S is a disjunctive subset of G, we know that G is the syntactic monoid of K. By Lemma 3.5.1, we must have $K \in \mathcal{F}$. $\square$

It will be useful to note the following characterization of the word problem of the syntactic monoid of a language.

Lemma 5.2.2 *Let L be a language over an alphabet X. Then the word problem W of the syntactic monoid of L is $I \cap D$, where $I = \text{INS}(L)$ and $D = \text{DEL}(L)$.*

Proof. We observe that

$$u \in W = [\lambda] \quad \Leftrightarrow \quad (w_1 w_2 \in L \Leftrightarrow w_1 u w_2 \in L)$$

$$
\begin{aligned}
&\Leftrightarrow \quad (w_1w_2 \in L \Rightarrow w_1uw_2 \in L) \text{ and } (w_1uw_2 \in L \Rightarrow w_1w_2 \in L) \\
&\Leftrightarrow \quad u \in I \text{ and } u \in D
\end{aligned}
$$

which is exactly what we wanted. □

Suppose $\Sigma = X \cup X^{-1}$, and define ${}^{-1} : \Sigma^* \to \Sigma^*$ inductively by:

- $\lambda^{-1} = \lambda$;
- if $a \in X$ then $(a\alpha)^{-1} = \alpha^{-1}a^{-1}$;
- if $a \in X^{-1}$ then $(a\alpha)^{-1} = \alpha^{-1}b$ where $a = b^{-1}$, and $b \in X$.

If A and B are subsets of Σ^*, then

$$
\begin{aligned}
(A \rightleftharpoons B)^{-1} &= \{x \in \Sigma^* : \exists \alpha, \beta \in \Sigma^*(u \equiv \alpha x \beta \in A \text{ and } v \equiv \alpha\beta \in B)\}^{-1} \\
&= \{x^{-1} \in \Sigma^* : \exists \alpha, \beta \in \Sigma^*(u \equiv \alpha x \beta \in A \text{ and } v \equiv \alpha\beta \in B)\} \\
&= \{x^{-1} \in \Sigma^* : \exists \alpha^{-1}, \beta^{-1} \in \Sigma^*(u^{-1} \equiv \beta^{-1}x^{-1}\alpha^{-1} \in A^{-1} \\
&\qquad \text{and } v^{-1} \equiv \beta^{-1}\alpha^{-1} \in B^{-1})\} \\
&= \{y \in \Sigma^* : \exists \gamma, \delta \in \Sigma^*(\gamma y \delta \in A^{-1} \text{ and } \gamma\delta \in B^{-1})\} \\
&= (A^{-1} \rightleftharpoons B^{-1})
\end{aligned}
$$

and

$$
\begin{aligned}
(A^c)^{-1} &= \{x \in \Sigma^* : x \notin A\}^{-1} &&= \{x^{-1} \in \Sigma^* : x \notin A\} \\
&= \{x^{-1} \in \Sigma^* : x^{-1} \notin A^{-1}\} &&= \{y \in \Sigma^* : y \notin A^{-1}\} \\
&= (A^{-1})^c,
\end{aligned}
$$

and we have the following lemma:

Lemma 5.2.3 *Let A and B be subsets of Σ^*. Then*

i. $(A \rightleftharpoons B)^{-1} = (A^{-1} \rightleftharpoons B^{-1})$*; and*

ii. $(A^c)^{-1} = (A^{-1})^c$.

We can now show the following:

Proposition 5.2.4 *Let L be a language over the alphabet $\Sigma = X \cup X^{-1}$, such that M_L is a group G, the function $^{-1}$ is defined as above, and X is a group generating set for G. Let $I = \text{INS}(L)$ and $D = \text{DEL}(L)$. Then $D = I^{-1} = \text{INS}(L^{-1})$.*

Proof. We must have $\alpha\alpha^{-1} \in W_\Sigma^m(G)$ for any $\alpha \in \Sigma^*$, so that $\alpha\alpha^{-1} \in I \cap D$ for any $\alpha \in \Sigma^*$ by Lemma 5.2.2.

Let $\alpha \in I$. Then $u\alpha^{-1}v \in L$ implies that $u\alpha\alpha^{-1}v \in L$ (since $\alpha \in I$), and thus $uv \in L$ (since $\alpha\alpha^{-1} \in D$), and we see that $\alpha^{-1} \in D$.

Let $\alpha^{-1} \in D$. Then $uv \in L$ implies that $u\alpha\alpha^{-1}v \in L$ (since $\alpha\alpha^{-1} \in I$), and thus $u\alpha v \in L$ (since $\alpha^{-1} \in D$), and we have $\alpha \in I$.

We now have that $D = \{\alpha^{-1} \in \Sigma^* : \alpha \in I\} = I^{-1}$, and thus,

$$D = I^{-1} = [(L^c \rightleftharpoons L)^c]^{-1} = [(L^{-1})^c \rightleftharpoons (L^{-1})]^c = \text{INS}(L^{-1})$$

by Lemma 5.2.3 and Proposition 2.8.1. $\square$

In particular, given Lemma 5.2.2, we have

Corollary 5.2.5 *If $L = L^{-1}$ is a language over the alphabet $\Sigma = X \cup X^{-1}$, the syntactic monoid of L is a group G, the function $^{-1}$ is defined as above, and X is a monoid generating set for G, then $W_\Sigma^m(G) = \text{INS}(L) = \text{DEL}(L)$.*

One general question which we shall look at is the following:

Question 5.2.6 *Given a language $L \subseteq \Sigma^*$, from a family $\mathcal{F}$ which is closed under inverse homomorphism, such that the syntactic monoid of L is a group, is the word problem of the syntactic monoid of L in $\mathcal{F}$?*

More generally:

Question 5.2.7 *Given a language L from family $\mathcal{F}$, what can we say about the word problem of M_L?*

Lemma 5.2.1 tells us that, when trying to answer Question 5.2.6, we may assume without loss of generality that the alphabet Σ^* contains inverses.

We can now give the answer to Question 5.2.6 when $\mathcal{F}$ is the class of regular langages.

Theorem 5.2.8 *If L is a regular language with syntactic monoid M then the word problem of M is a regular language.*

Note that this theorem in the case where M is a group follows immediately from Theorem 2.9.6 and Corollary 5.1.4.

Proof. Let $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ be the minimal complete deterministic finite automaton recognizing L. For each state $q \in Q$, consider the language $L_q \subseteq \Sigma^*$ of all words $w \in \Sigma^*$ for which $\delta(q, w) = q$. We claim that $\bigcap_{q \in Q} L_q$ is the word problem W of M. By Lemma 5.2.2 we know that $W = \textsc{ins}(L) \cap \textsc{del}(L)$.

Suppose $u \in \bigcap_{q \in Q} L_q$ and that $w_1, w_2 \in \Sigma^*$. Since $u \in \bigcap_{q \in Q} L_q$, we have $\delta(q_0, w_1 w_2) = \delta(q_0, w_1 u w_2)$ and hence $w_1 w_2 \in L \Leftrightarrow w_1 u w_2 \in L$, and $u \in \textsc{ins}(L) \cap \textsc{del}(L) = W$.

Conversely, suppose that $u \notin \bigcap_{q \in Q} L_q$: that is, there exists $r \in Q$ with $u \notin L_r$. By completeness and minimality we can choose $w_1 \in \Sigma^*$ such that $\delta(q_0, w_1) = r$. Since $u \notin L_r$, $\delta(r, u) \neq r$. Let $\delta(r, u) = r'$. By minimality there exists $w_2 \in \Sigma^*$ such that $\delta(r, w_2) \in F$ and $\delta(r', w_2) \notin F$, or vice versa. We have shown that there exist w_1 and w_2 in Σ^* such that $\delta(q_0, w_1 w_2) \in F$ and $\delta(q_0, w_1 u w_2) \notin F$, or vice versa. Hence $w_1 w_2 \in L$ and $w_1 u w_2 \notin L$, or vice versa, and we cannot have $u \in W$. $\square$

Theorem 3.5.3 gives a partial positive answer to Question 5.2.6 in the case of context-free languages where the syntactic monoid is assumed to be cancellative:

Corollary 5.2.9 *Let L be a context-free language with cancellative syntactic monoid M. If the image of L in M is finite then M has a context-free word problem.*

Proof. Assume that $L \neq \emptyset$. Then the image of L in M_L is a finite non-empty context-free set, and by Theorem 3.5.3 all finite subsets of M_L are context-free; in particular $\{1\} \in \mathcal{CF}(M_L)$.

If L is empty then its syntactic monoid is the trivial group, which has context-free word problem. $\square$

In general, the word problem of a group which is the syntactic monoid of a context-free language need not be context-free. A particular example from [28] is the group $C_\infty \times C_\infty$.

Example 5.2.10 ([28]) *Let $\Sigma = \{a, b, c, d\}$, and let*

$$L = \{w \in \Sigma^* : |w|_a = |w|_b \text{ or } |w|_c = |w|_d\}.$$

Then L is a context-free language with syntactic monoid isomorphic to $C_\infty \times C_\infty$.

In the case of deterministic context-free languages, the answer to Question 5.2.6 is unknown.

The following conjecture was made in [12].

Conjecture 5.2.11 *Let $L \subseteq \Sigma^*$ be a deterministic context-free language with syntactic monoid a group G. Then G is a context-free group.*

This reduces to our Question 5.2.6 for deterministic context-free languages, since, by Theorem 3.6.2 the word problem of a context-free group is always deterministic context-free. It is noted in [12] that if Conjecture 5.2.11 could be proved then it would lead to a proof of the conjecture of Sakarovitch ([26, 29]), that the thin syntactic monoids are exactly the deterministic monoids, in the special case of groups.

We finish this chapter by looking at the syntactic monoids of recursive languages.

Proposition 5.2.12 *Let L be a recursive language with syntactic monoid M. Then the complement of the word problem of M is recursively enumerable.*

Proof. We shall assume that L is infinite: were L finite it would be regular, and thus the word problem of M would be regular, and hence certainly recursive. Since L is recursive we can assume the existence of a Turing machine $\mathcal{T}_1$ which enumerates the words of L, and a machine $\mathcal{T}_2$ which, upon reading a word of Σ^*, always terminates, and gives the answer yes if and only if the word is in L. We construct a machine $\mathcal{T}$ which, when given input u, halts if and only if u is in the complement of the word problem of M.

Our machine $\mathcal{T}$ uses a copy of $\mathcal{T}_1$ to enumerate words of L. For each word $w = x_1x_2 \ldots x_n \in L$, $\mathcal{T}$ inserts u into w in the $n+1$ possible places in turn and checks for each one, using a copy of $\mathcal{T}_2$ that the resulting word is in L. If one of these words fails to be in L, then $\mathcal{T}$ halts.

For each occurrence of u as a subword $x_i \ldots x_j$ of w, $\mathcal{T}$ uses a copy of $\mathcal{T}_2$ to see if $x_1 \ldots x_{i-1}x_{j+1} \ldots x_n$ is in L. If this is not the case then $\mathcal{T}$ halts.

Having checked to see if u can be inserted or deleted at any point in w in such a way that we finish up with a word not in L, $\mathcal{T}$ goes back to its copy of $\mathcal{T}_1$ and starts the process again with a new w. Since L is infinite, $\mathcal{T}$ will never run out of words to work on. $\Box$

Combining Theorem 3.5.2 and the previous result gives:

Proposition 5.2.13 *Let $\mathcal{F}$ be the context-sensitive languages, the recursive languages, or the recursively enumerable languages. If G is a finitely generated group and A is a finite, nonempty, disjunctive, $\mathcal{F}$-subset of G, then the word problem of G is in $\mathcal{F}$.*

Proof. If $\mathcal{F}$ is the context-sensitive languages or the recursively enumerable languages, then by Theorem 3.5.2 every finite subset of G is an $\mathcal{F}$-subset, and in particular $\{1\}$ is a $\mathcal{F}$-subset.

If $\mathcal{F}$ is the recursive languages then A is an $\mathcal{RE}$-subset of G and thus $\{1\}$ is also an $\mathcal{RE}$-subset. By Proposition 5.2.12, $G \setminus \{1\}$ is recursively enumerable. So $\{1\}$ is recursive. $\square$

We now have a partial answer to Question 5.2.6 in the case of recursive languages.

Proposition 5.2.14 *Let $L \subseteq \Sigma^*$ be a recursive language, with syntactic monoid a group G, and suppose that the image of L under the syntactic morphism is finite. Then G has a recursive word problem.*

Proof. The image L under its syntactic morphism is always disjunctive. If L is nonempty, then its image in G is nonempty and we may apply Proposition 5.2.13. If L is empty then its syntactic monoid is the trivial group. $\square$

Chapter 6

Reduced Word Problems

6.1 Haring-Smith's Conjecture

The purpose of this section is to prove Theorem 6.1.1, which includes (as the equivalence of parts i and vi) the conjecture of Haring-Smith that a group has strict deterministic reduced word problem for some monoid generating set if and only if it is a finite extension of a plain group. Saying that a reduced word problem is strict deterministic is equivalent to just saying that it is a deterministic context-free language (since it is prefix-free by definition).

Theorem 6.1.1 *Let G be a finitely generated group. The following are equivalent:*

i. for some monoid generating set X, the reduced word problem of G with respect to X is deterministic context-free;

ii. for some monoid generating set X, the reduced word problem of G with respect to X is context-free;

iii. G is a context-free group;

iv. for every monoid generating set X, the reduced word problem of G with respect to X is deterministic context-free;

v. for every monoid generating set X, the reduced word problem of G with respect to X is context-free;

vi. G is a finite extension of a plain group.

Proof. It is clear that part i implies ii. Next we prove that part ii implies part iii.

Let X be a monoid generating set for a group G, such that $R_X^m(G)$ is context-free. Since the class of context-free languages is closed under Kleene star, and $W_X^m(G) = R_X^m(G)^*$, by Proposition 4.1.1, $W_X^m(G)$ is context-free, and we have shown that part ii implies part iii.

Let G be a context-free group, and let X be a finite monoid generating set for G, so that, by Theorem 3.6.2, $W_X^m(G)$ is deterministic context-free. Since $R_X^m(G) = \textsc{min}(W_X^m(G) \cap X^+)$ by Proposition 4.1.2, it must be deterministic context-free, by the closure of the deterministic context-free languages under intersections with regular sets, and Lemma 2.7.2. We have proved that part iii implies part iv, and it is immediate that part iv implies part v.

If part v is true, take X to be any monoid generating set for G. Since $R_X^m(G)$ is context free, $W_X^m(G) = R_X^m(G)^*$ is context-free, and so $W_X^m(G)$ is deterministic context-free by Theorem 3.6.2, and $R_X^m(G)$ is deterministic context-free by Lemma 2.7.2 and the closure of the deterministic context-free languages under intersections with regular sets. So we have shown that parts i, ii, iii, iv and v are equivalent. It therefore only remains to show that part vi is equivalent to these five conditions.

A context-free group G is a finite extension of a free group, and, since a free group is certainly plain, G must be a finite extension of a plain group, and so we have proved that part iii implies part vi. Let G be a finite extension of a plain group H. Since H is plain, by Theorem 3.7.2, it has simple reduced word problem for some generating set. A simple language is certainly context-free, and, by the equivalence of parts ii and iii, we know that H is a context-free group. The class of context-free groups is closed under taking finite extensions, so that G must also be context-free. $\square$

An obvious extension of the work of Haring-Smith would be to classify the groups with reduced word problem in other classes of languages. In the case of the one-counter languages, Herbst's characterization of the one-counter groups can play much the same role as Muller and Schupp's characterization of the context-free groups. Substituting one-counter for context-free throughout, and using Theorem 3.6.3 in place of Theorem 3.6.2 we see that the proof of the equivalence of parts i, ii, iii, iv and v of Theorem 6.1.1 works equally well, and so we have:

Theorem 6.1.2 *Let G be a finitely generated group. The following are equivalent:*

i. for some monoid generating set X, the reduced word problem of G with respect to X is deterministic one-counter;

ii. for some monoid generating set X, the reduced word problem of G with respect to X is one-counter;

iii. G is a one-counter group;

iv. for every monoid generating set X, the reduced word problem of G with respect to X is deterministic one-counter;

v. for every monoid generating set X, the reduced word problem of G with respect to X is one-counter;

6.2 Monoid Generating Sets

In this section we shall look at the analogue of Theorems 3.7.1 and 3.7.2 in the case of monoid generating sets.

We start by constructing a grammar Γ from the finite irreducible word problem of a group G with respect to monoid generating set X. The construction is the same as that in [10].

Let $I = \{w_1, w_2, \ldots, w_n\}$, where $w_i \equiv a_{i,1}a_{i,2}\ldots a_{i,n_i}$ for $i \in \{1, \ldots, n\}$, be the irreducible word problem of G with respect to X; let I_s be the suffix closure of I.

The set of terminals for our grammar shall be X. The set of non-terminal symbols, V, will consist of a symbol A_g for each $g \in G$ which is represented by an element of I_s. Since the identity element of the group is represented by the empty word, which is in I_s, there is a symbol A_1 corresponding to it. This symbol will be the start symbol for our grammar, and we shall therefore write S in place of A_1. We note that for each $a \in X$, there must be a symbol $A_{a^{-1}} \in V$, since there must be a word starting with a which is equal to the identity, and a minimal such word must be in I. We now have a set of terminals, a set of non-terminals, and a start symbol; so all that remains to be defined is the set of productions. This is constructed in the following way:

- $S \to a_{i,1}A_{a_{i,1}^{-1}} \in P$ for $i \in \{1, \ldots, n\}$;
- if there is a word in $I_s \setminus \{\lambda\}$ representing the (non-identity) group element $a^{-1}g$, then $A_g \to aA_{a^{-1}g} \in P$;
- if $a^{-1}g$ is the identity element, then $A_g \to a \in P$;
- if $a^{-1}g$ is not represented by a word in I_s, then $A_g \to aA_{a^{-1}}A_g \in P$.

Since, for each pair of one terminal a, and one non-terminal A, we have constructed exactly one rule of the form $A \to a\alpha$, our grammar is simple.

Lemma 6.2.1 *The language of the grammar we have just constructed is $R^m_X(G)$.*

Proof. We prove by induction that the words derived from each non-terminal A_g all represent g in the group. In a derivation $A_g \to a$ of length one, we have, by definition, that $a^{-1}g = 1$, and thus that $a = g$. Let us assume that we have proved that every word which can be derived from a symbol A_g using a derivation of length less than or equal to m, where $m \geqslant 1$, is equal in the group to g, for every non-terminal symbol $A_g \in V$. Let w be a word which can be derived from A_g with a derivation of length $m + 1$. The derivation of w must start in one of the following ways:

- $A_g \to aA_{a^{-1}g}$;
- $A_g \to aA_{a^{-1}}A_g$.

The lengths of the derivations from the non-terminals on the right hand sides of the rules above must in all cases be less than or equal to m. So in the first case, $A_{a^{-1}g}$ derives a word u which is equal to $a^{-1}g$. In this way A_g derives $au = aa^{-1}g = g$.

In the second case, $A_{a^{-1}}$ derives a word u equal to a^{-1}, and the A_g on the right hand side derives, in fewer than m moves, a word v which is equal to g. Thus the A_g on the left hand side of the production derives $auv = aa^{-1}g = g$.

In particular, we have shown that every word which can be derived from $S = A_1$ is equal to 1, and therefore that $L(\Gamma) \subseteq W^m_X(G)$. Since the grammar Γ is simple, $L(\Gamma)$ is prefix-free, and thus $L(\Gamma) \subseteq R^m_X(G)$.

To complete the proof of the Lemma we need to show that every word which is in $R^m_X(G)$ is also in $L(\Gamma)$. This follows from the fact that for every non-terminal A, and terminal a, there is exactly one rule of the form $A \to a\alpha$ in P. Given a word $w \equiv a_1 a_2 \ldots a_k \in R^m_X(G)$, we construct a leftmost derivation of w in Γ in the only possible way. Assume that we have already constructed the first m moves of the derivation $S \xrightarrow{*} a_1 \ldots a_m A\beta$, where $m < k$. The next letter of w is a_{m+1}, so the next production must be of the form $A \to a_{m+1}\alpha$, and in $m+1$ moves, we derive the string $a_1 \ldots a_m a_{m+1}\alpha\beta$. If $\alpha\beta$ is empty then the string $a_1 \ldots a_{m+1}$ is in $L(\Gamma) \subseteq R^m_X(G)$, so this can only happen if $m = k-1$, and our derivation cannot come to an end before the end of w. If $m = k-1$, and $\alpha\beta$ is not the empty string, then we have derived the string $a_1 \ldots a_k\alpha\beta \equiv w\alpha\beta$. There must be a non-empty string u which derives from $\alpha\beta$, and so we have that $S \xrightarrow{*} wu$, where u is non-empty. This cannot happen, since we would have $wu \in L(\Gamma) \subseteq R^m_X(G)$ with a non-empty proper prefix $w \in R^m_X(G)$. So we know that our derivation does finish at the end of w. $\square$

The previous lemma (together with the preceding discussion) gives a proof of the following theorem:

Theorem 6.2.2 *If a group G has finite irreducible word problem with respect to a monoid generating set X, then G has simple reduced word problem with respect to X.* $\square$

In contrast to the situation for group generating sets, the converse of Theorem 6.2.2 remains open.

Question 6.2.3 *If a group G has simple reduced word problem with respect to some monoid generating set X, does the irreducible word problem of G with respect to X have to be finite?*

Chapter 7

Finite Irreducible Word Problems

7.1 Preliminaries

The following observation, implicit in [20], tells us that the groups presented by finite special $[\lambda]$-confluent string-rewriting systems and the groups which have finite irreducible word problem with respect to some monoid generating set are in fact one and the same.

Proposition 7.1.1 *Let G be a group, and X a finite monoid generating set for G. Then G can be presented by a finite special $[\lambda]$-confluent string-rewriting system over X if and only if it has finite irreducible word problem with respect to X.*

Proof. Let R be a finite special $[\lambda]$-confluent string-rewriting system over X which presents G. If $w \in I = I_X^m(G)$ then, since $w \in W_X^m(G)$, there is a rule $u \to \lambda$ which can be applied to w. Now u is a non-empty subword of w which is equal to the identity, so $u \equiv w$. We have shown that, for any $w \in I$, there must be a rule $w \to \lambda$ in R. Since R has only finitely many rules, I must be finite.

Conversely, let G be a group with finite monoid generating set X such that $I = I_X^m(G)$ is finite. Then $\{w \to \lambda : w \in I\}$ is a finite special $[\lambda]$-confluent string-rewriting system which presents G. □

7.2 Special Rewriting for $C_\infty \times C_k$

We know from [20] that a group can be presented by a finite special $[\lambda]$-confluent string-rewriting system which provides inverses of length one for each generator if and only if it is a plain group, but it is left open there as to whether or not this continues to be the case if the condition on inverses is dropped. Our task here is to give an example to show that this is not the case.

Proposition 7.2.1 *Let G be the direct product of the infinite cyclic group and a finite cyclic group. Then G can be presented by a finite special $[\lambda]$-confluent string-rewriting system, but G is not plain.*

Proof. Firstly, we note that G cannot be plain, since a group cannot be both a non-trivial direct product and a non-trivial free product: see for example [18, page 177].

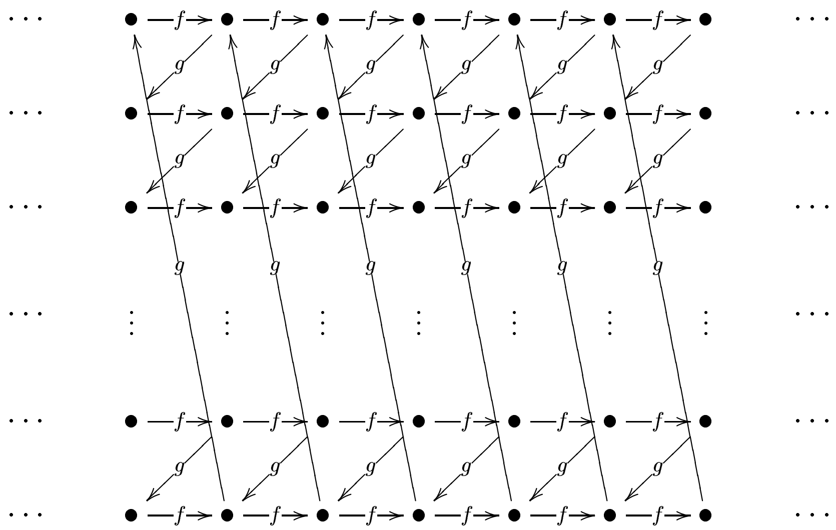

Figure 7.1: The Cayley graph of $C_\infty \times C_k = \langle f : \ \rangle \times \langle a : a^k \rangle$ with respect to the monoid generating set $\{f, g\}$, where $g = af^{-1}$.

The standard group presentation for G is $\langle f : \ \rangle \times \langle a : a^k \rangle$. Let $X = \{f, g\}$, where $g = af^{-1}$. The Cayley graph of G with respect to X is shown in Figure 7.1. We claim that X is a monoid generating set for G with respect to which the irreducible word problem of G is finite.

Any element of G can be written in terms of the symbols f, f^{-1} and a. Since $f \in X$, $a = fg$ and $f^{-1} = f^{k-1}g^k$, we see that X is a monoid generating set for G.

A word w is in $W_X^m(G)$ if and only if it satisfies $|w|_f = |w|_g$ and $|w|_g \equiv 0 \bmod k$. Together these conditions give that $|w|_f = |w|_g = nk$, and $|w| = 2nk$, for some $n \geqslant 0$. We shall show that if $n > 1$ then w cannot be in $I = I_X^m(G)$, so that I is just the set of words which contain exactly k

instances of f and k instances of g, and is thus a finite set.

Assume that $w \equiv z_1 \dots z_{2nk} \in I$ and that $n > 1$. Consider the word $z_1 \dots z_{2k}$. This subword of w must contain different numbers of instances of f and g, or else it would be equal to the identity in G. We may therefore assume without loss of generality that it contains more instances of f than it does of g. Now suppose that, for some i, the subword $z_i \dots z_{i+2k-1}$ contains more instances of f than of g. The difference between $|z_i \dots z_{i+2k-1}|_f$ and $|z_{i+1} \dots z_{i+2k}|_f$ must be at most 1, and so $|z_{i+1} \dots z_{i+2k}|_f \geqslant k$. We cannot have $|z_{i+1} \dots z_{i+2k}|_f = k$ because this would mean that $z_{i+1} \dots z_{i+2k}$ is equal to the identity; so $|z_{i+1} \dots z_{i+2k}|_f > k$. By induction we see that every subword of w consisting of $2k$ consecutive symbols contains more instances of f than it does instances of g. Since $|w|$ is a multiple of $2k$, we may consider w as the concatenation of n words of $2k$ symbols, and it is clear that $|w|_f > |w|_g$, a contradiction. $\square$

A group of the form $C_\infty \times C_n$ has a finitely generated abelian subgroup which is neither finite nor infinite cyclic and thus by Theorem 3.8.5, cannot be presented by a finite monadic confluent string-rewriting system. We therefore have examples of groups which can be presented by finite special $[\lambda]$-confluent string-rewriting systems, but which cannot be presented by a finite monadic confluent string-rewriting system. It is not known whether or not there exists a group which can be presented by a finite monadic confluent system, but not by a finite special $[\lambda]$-confluent system: such a group would, of course, be a counter-example to Conjecture 3.8.4.

7.3 Infinite Cyclic Central Subgroups

Having demonstrated that the class of groups which may be presented by finite special $[\lambda]$-confluent string-rewriting systems strictly contains the plain groups, the obvious question is:

Question 7.3.1 *Which groups can be presented by finite special $[\lambda]$-confluent string-rewriting systems?*

We shall answer this question for groups which have an infinite cyclic central subgroup. We start with two lemmas.

Lemma 7.3.2 *Let X be a finite monoid generating set for the group G. Suppose there are a finite set $S \subseteq G$, an element $g \in G$, and an infinite set P of pairs of words over X such that, for all $(u_1, u_2) \in P$, both u_1 and u_2 have no non-empty subwords equal to the identity, $u_1 g u_2 \in S$, and $u_+ g u_- = 1$ implies $u_+ \equiv u_- \equiv \lambda$ whenever u_+ is a suffix of u_1 and u_- is a prefix of u_2. Then $I_X^m(G)$ is infinite.*

Proof. For each $s \in S$, let the word v_s represent a simple path in the Cayley graph Γ of G connecting s to 1 so that $v_s = s^{-1}$. Since S is finite, there is an integer B such that the length of each such path does not exceed B. Let $C = |v_g|$, where v_g is a word representing g.

Suppose $I_X^m(G)$ were finite. By Proposition 7.1.1 there is then a finite special $[\lambda]$-confluent string-rewriting system R presenting G. Let M be larger than the length of the left hand side of any rule in R. Observe that an application of any R-rule to the label of any path in Γ corresponds to the removal from that path of a loop of length less than M.

Fix an arbitrary integer $L \geqslant M$. Since P is infinite, there must be a pair $(u_1, u_2) \in P$ with $|u_1| + |u_2| \geqslant L$. Consider the word $u_1 v_g u_2 v_s$, where $u_1 v_g u_2 = s \in S$ so that $u_1 v_g u_2 v_s = 1$. By the assumptions on (u_1, u_2) and L, the application of any R-rule to (any R-descendant of) the word $u_1 v_g u_2 v_s$ must remove at least one edge of the loop $u_1 v_g u_2 v_s$ outside the subpaths labelled by u_1 and u_2. Therefore one reaches an R-irreducible descendant w of $u_1 v_g u_2 v_s$ in at most $B + C$ steps. We have $w = 1$ and

$$|w| \geqslant |u_1 v_g u_2 v_s| - M(B + C) \geqslant L - M(B + C).$$

Since this holds for all $L \geqslant M$, we see that there are non-trivial R-irreducible words in $W_X^m(G)$. This contradicts the fact that R is special and $[\lambda]$-confluent, and completes the proof. $\square$

Lemma 7.3.2 has the following consequence:

Lemma 7.3.3 *Let X be a finite monoid generating set for the group G. Suppose there are a finite set $S \subseteq G$ and an infinite set U of words over X such that no $u \in U$ has a non-empty proper subword equal to the identity, and every $u \in U$ is equal in G to an element of S. Then $I_X^m(G)$ is infinite.*

Proof. We may clearly assume that no element of U is equal to the identity. Then Lemma 7.3.2 applies with $g = 1$ and $P = \{ (\lambda, u) \colon u \in U \}$. $\square$

We may now characterize the groups which may be presented by finite special $[\lambda]$-confluent string-rewriting systems amongst those groups which are split extensions with infinite cyclic quotient and finite kernel.

Proposition 7.3.4 *A group G which is of the form $H \rtimes F$, where H is a finite normal subgroup of G and F is an infinite cyclic group, can be presented by a finite special $[\lambda]$-confluent string-rewriting system if and only if $G = H \times F$ and H is cyclic.*

Proof. Proposition 7.2.1 tells us that the "if" part the theorem is true.

To prove the "only if" direction, we shall assume that G is not of the form $H \times F$ for any finite cyclic group H, and deduce that G has infinite irreducible word problem with respect to any monoid generating set.

Let X be a finite monoid generating set for G. We may assume that there are no redundant elements in X, since if $I_X^m(G)$ is infinite, then adding more elements to X is certainly not going to make it finite.

Fix a generator f (not necessarily in X) of the infinite cyclic group F and let $\phi : G \to F$ be the natural projection of G onto F. We start by partitioning X into three sets, P, N, and Z, where $P = \{x \in X : x\phi = f^i \text{ and } i > 0\}$, $N = \{x \in X : x\phi = f^i \text{ and } i < 0\}$, and $Z = \{x \in X : x\phi = 1\} = X \cap H$. We must have $P \neq \emptyset$ and $N \neq \emptyset$ in order that we have in X^* both words which are equal to f and words which are equal to f^{-1}.

We shall split the proof that $I_X^m(G)$ is infinite into several cases.

Case 1 $Z \neq \emptyset$.

Fix $a \in Z$, $x \in P$ and $y \in N$. We shall construct a sequence $(w_r : r \geqslant 0)$ of distinct words which satisfy the following conditions:

i. $w_r \in H \setminus \{1\}$;

ii. w_r has no non-empty proper subword which is equal to the identity;

iii. every x in w_r occurs to the left of the first y.

Let $m, n > 0$ be minimal such that $x^m y^n \in H$. We cannot have both $x^m y^n = 1$ and $x^m a y^n = 1$, since a would then be redundant. We may therefore choose w_0 to be either $x^m y^n$ or $x^m a y^n$ so that it satisfies condition i. The minimality of m and n tells us that w_0 satisfies condition ii, and it is clear that w_0 satisfies condition iii.

Assume that we have constructed w_r which satisfies conditions i, ii and iii. We cannot have both $x^m w_r y^n = 1$ and $x^m w_r a y^n = 1$, so we may choose w_{r+1} to be $x^m w_r y^n$, if this is not the identity, and $x^m w_r a y^n$ otherwise, so that w_{r+1} satisfies condition i. Condition iii for w_r, and the fact that all of the new instances of x are introduced on the left, and all of the new instances of y on the right, ensure that w_{r+1} satisfies condition iii. Our construction is complete if we can show that w_{r+1} satisfies condition ii.

By conditions i and ii for w_r, we know that there is no non-empty subword of w_r which is equal to the identity. Since $a\phi = w_r\phi = 1$, we have $(x^i w_r a y^j)\phi = (x^i w_r y^j)\phi = (x^i y^j)\phi$. But m and n are minimal such that

$(x^m y^n)\phi = 1$, and hence no subword of w_{r+1} the form $x^i w_r y^j$ or $x^i w_r a y^j$, with $0 < i < m$ and $0 < j < n$, is equal to the identity. By condition iii for w_r, any word v which is a suffix of w_r or of $w_r a$, must satisfy $v\phi = f^k$, for some $k \leqslant 0$. Thus any word of the form vy^i with $i > 0$, and v a suffix of w_r or of $w_r a$, must satisfy $(vy^i)\phi = f^k$, for some $k < 0$, and thus cannot be equal to the identity. The situation is similar for words of the form $x^i u$, where u is a prefix of $w_r a$, and $i > 0$. The only remaining possibility for a subword of w_{r+1} being equal to the identity is where we have $w_{r+1} \equiv x^m w_r a y^n$ (and hence $x^m w_r y^m = 1$), and the subword is $w_r a$. But then $y^n x^m w_r = 1$, and we have $y^n x^m = y^n x^m w_r a = a$, and a is redundant.

We have constructed an infinite sequence of words w_r of increasing length which are equal to elements of the finite group H and which have no non-empty subword equal to the identity, and we may therefore appeal to Lemma 7.3.3 to see that $I_X^m(G)$ is infinite.

Case 2 $|P| = |N| = 1$, $Z = \emptyset$.

Let $P = \{x\}$ and $N = \{y\}$ so that $X = \{x, y\}$. If x and y commute, then G is the direct product of an infinite cyclic group and a finite cyclic group, so we may assume that they do not commute. Let $i > 0$ and $j < 0$ be such that $x\phi = f^i$ and $y\phi = f^j$.

We shall split this case into three subcases. In Cases 2.1 and 2.2 we shall construct a sequence $(v_r : r \geqslant 0)$ of words with the following properties:

i. $v_r y = x^{-s_r}$ for some $s_r > 0$;

ii. $s_r > s_{r-1}$ for $r > 0$;

iii. $v_r y$ has no non-empty subword equal to the identity;

iv. v_r has no non-empty prefix equal (in G) to a negative power of x.

The set $\{w_r = x^{s_r} v_r y : r \geqslant 0\}$ is then an infinite subset of $I_X^m(G)$.

CASE 2.1 $|i| \neq |j|$.

We assume, without loss of generality, that $|i| > |j|$.

We shall require v_r to satisfy one further condition

v. $v_r \equiv y^{t_0-1} x y^{t_1-1} x \dots x y^{t_r-1}$ with all $t_k > 0$ (so that $v_r \equiv v_{r-1} x y^{t_r-1}$ for $r > 0$).

in addition to conditions i to iv.

It is easily seen that there exist $s > 0$ and $t > 0$ such that $x^s y^t = 1$. Let s_0 and t_0 be the minimal such s and t, and then let v_0 be y^{t_0-1}. Clearly v_0 satisfies properties i to v. Assume that we have constructed v_0 to v_r satisfying conditions i to v.

If there is no $k \geqslant 0$ such that $v_r x y^k$ is equal to a negative power of x then no word of the form $x^l v_r x y^k$ with $l > 0$ and $k \geqslant 0$ is equal to the identity. Elementary arithmetic involving i, j, and s_r shows that for each $k > 0$ there is an $l_k > 0$ satisfying $x^{l_k} v_r x y^k \in \bigcup_{|m| \leqslant i} f^m H$. Therefore Lemma 7.3.2 applies with $S = \bigcup_{|m| \leqslant i} f^m H$, $g = x v_r x$, and $P = \{(x^{l_k - 1}, y^k) : k > 0\}$ to establish that $I_X^m(G)$ is infinite.

We may therefore assume that there exists $k \geqslant 0$ such that $v_r x y^k$ is equal to a negative power of x. Let t_{r+1} be the least such k. If t_{r+1} were equal to zero, then we would have $v_r x = x^{-l}$, for some $l > 0$, and hence $v_r = x^{-l-1}$, a contradiction. Since $t_{r+1} \neq 0$, we may let v_{r+1} be $v_r x y^{t_{r+1}-1}$. By construction, v_{r+1} clearly satisfies conditions i, iv and v.

Next, we check that v_{r+1} satisfies condition ii. If s_{r+1} were equal to s_r, then we would have $v_r y = v_r x y^{t_{r+1}}$, and thus $x y^{t_{r+1}-1} = 1$, a contradiction since x and y do not commute. If s_r were greater than s_{r+1}, then $v_r y x^{s_r} = 1 = v_{r+1} y x^{s_{r+1}} = v_r x y^{t_{r+1}} x^{s_{r+1}}$, and thus

$$y x^{s_r - s_{r+1}} = x y^{t_{r+1}}. \tag{7.1}$$

If we consider the projection of these words onto F, we see that $f^{j+(s_r-s_{r+1})i} = f^{i+t_{r+1}j}$, which implies that $i(s_r - s_{r+1} - 1) = j(t_{r+1} - 1)$ and thus that

$$|j|(1 - t_{r+1}) = |i|(s_r - 1 - s_{r+1}).$$

Since $s_{r+1} \leqslant s_r - 1$, both sides of this equation must be non-negative, and thus we must have $t_{r+1} \leqslant 1$. We have already shown that $t_{r+1} \neq 0$, so the only possibility is $t_{r+1} = 1$, and hence $|i|(s_r - 1 - s_{r+1}) = 0$ and $s_{r+1} = s_r - 1$. Substituting these values into (7.1) gives us $yx = xy$, a contradiction. We have shown that we cannot have $s_r = s_{r+1}$ or $s_r > s_{r+1}$, so condition ii is satisfied.

By condition v for v_r, the word v_{r+1} we have constructed is of the form

$$v_r x y^{t_{r+1}-1} \equiv y^{t_0-1} x y^{t_1-1} x \ldots x y^{t_r-1} x y^{t_{r+1}-1}.$$

By condition iii for v_r and the fact that x and y do not commute, the only subwords of $v_{r+1}y$ which can be equal to the identity are those of the form vxy^k, where $v_r \equiv uv$, v is non-empty and $0 \leqslant k \leqslant t_{r+1}$. If such a word vxy^k is equal to the identity, then $u = uvxy^k \equiv v_rxy^k$ and hence $uy^{t_{r+1}-k} = v_rxy^{t_{r+1}} = x^{-s_{r+1}}$. Now let r_u be the least n such that u is a prefix of v_n. We know that u is a prefix of v_r, so r_u must be less than or equal to r.

If $r_u > 0$ then we have $u \equiv v_{r_u-1}xy^l$, for some $l \geqslant 0$. If $r_u = 0$, then $u \equiv y^l$ for some $0 \leqslant l < t_0$. In either case,

$$x^{-s_{r_u}} = v_{r_u}y \equiv uy^{t_{r_u}-l} = v_rxy^{k+t_{r_u}-l} \tag{7.2}$$

and $k + t_{r_u} - l \geqslant t_{r+1}$ by the minimality of t_{r+1}.

On the other hand, $x^{-s_{r+1}} = v_rxy^{t_{r+1}}$ which together with (7.2) implies that

$$x^{-s_{r+1}}y^{-t_{r+1}} = v_rx = x^{-s_{r_u}}y^{-(k+t_{r_u}-l)}.$$

Since $s_{r+1} > s_{r_u}$ by ii, we have $t_{r+1} > k+t_{r_u}-l$, which contradicts the earlier inequality. We have thus shown that v_{r+1} satisfies condition iii.

Case 2.2 $|i| = |j|$ *and* $x^2y^2 \neq 1$.

Note that if $|i| = |j|$ then we must have $|i| = |j| = 1$ or we would not have a word in X^* representing f.

We again construct a sequence of words v_r which satisfies the properties i to iv, and which also satisfies a new condition v:

v. $v_r \equiv y^{t_0-1}xy^{t_1-1}x\ldots xy^{t_r-1}$ with all $t_k > 2$ (so that $v_r \equiv v_{r-1}xy^{t_r-1}$ for $r > 0$).

Let t_0 be the least k such that $x^ky^k = 1$ (by assumption t_0 is greater than two) and let $v_0 \equiv y^{t_0-1}$. It is clear that v_0 satisfies conditions i to v with $s_0 = t_0$. Assume that we have constructed v_0 to v_r which satisfy conditions i to v.

If there is no $k \geqslant 0$ such that v_rxy^k is equal to a negative power of x then, just as in Case 2.1, an application of Lemma 7.3.2 shows that $I_X^m(G)$ is infinite.

We may therefore assume that there is some $k \geqslant 0$ such that v_rxy^k is equal to a negative power of x. Let t_{r+1} be the least such k, and let v_{r+1} be $v_rxy^{t_{r+1}-1}$. In order to do this, we need to know that t_{r+1} is not equal to zero. If t_{r+1} were equal to zero then we would have $v_rx = x^{-p}$ and thus $v_r = x^{-p-1}$, for some $p > 0$, contradicting condition iv for v_r. Our construction clearly implies that conditions i and iv are satisfied by v_{r+1}.

Since a word in X can be equal to the identity only if it contains the same number of instances of x as it does instances of y it is easy to see that $s_{r+1} = s_r + t_{r+1} - 2$. If t_{r+1} were equal to one then we would have $s_r = s_{r+1} + 1$. Now $v_rxy \equiv v_{r+1}y = x^{-s_{r+1}}$ and $v_ry = x^{-s_r} \equiv x^{-s_{r+1}-1}$. Hence $v_rxy = x^{-s_{r+1}} = v_ryx$, which implies that $xy = yx$, a contradiction. If t_{r+1} were equal to two then we would have $s_r = s_{r+1}$. Now $v_rxy^2 \equiv v_{r+1}y = x^{-s_{r+1}} \equiv x^{-s_r} = v_ry$, which implies that $xy = 1$, a contradiction. We can therefore say that t_{r+1} is greater than two: this, and the fact that v_r satisfies condition v, imply that v_{r+1} satisfies condition v. Since $s_{r+1} = s_r + t_{r+1} - 2$ and $t_{r+1} > 2$, we see that $s_{r+1} > s_r$ and v_{r+1} also satisfies condition ii.

The only condition we have left to check is iii. By condition v the word v_{r+1} that we have constructed is of the form

$$v_{r+1} \equiv v_r x y^{t_{r+1}-1} \equiv y^{t_0-1} x y^{t_1-1} x \ldots x y^{t_r-1} x y^{t_{r+1}-1},$$

where for $0 \leqslant n \leqslant r+1$ we know that $t_n - 1 \geqslant 2$. By the inductive hypothesis we know that no subword of v_r is equal to the identity, so the only words we need to consider to show that $v_{r+1}y$ has no non-empty proper subwords which are equal to the identity are those of the form vxy^l, where v is a non-empty suffix of v_r and $0 \leqslant l \leqslant t_{r+1}$. Any proper subword of $v_{r+1}y$ which is equal to the identity must contain the same number of instances of x as it does instances of y. Since $t_n - 1 \geqslant 2$ for $0 \leqslant n \leqslant r+1$, the only possibility is $xy^{t_r-1}x = 1$, with $t_r = 3$, which would imply that $x^2y^2 = 1$, a contradiction.

Case 2.3 $x^2y^2 = 1$.

Let $w_r \equiv x^r yxy^r$. We shall show that $w_r \neq 1$ for all $r \geqslant 0$. Given this, since any non-empty subword of w_r which is in H must be equal to w_n for some $n \leqslant r$, no w_r has a non-empty subword equal to the identity. Clearly $w_r \in H$ for every $r \geqslant 0$, and we may apply Lemma 7.3.3 with $U = \{w_r : r \geqslant 0\}$ and $S = H$ to show that $I_X^m(G)$ is infinite.

If $w_0 \equiv yx = 1$ then x and y commute, a contradiction. We know that $x^2y^2 = 1$, and thus $xyyx = 1$. If $w_1 \equiv xyxy = 1$ then $xy = yx$, and again, x and y commute. Assume that $r \geqslant 0$ is minimal such that $x^r yxy^r = 1$ (we have shown that $r \geqslant 2$). Since $x^2y^2 = 1$, we have $x^{r-2}yxy^{r-2} = 1$, contradicting the minimality of r and thus there is no $r \geqslant 0$ such that $x^r yxy^r = 1$.

Case 3 $|P| = 1$, $|N| \geqslant 2$, $Z = \emptyset$.

Let $P = \{x\}$, with $x\phi = f^i$ and $i > 0$. We construct inductively a sequence $(w_r : r \geqslant 0)$ of words such that no prefix of w_r is equal to a negative power of x. Let w_0 be a single letter word consisting of an element of N which is not equal to a negative power of x. Such an element must exist, since if each element of N is equal to a power of x then G must be generated by the set $\{x, x^{-1}\}$, which clearly cannot be the case. Let w_{r+1} be $w_r y$ where y is an element of N such that $x^l w_r y \neq 1$ for all $l \geqslant 0$. Such a y must exist, since there are at least two elements in N, and if y_1 and y_2 are elements of N, and l_1 and l_2 are natural numbers, with $l_2 \geqslant l_1$, such that $x^{l_1} w_r y_1 = x^{l_2} w_r y_2 = 1$, then $y_1 x^{l_1} w_r = y_2 x^{l_2} w_r$, and thus $y_1 = y_2 x^{l_2 - l_1}$, and y_1 is redundant. There can therefore be only one element y' of N such that $x^l w_r y' = 1$ for some $l \geqslant 0$, and the other elements of N must be suitable candidates for y. Note that $w_r \in N^*$, so that w_r is simple.

For each w_r, let $t_r \geqslant 0$ be minimal such that $(x^{t_r} w_r)\phi = f^j$ with $j \geqslant 0$. Clearly each word $x^{t_r} w_r$ is simple, and for any $r \geqslant 0$ we have $x^{t_r} w_r \in f^k H$ with $i > k \geqslant 0$, and we may apply Lemma 7.3.3 with $S = \bigcup_{0 \leqslant k < i} f^k H$ to settle this case.

Case 4 $|P| \geqslant 2$, $|N| = 1$, $Z = \emptyset$.

This case is similar to Case 3.

Case 5 $|P| \geqslant 2$, $|N| \geqslant 2$.

We construct words w_r inductively as follows; each w_r will be of the form uv with $u \in P^*$ and $v \in N^*$. Let $w_0 \equiv x$, for some $x \in P$. If $w_r\phi = f^i$ with $i \geqslant 0$, then let $w_{r+1} \equiv w_r y$ for some $y \in N$ such that $w_r y$ has no suffix equal to 1. If $w_r\phi = f^i$ with $i < 0$, then let $w_{r+1} \equiv x w_r$ for some $x \in P$ such that $x w_r$ has no prefix equal to 1. We must now check that we can do this. If y_1 and y_2 are distinct elements of N such that $vy_1 = 1$ for some suffix v of w_r, and $uvy_2 = 1$ for some suffix uv of w_r, then $y_1 = y_2 u$, where $u \in P^*$, and y_1 is redundant. The situation is similar when we prefix w_r by an element of P.

We now have an infinite set $\{w_r : r \geqslant 0\}$ of words, none of which has a subword equal to the identity, and each of which is equal to an element of the finite set $S = \bigcup_{|l| \leqslant M} f^l H$, where M is minimal such that $X \subseteq \bigcup_{|l| \leqslant M} f^l H$. We may therefore apply Lemma 7.3.3.

This case completes the proof of Proposition 7.3.4. $\square$

We shall need the following observation; for completeness we include a proof.

Lemma 7.3.5 *If G is a virtually free group containing an infinite cyclic central subgroup Z, then G is a semi-direct product $H \rtimes F$ where H is a finite normal subgroup of G and F is an infinite cyclic subgroup.*

Proof. By definition, G contains a free subgroup F of finite index. Since G is infinite, F is non-trivial.

If F is non-cyclic, then $Z(F) = 1$, so that $F \cap Z = 1$ (as $F \cap Z \leqslant Z(F)$). But then G contains the subgroup $F \times Z$, contradicting the fact that F has finite index in G. So F is infinite cyclic and then G is virtually cyclic.

By Theorem 5.4 of [14], G has a finite normal subgroup H such that G/H is isomorphic to C_∞ or to $C_2 * C_2$. In the latter case, G/H would have trivial centre, contradicting the fact that ZH/H must be an infinite central subgroup of G/H. So G/H is isomorphic to C_∞.

Choose a to be an element of infinite order in G such that $G/H = \langle aH \rangle$; thus $G = \langle H, a \rangle$. Let $F = \langle a \rangle$, so that $G = HF$. Since F has no non-trivial finite subgroups, we have that $H \cap F = 1$ and so G is the semi-direct product $H \rtimes F$ as required. $\square$

We are now in a position to prove the main result of this section:

Theorem 7.3.6 *A group G which has an infinite cyclic central subgroup can be presented by a finite special $[\lambda]$-confluent string-rewriting system if and only if $G = H \times F$, where H is finite cyclic and F is infinite cyclic.*

Proof. A string-rewriting system is said to be *monadic* if every rule is of the form $u \to v$, where $|u| > |v|$, and $|v| \leqslant 1$. It is shown in [20] that the groups which can be presented by finite monadic $[\lambda]$-confluent string-rewriting systems are exactly the virtually free groups, so any group which can be presented by a finite special $[\lambda]$-confluent string-rewriting system is certainly virtually free.

If G can be presented by a finite special $[\lambda]$-confluent string-rewriting system then, by Lemma 7.3.5, G is of the form $H \rtimes F$ where H is a finite normal subgroup of G and F is infinite cyclic. By Proposition 7.3.4, G is the direct product of a finite and an infinite cyclic group.

The converse follows directly from Proposition 7.3.4. $\square$

7.4 Further Examples

In this section we shall give several more examples of groups which may be presented by finite special $[\lambda]$-confluent string-rewriting systems. The following example shows that there are non-abelian groups which are not plain and which can be presented by such a system.

Example 7.4.1 *The group $C_{2k} *_{C_k} C_{2k}$ can be presented by a finite special $[\lambda]$-confluent string-rewriting system.*

Proof. The Cayley graph of $C_{2k} *_{C_k} C_{2k} = \langle a, b : a^{2k}, a^2 b^{-2} \rangle$ with respect to the monoid generating set $\{a, b\}$ (pictured in Figure 7.2) is isomorphic as an unlabelled graph to the Cayley graph of $C_\infty \times C_k = \langle f : \ \rangle \times \langle a : a^k \rangle$ with respect to $\{f, g\}$, where $g = af^{-1}$ (pictured in Figure 7.1). The number of closed simple loops through any point in the two graphs must therefore be the same. $\square$

The next proposition gives us a way of constructing a new group which can be presented by a finite special $[\lambda]$-confluent string-rewriting system from

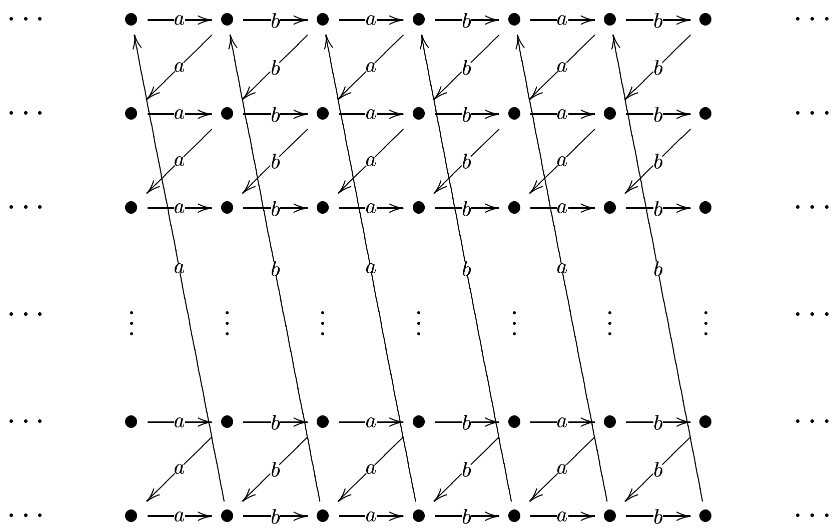

Figure 7.2: The Cayley graph of $C_{2k} *_{C_k} C_{2k} = \langle a, b : a^{2k}, a^2 b^{-2} \rangle$ with respect to the monoid generating set $\{a, b\}$.

a group which has finite irreducible word problem with respect to a monoid generating set which contains a generator of order two.

Proposition 7.4.2 *Let G be a group with a subgroup $C = \langle c \rangle$ of order two and suppose that $I_X^m(G)$ is finite and that $c \in X$. Let D be a cyclic group $\langle d \rangle$ of order four and let $P = G *_C D$ (where c is identified with d^2). Then $I_{X \cup \{d\}}^m(P)$ is finite.*

Proof. Let the word w represent a simple loop in the Cayley graph of P with respect to $X \cup \{d\}$. If the only symbol in w is d then clearly $|w| \leqslant 4$. If w contains any letters other than d then we may assume, by taking a cyclic permutation of w if necessary, that the last letter of w is not d.

Now, if w contains a subword of the form dd then the word obtained by replacing an occurrence of dd by c in w still is a simple loop. Applying similar replacements to the resulting word we eventually arrive at a word v with length at least half that of w and such that v does not contain any subwords of the form dd.

If d occurs in v then v has the form $du_1du_2\dots du_n$, where the u_i are non-empty words over X. By the normal form theorem for amalgamated free products at least one of u_i has to satisfy $u_i \in C$ because $v = 1$. As v is a simple loop, $u_i \neq 1$. Hence $u_i = c$ and therefore $du_id = 1$. Since v is simple, this implies $n = 1$. But $du_1 \neq 1$; thus d does not occur in v. So $v \in X^*$ and therefore $|v|$ is bounded, as is $|w| \leqslant 2|v|$. □

We give one further set of examples of groups with finite irreducible word problem. While the generator c in the following example is redundant, its presence both brings the situation under the scope of Proposition 7.4.2 and simplifies the proof.

Example 7.4.3 *The direct product $P = \langle c : c^2\rangle \times \langle f_1, f_2, \dots, f_n : \ \rangle \simeq C_2 \times F_n$ has finite irreducible word problem with respect to the monoid generating set $\{f_1, \dots, f_n, c, g_1, \dots, g_n\}$ where $g_i = {f_i}^{-1}c$.*

Proof. Let w be (the label of) a simple loop in the Cayley graph of P. Replace each occurrence of g_if_i and f_ig_i in w by c and call the resulting word v. Since $g_if_i = f_ig_i = c$, v is a simple loop. Observe that v has no proper subwords of the form cc for otherwise it cannot be simple.

Consider the word $\tilde{v}$ obtained by deleting from v all occurrences of c and replacing each occurrence of g_i by the symbol ${f_i}^{-1}$. Since $v \mapsto \tilde{v}$ corresponds to the projection $C_2 \times F_n \to F_n$ and $v = 1$, we must have $\tilde{v} = 1$. If $\tilde{v}$ is empty then $v \equiv cc$ so $|w| \leqslant 2|v| = 4$. Otherwise $\tilde{v}$ must contain a subword of the form $f_i{f_i}^{-1}$ or ${f_i}^{-1}f_i$. Since neither g_if_i nor f_ig_i nor cc can occur in v as (proper) subwords, a subword of the form f_icg_i or g_icf_i does. However $f_icg_i = g_icf_i = 1$ which, in view of the simplicity of v implies $|v| \leqslant 3$ and thus $|w| \leqslant 6$. $\square$

Chapter 8

Infinite Irreducible Word Problems

8.1 Regular Irreducible Word Problem

Having considered groups whose irreducible word problem is a finite language, the next class of languages which we would naturally consider is the regular languages. It is an easy consequence of the pumping lemma that we must look further than this class of languages if we wish to find a wider class of groups.

Proposition 8.1.1 *Let G be a group with finite monoid generating set X. If the irreducible word problem of G with respect to X is regular then it is finite.*

Proof. Suppose that $I = I^m_X(G)$ is regular. By the Pumping Lemma (2.3.2) there is a constant N such that, if $z \in I$ with $|z| \geqslant N$, then there exist words u, v and w with $z \equiv uvw$, $|v| < N$ and $uv^iw \in I$ for all $i \in \mathbb{N}$.

If we now consider u, v and w as elements of G, we have that $uw = uvw = 1$ and hence that $v = 1$. But v is a proper subword of uvw, a contradiction. Thus any word in I has length less than N, and so there can be only finitely many of them. $\square$

8.2 Context-Free Irreducible Word Problem

Next we consider the groups which have context-free irreducible word problem with respect to some group generating set. We start with an example of a group which is in this class, but is not a plain group.

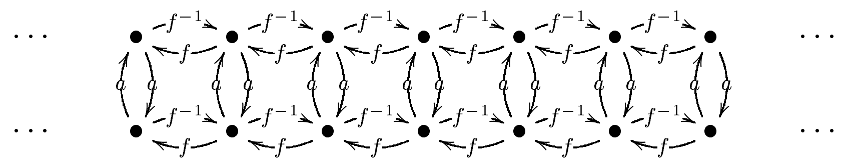

Figure 8.1: The Cayley Graph of $\langle f :\rangle \times \langle a : a^2 \rangle$ with respect to the group generating set $\{f, a\}$.

Example 8.2.1 *The irreducible word problem of the group G defined by the presentation $\langle f : \ \rangle \times \langle a : a^2 \rangle$ with respect to the group generating set $\{a, f\}$ is context-free but not finite.*

Proof. We write A for a^{-1} and F for f^{-1}; note that a and A represent the same element of G.

A word w in the symbols $\{a, A, f, F\}$ is equal to the identity in G if and only if it contains an even number of instances of elements of $\{a, A\}$ and, in addition, we have that $|w|_f = |w|_F$. Let L be the language consisting of the words of the following forms, all of which are equal to the identity in G and are simple:

- fF, Ff, aA, Aa, aa, and AA;
- $f^i b_1 F^j b_2 f^k$, where $b_1, b_2 \in \{a, A\}$ and $i + k = j$, $j > 0$, $k \geqslant 0$;
- $F^i b_1 f^j b_2 F^k$, where $b_1, b_2 \in \{a, A\}$ and $i + k = j$, $j > 0$, $k \geqslant 0$.

L is a union of context-free languages and is therefore context-free.

We shall show that any word from the word problem which contains more than two instances of elements of $\{a, A\}$ cannot be simple, and hence that L is the irreducible word problem I of G with respect to the group generating set $\{a, f\}$.

Assume, for a contradiction, that there is a word w in I of the form

$$g_1{}^{i_1} b_1 g_2{}^{i_2} b_2 g_3{}^{i_3} \dots g_{n-1}{}^{i_{n-1}} b_{n-1} g_n{}^{i_n},$$

where $n > 3$, each g_j is either f or F, each b_i is a or A, and where $i_j > 0$ for each j such that $2 \leqslant j \leqslant n - 1$. At least one of i_1 and i_n must be non-zero, otherwise, by removing b_1 and b_{n-1} from w, we would produce a proper subword of w equal to the identity.

We assume that $i_1 > 0$ and that g_1 is f (the other cases are similar). If $i_n > 0$, then g_n must also be f (otherwise we could remove the first and last symbol of w to leave a proper subword equal to the identity). We must have $g_j = g_{j+1}$ for $2 \leqslant j \leqslant n-1$, otherwise, $b_{j-1}{g_j}^{i_j}b_j{g_{j+1}}^{i_{j+1}}b_{j+1}$ would contain a proper subword equal to the identity. Thus $g_2 = g_3 = \cdots = g_{n-1} = F$. So we now have that w must be of the form

$$f^{i_1}b_1F^{i_2}b_2F^{i_3}\ldots F^{i_{n-1}}b_{n-1}f^{i_n},$$

where $i_j > 0$ for each j such that $1 \leqslant j \leqslant n-1$. We must have that $i_1 < i_2$ in order to prevent $f^{i_1}b_1F^{i_2}b_2$ from having a subword equal to the identity, and, similarly, we must have that $i_{n-1} > i_n$. From this we deduce that $i_1 + i_n < i_2 + \cdots + i_{n-1}$, a contradiction, since we must have $i_1 + i_n = i_2 + \cdots + i_{n-1}$ for w to be in the word problem. $\Box$

The previous example shows us that it is possible for a group to have irreducible word problem, with respect to some monoid generating set, which is context-free, but not finite. We already know however that the group in question has finite irreducible word problem with respect to another generating set. The question remains as to whether there is a group which has context-free irreducible word problem with respect to some monoid or group generating set which does not have finite irreducible word problem with respect to any monoid generating set.

Having shown that the plain groups are a proper subclass of the class of groups with context-free irreducible word problem for some group generating set, our next task must be to put some bound on how complex these groups can be. It turns out that, even if we allow monoid generating sets, such groups must be context-free:

Proposition 8.2.2 *If G is a group with a finite monoid generating set X, and if $I_X^m(G)$ is context-free, then $W_X^m(G)$ is context-free.*

Proof. If $I_X^m(G)$ is context-free, then $I_X^m(G) \cup \{\lambda\}$ is context-free, and the result follows from Propositions 4.1.3 and 2.8.2. $\Box$

The converse of Proposition 8.2.2 is false, as the following example shows. In fact, we will subsequently prove that all infinite context-free groups have non-context-free irreducible word problem with respect to some finite group generating set.

Example 8.2.3 *Let $G = \langle f : \ \rangle$ be the free group on one generator, and let $a = f$, $b = f^2$ and $c = f^6$. Then G has non-context-free irreducible word problem with respect to the group generating set $X = \{a, b, c\}$.*

Proof. We let $I = I_X^g(G)$ and $W = W_X^g(G)$, and let A, B and C denote a^{-1}, b^{-1} and c^{-1} respectively. Let Y be the language denoted by the regular expression $abc^*BC^*Bc^*bAC^*$ and let $L = I \cap Y$. If L can be shown not to be context-free then it follows that I is not context-free, since the class of context-free languages is closed under intersection with regular sets.

Consider a typical element $w \equiv abc^iBC^jBc^kbAC^l$ of Y. We need to find conditions which tell us exactly when such an element of Y is also in I. Firstly, we must have that $i + k = j + l$ in order that w is in W. We also need to make sure that no subword of w is in W.

Since any instance of b, B, c or C in w is equivalent to an even number of instances of f, the instance of a can only cancel out with the instance of A. So any subword of w which is equal to the identity in G and which contains the instance of a must also contain the instance of A and vice-versa, and hence must be the entire word w. So we need only consider subwords of $bc^iBC^jBc^kb$. There are not enough instances of b and B in this subword to cancel out an instance of c or C; so, in any subword which is equal to the identity, the instances of b must cancel with those of B, and the instances of c must cancel with those of C.

Hence any proper subword of $bc^iBC^jBc^kb$ which is equal to the identity must be either a subword of bc^iBC^j or a subword of C^jBc^kb; so, to make sure there are no such subwords, we must have that $i > j$, $j < k$ and that $i + k \neq j$. If $i > j$ and $j < k$ then $i + k > j$, so this last condition may be dropped, and the language L we need to consider is

$$\{abc^iBC^jBc^kbAC^l : i + k = j + l, i > j \text{ and } j < k\}.$$

Let $M = (\{q_0, q_1, q_2, q_3, q_4, q_5\}, \{a, b, c, A, B, C\}, \{x, y, z\}, \delta, q_0, \{q_5\})$ be the GSM, where

$$\begin{array}{lll} \delta(q_0, a) = \{(q_1, \lambda)\}, & \delta(q_1, b) = \{(q_2, \lambda)\}, & \delta(q_2, c) = \{(q_2, x)\}, \\ \delta(q_2, B) = \{(q_3, \lambda)\}, & \delta(q_3, C) = \{(q_3, y)\}, & \delta(q_3, B) = \{(q_4, \lambda)\}, \\ \delta(q_4, c) = \{(q_4, z)\}, & \delta(q_4, b) = \{(q_5, \lambda)\}, & \\ \delta(q, d) = \{(q_5, \lambda)\} & \text{for all other } (q, d). & \end{array}$$

The image of L under M is the language $L' = \{x^i y^j z^k : i > j \text{ and } j < k\}$. If we can prove that L' is not context-free then it follows that L is not context-free, as the class of context-free languages is closed under GSM-mappings.

We use Ogden's Lemma (Lemma 2.4.1) to prove that L' is not context-free. If L' were context-free, then there would exist a constant N such that, if we pick any word w in L' and mark N or more positions in w, then we can write w as $u_1 v_1 u_2 v_2 u_3$ in such a way that $v_1 v_2$ contains at least one marked position, $v_1 u_2 v_2$ contains at most N marked positions, and, for all $r \geqslant 0$, $u_1 {v_1}^r u_2 {v_2}^r u_3 \in L'$.

Consider the word $w \equiv x^{N+2} y^{N+1} z^{N+2}$ in L' with all the instances of y marked. In order that $w_2 \equiv u_1 {v_1}^2 u_2 {v_2}^2 u_3 \in L'$ we must have either that v_1 is a subword of x^{N+2} or y^{N+1} and v_2 is a non-empty subword of y^{N+1}, or else that v_1 is a non-empty subword of y^{N+1} and v_2 is a subword of y^{N+1} or z^{N+2}. In the first case $|w_2|_y \geqslant |w_2|_z$; in the second case $|w_2|_y \geqslant |w_2|_x$. So the word w does not satisfy the conclusion of Ogden's Lemma, and therefore L' cannot be context-free. $\square$

Given this result, we have:

Corollary 8.2.4 *Every infinite context-free group has non-context-free irreducible word problem for some finite group generating set.*

Proof. By Theorem 3.6.2 every infinite context-free group G contains the free group $\langle f : \ \rangle$ as a subgroup. Let a, b, c, A, B and C be as in Example 8.2.3, and then choose a finite group generating set Z for G which includes a, b and c. We now consider the intersection of $I^g_Z(G)$ with $\{a, b, c, A, B, C\}^*$. This set was shown not to be context-free in Example 8.2.3; since the intersection of a context-free set and a regular set is necessarily context-free, $I^g_Z(G)$ cannot be context-free. $\square$

It is interesting to note that we now have three generating sets for the group $\langle f : \ \rangle \times \langle a : a^2 \rangle$:

- a monoid generating set with respect to which the irreducible word problem is finite (Proposition 7.2.1);
- a group generating set with respect to which the irreducible word problem is context-free, but not finite (Example 8.2.1); and
- a group generating set with respect to which the irreducible word problem is not context-free (Corollary 8.2.4).

Although every infinite context-free group has a finite group generating set with respect to which its irreducible word problem is non-context-free, there is something we can say about the irreducible word problem for any monoid generating set: it must have a context-free complement.

Proposition 8.2.5 *Let G be a group with finite monoid generating set X. If $W_X^m(G)$ is context-free then the complement in X^* of $I_X^m(G)$ is context-free.*

Proof. Let $I = I_X^m(G)$ and $W = W_X^m(G)$. The complement $X^* \setminus I$ of I is the union of the complement of W and the set Y of all words which contain a proper subword which is in W. Now $Y = X^*WX^+ \cup X^+WX^*$ is context-free since the class of context-free languages is closed under concatenation and union. The complement of W is context-free since W is deterministic context-free by Theorem 3.6.2. Thus the complement of I must be context-free as it is the union of two context-free languages. $\square$

Propositions 8.2.2 and 8.2.5 together with Proposition 3.5.4 give us the following piece of information about which context-free languages can be the irreducible word problems of groups:

Corollary 8.2.6 *Let G be a group with context-free irreducible word problem for some finite monoid generating set. Then the complement of the irreducible word problem of G is context-free with respect to any finite monoid generating set.*

The problem remains of finding alternative characterizations of the class of groups which have context-free irreducible word problem with respect to some monoid or group generating set. We have not yet ruled out the possibility that every context-free group has a finite group generating set with respect to which it has context-free irreducible word problem, although this seems unlikely.

8.3 One-Counter Irreducible Word Problem

Proposition 8.2.5 and Corollary 8.2.6 carry across to the one-counter groups. For instance, we have:

Corollary 8.3.1 *Let G be a group with finite monoid generating set X. If $W_X^m(G) \in \mathcal{OC}$ then the complement in X^* of $I_X^m(G)$ is one-counter.*

On the other hand, the analogue of Proposition 8.2.2 in the one-counter case fails. The free group on two generators provides a simple example of a group which has a set of generators for which the irreducible word problem is finite, and therefore certainly one-counter, but the word problem is not one-counter. This reflects the fact that, unlike the case of the context-free languages, the insertion closure of a one-counter language need not be one-counter.

Bibliography

[1] A. V. Anisimov. *Group languages.* Kibernetica, 4(1971), pages 18–24.

[2] A. V. Anisimov. *Some algorithmic problems for groups and context-free languages.* Kibernetica, 8(1972), pages 4–11.

[3] J.-M. Autebert, L. Boasson, and G. Sénizergues. *Groups and NTS languages.* J. Comput. System Sci., 35(1987), pages 243–267.

[4] J. Avenhaus and K. Madlener. On groups defined by monadic Thue systems. In *Algebra, Combinatorics and Logic in Computer Science. Colloquia Mathematica Societatis János Bolyai 42*, pages 63–71, Györ (Hungary), 1983.

[5] J. Avenhaus, K. Madlener, and F. Otto. *Groups presented by finite two-monadic Church-Rosser Thue systems.* Trans. Amer. Math. Soc., 297(1986), pages 427–443.

[6] J. Berstel. *Transductions and Context-Free Languages.* Teubner, Stuttgart, 1979.

[7] Y. Cochet. Church–Rosser congruences on free semigroups. In *Algebraic Theory of Semigroups. Colloquia Mathematica Societatis János Bolyai 20*, pages 51–60, Szeged (Hungary), 1979.

[8] M. J. Dunwoody. *The accessibility of finitely presented groups.* Invent. Math., 81(1985), pages 449–457.

[9] R. H. Gilman. Computations with rational subsets of confluent groups. In J. Fitch, editor, *EUROSAM 84 : International Symposium on Symbolic and Algebraic Computation*, volume 174 of *Lecture Notes in Computer Science*, pages 207–212. Springer-Verlag, 1984.

[10] R. H. Haring-Smith. *Groups and simple languages.* Trans. Amer. Math. Soc., 279(1983), pages 337–356.

[11] M. A. Harrison. *Introduction to Formal Language Theory.* Addison Wesley, 1978.

[12] T. Herbst. *On a subclass of context-free groups.* RAIRO Inform. Théor., 25(1991), pages 255–272.

[13] T. Herbst. *Some remarks on a theorem of Sakarovitch.* J. Comput. System Sci., 44(1992), pages 160–165.

[14] T. Herbst and R. M. Thomas. *Group presentations, formal languages and characterizations of one-counter groups.* Theoret. Comput. Sci., 112(1993), pages 187–213.

[15] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison Wesley, 1979.

[16] J. M. Howie. *Automata and Languages.* Oxford, 1991.

[17] M. Ito, L. Kari, and G. Thierrin. *Insertion and deletion closure of languages.* Theoret. Comput. Sci., 183(1997), pages 3–19.

[18] R. C. Lyndon and P. E. Schupp. *Combinatorial Group Theory.* Springer-Verlag, 1977.

[19] K. Madlener and F. Otto. *Commutativity in groups presented by finite Church-Rosser Thue systems.* RAIRO Inform. Théor., 22(1988), pages 93–111.

[20] K. Madlener and F. Otto. *About the descriptive power of certain classes of finite string-rewriting systems.* Theoret. Comput. Sci., 67(1989), pages 143–172.

[21] D. E. Muller and P. E. Schupp. *Groups, the theory of ends, and context-free languages.* J. Comput. System Sci., 26(1983), pages 295–310.

[22] D. E. Muller and P. E. Schupp. *The theory of ends, pushdown automata, and second-order logic.* Theoret. Comput. Sci., 37(1985), pages 51–75.

[23] D. W. Parkes and R. M. Thomas. *Reduced and irreducible word problems of groups.* Technical Report 4, Department of Mathematics and Computer Science, University of Leicester, 1999.

[24] J.-F. Perrot and J. Sakarovitch. A theory of syntactic monoids for context-free languages. In B. Gilchrist, editor, *Information Processing 77*, pages 69–72. North Holland, 1977.

[25] J.-E. Pin. Syntactic semigroups. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 1, chapter 10, pages 679–746. Springer-Verlag, 1997.

[26] J. Sakarovitch. Monoïdes syntactiques et langages algébriques. Thèse 3ème cycle math., Univ. Paris VII, Paris, 1976.

[27] J. Sakarovitch. *An algebraic framework for the study of the syntactic monoids: application to the group languages.* Lecture Notes in Computer Science, 45(1977), pages 510–516.

[28] J. Sakarovitch. Sur les groupes infinis, consideres comme monoïdes syntaxiques de langages formels. In *Séminaire P. Dubreil 1975/6*, number 586 in Lecture Notes in Mathematics. Springer-Verlag, 1977.

[29] J. Sakarovitch. *Sur une propriété d'itération des langages algébriques déterministes.* Math. Systems Theory, 14(1981), pages 247–288.