ON PLAUSIBLE COUNTEREXAMPLES TO LEHNERT'S CONJECTURE

Daniel Bennett

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

2018

On Plausible Counterexamples to Lehnert's Conjecture

Daniel Bennett

University of
St Andrews

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

12th December 2017

Declarations

Candidate's declarations

I, Daniel Bennett, hereby certify that this thesis, which is approximately 41000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for a higher degree.

I was admitted as a research student in September 2013 and as a candidate for the degree of PhD Mathematics in September 2013; the higher study for which this is a record was carried out in the University of St Andrews between 2013 and 2017.

I received funding from an organisation or institution and have acknowledged the funder(s) in the full text of my thesis.

Date: **Signature of candidate:**

Supervisor's declarations

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

Date: **Signature of supervisor:**

Date: **Signature of supervisor:**

Permission for publication

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that my thesis will be electronically accessible for personal or research use unless exempt by award of an embargo as requested below, and that the library has the right to migrate my thesis into new electronic forms as required to ensure continued access to the thesis. I have obtained any third-party copyright permissions that may be required in order to allow such access and migration, or have requested the appropriate embargo below.

The following is an agreed request by candidate and supervisors regarding the publication of this thesis:

PRINTED COPY: No embargo on print copy

ELECTRONIC COPY: No embargo on electronic copy

Date: **Signature of candidate:**

Date: **Signature of supervisor:**

Date: **Signature of supervisor:**

Acknowledgements

To my wife Rachel, over the past four years we began our relationship, got engaged and eventually married. The support and love you have shown through all that time is without parallel.

Special thanks goes to my supervisor Collin Bleak, without whom I never would have applied to do a PhD! It was fantastic working with you. Thank you for your patience, support and friendship over these past years.

I am grateful to all my family, especially Mum and Dad, for the moral and loving support they constantly give. This would have been impossible without all of you.

To all the folk who have been at Tayport Gospel Hall over the past eight years. It has been like a home from home. Your christian love and support has made my time up here so special.

I am grateful to all my office mates and colleagues who provided a great place to work and an even better place to solve cryptic crosswords.

Finally, and most importantly, all thanks goes to my Lord and Saviour Jesus Christ, without whom I would have, and be, nothing. Soli Deo gloria.

This work was supported by the EPSRC.

Abstract

A group whose co-word problem is a context free language is called $co\mathcal{CF}$. Lehnert's conjecture states that a group G is $co\mathcal{CF}$ if and only if G embeds as a finitely generated subgroup of R. Thompson's group V. In this thesis we explore a class of groups, $\mathfrak{F}_{aug}$, proposed by Berns-Zieze, Fry, Gillings, Hoganson, and Mathews to contain potential counterexamples to Lehnert's conjecture. We create infinite and finite presentations for such groups and go on to prove that a certain subclass of $\mathfrak{F}_{aug}$ consists of groups that do embed into V.

By Anisimov a group has regular word problem if and only if it is finite. It is also known that a group G is finite if and only if there exists an embedding of G into V such that its natural action on $\mathfrak{C}_2 := \{0,1\}^\omega$ is free on the whole space. We show that the class of groups with a context free word problem, the class of $\mathcal{CF}$ groups, is precisely the class of finitely generated demonstrable groups for V. A demonstrable group for V is a group G which is isomorphic to a subgroup in V whose natural action on $\mathfrak{C}_2$ acts freely on an open subset. Thus our result extends the correspondence between language theoretic properties of groups and dynamical properties of subgroups of V. Additionally, our result also shows that the final condition of the four known closure properties of the class of $co\mathcal{CF}$ groups also holds for the set of finitely generated subgroups of V.

Contents

Declarations **ii**

Acknowledgements **iv**

Abstract **v**

1 Introduction **2**

1.1 Definitions . . . 3
1.2 Background . . . 5
1.3 Results and Implications . . . 8

2 Background and Definitions **11**

2.1 Formal Languages . . . 11
2.2 Group presentations . . . 17
2.3 R. Thompson's groups . . . 19

3 The Demonstrable Groups for V **41**

3.1 Statement of results . . . 42
3.2 Demonstrable groups . . . 44
3.3 The countable virtually free groups are in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$ . . . 46
3.4 Finitely generated groups in $\mathfrak{D}_{V,\mathfrak{C}_2}$ are virtually free . . . 51

4 Presentations for Groups in $\mathfrak{F}_{aug}$ **57**

4.1 Statement of results . . . 59
4.2 The groups $\mathfrak{V}_{aug}$ . . . 59
4.3 An infinite presentation for $F_{(G,\theta)}$ . . . 74
4.4 A finite presentation for $F_{(G,\theta)}$ . . . 104

5 Embedding Groups from $\mathfrak{F}_{aug}$ into V **125**

5.1 Statement of results . . . 125
5.2 Isomorphism result . . . 126
5.3 Embedding results . . . 134
5.4 Future research . . . 138

Chapter 1

Introduction

The subject of this thesis lies in the intersection between formal language theory and group theory, an area of study which has a long history in mathematics. The origins of this intersection lie in combinatorial group theory, which began with the first systematic study of the matter by Walther Von Dyck in 1882 [32]. Thus began the concept of a group presentation, an abstract way of defining a group G using a set of generators X and a set of relators R. In 1911 Max Dehn introduced several formal questions that could be asked in association with group presentations [16]. One such is the *Word Problem* which asks, for a group G with finite generating set X, whether there exists a algorithm for determining in finite time whether or not any given finite product of generators of G is equal to the identity. It is the concept of the word problem which is at the foundation of the work in this thesis.

From the word problem of a group G with finite generating set X, there naturally arises a formal language over the alphabet $X^{\pm} := X \sqcup X^{-1}$, which we call $WP(G, X)$. Informally, the language $WP(G, X)$ is defined to be the set of strings from $(X^{\pm})^*$ which, under product in G, equal the identity of G. Therefore, the word problem is asking whether there exists an algorithm that can, in finite time, determine whether or not a word from the language $(X^{\pm})^*$ is contained in $WP(G, X)$. In an abuse of terminology, we will call $WP(G, X)$ the word problem of G with respect to X. Formal languages can be classified by their complexity and it is known that for some of these language classes the class of the word problem of a finitely generated group G is independent of the choice of finite generating set (see [20]). Thus one can introduce a well defined classification of the finitely generated groups by the language class of their respective word problems. Conversely, for a group G with finite generating set X one can also consider the complement of its word problem, its *co-word* problem. In this thesis we focus on the class of groups with a context free co-word problem, which we call the *co*$\mathcal{CF}$ groups.

The main motivation for our work is to answer a conjecture by Jörg Lehnert in [22]. After applying the work of [7] the conjecture states that a group G is in the class of $co\mathcal{CF}$ groups if and only if G embeds as a finitely generated subgroup of R. Thompson's group V. The authors of [3] introduce a class of groups, which we call $\mathfrak{V}_{aug}$, which they suggest may contain counterexamples to Lehnert's conjecture. In this thesis we study a class of groups $\mathfrak{F}_{aug}$, related to those in $\mathfrak{V}_{aug}$. We produce infinite and finite presentations for the groups in $\mathfrak{F}_{aug}$, and show that a subclass of these groups embed into R. Thompson's group V. Ultimately we are unable to find a counterexample, or find a resolution to Lehnert's conjecture. However, we see no evidence to suggest that the groups from $\mathfrak{F}_{aug}$ or $\mathfrak{V}_{aug}$ will not contain a counterexample to the conjecture, and continue to propose these groups as an area of study in this regard.

This thesis will take the following outline. In Chapter 2 we give a preliminary introduction to the core areas and objects of study contained in the thesis. In Chapter 3 we will positively answer a question posed in [3] and in doing so further explore and establish the correspondence between language theoretic properties of groups and dynamical properties of the subgroups of R. Thompson's group V. In Chapter 4 we will construct infinite and finite presentations for all groups in the class $\mathfrak{F}_{aug}$. In Chapter 5 we will use the presentations from Chapter 4 to prove that groups from a certain subclass of $\mathfrak{F}_{aug}$ do embed into R. Thompson's group V.

The remainder of the introduction divides into three sections. We first provide basic definitions so that the content which follows may be understood. We go on to give a brief, and non-comprehensive, history of the work that has already been done in the research area surrounding this thesis, including important results which motivated our work. In the final section we expand on the outline of the thesis given in the previous paragraph, stating the main results and explaining how they fit into, and build on, the current knowledge of the subject area.

1.1 Definitions

Formal languages

Let Λ be a set of symbols, which we call an **alphabet**. A **letter** from the alphabet Λ is a symbol a in Λ. A **word** over an alphabet Λ is a sequence, or string, of letters from Λ. We set Λ^* to be the set of all finite words over the alphabet Λ, including the empty word which we will denote by ε. Note that Λ^* is also a monoid under the binary operation of concatenation of strings.

A **formal language L over an alphabet** Λ is a subset of Λ^*. Formal languages are known to exist in different classes, differing by the "complexity" of the languages contained within each class. We have not yet defined what we mean by the "complexity" of a language, but it sufficient for this stage of our discussion to understand that such a concept exists and can be formally defined (see, for example, [21]). One of the first classifications of formal languages was given by Noam Chomsky in [14]. From his work we have the **Chomsky hierarchy**, a collection of four classes of languages in which the classes of lower complexity are contained within those of higher complexity. The four classes and the respective containments are given below.

$$\text{Regular} \quad \subset \quad \text{Context Free} \quad \subset \quad \text{Context Sensitive} \quad \subset \quad \text{Recursively Enumerable}$$

We say that a class $\mathcal{C}$ of languages is **closed under inverse homomorphisms** if, for any finite alphabets A and B, and any monoid homomorphism $\theta : A^* \to B^*$, we have L in $\mathcal{C}$ where $L \subset B^*$, then $(L)\theta^{-1} \subset A^*$ is also in $\mathcal{C}$. It is a well known result that regular, context free ($\mathcal{CF}$), context sensitive and recursively enumerable languages are all closed under inverse homomorphisms.

Group presentations

Let G be a group with generating set A. Let X be a set of symbols such that the map $X \to A$ is a bijection of sets. Let F_X be the free group with basis X. The map $X \to A$ extends to a surjective group homomorphism $\phi : F_X \to G$, and thus, by the First Isomorphism Theorem, $G \cong F_X/ker(\phi)$. Suppose R is a subset of F_X such that $ker(\phi)$ is the normal closure of R, that is, $ker(\phi)$ the smallest normal subgroup of F_X containing R. Then the expression $\langle X|R\rangle$ uniquely determines the group G up to isomorphism; we call it a **presentation** for G. We call X a set of **generators** and R a set of **relators**.

Word problem

Let $X^{-1} := \{x^{-1} : x \in X\}$. By the map $\phi : F_X \to G$ above one can identify words over the alphabet $X^{\pm} := X \sqcup X^{-1}$ to elements in G. The language of the word problem of G with respect to X, or $WP(G, X)$, is defined as

$$WP(G, X) := \{w : w \text{ is in } (X^{\pm})^* \text{ such that } w =_G 1\}.$$

By $w =_G 1$ we mean that the word w is equivalent to the identity in the group G. That is, $w =_G 1$ if and only if w is in the coset of F_X/N containing the empty word. Often we abuse our terminology and shorten the description of $WP(G, X)$ to simply the *word problem of G with respect to X.*

The definition of $WP(G, X)$ depends on the set of generators of the group G. However, there exists the following well known result (see [20]).

Proposition 1.1. *Let G_1 and G_2 be two isomorphic groups defined by presentations $\langle X|R\rangle$ and $\langle Y|S\rangle$ respectfully, where X and Y are both finite. If $\mathcal{C}$ is a language class closed under inverse homomorphisms then $WP(G_1, X) \in \mathcal{C}$ if and only if $WP(G_2, Y) \in \mathcal{C}$.*

The proposition states that for a group G the language class of $WP(G, X)$ is independent of our choice of finite generating set X, provided that language is closed under inverse homomorphisms. As stated in the previous section, those languages in the Chomsky hierarchy are known to have this property, and for the sake of this work, those are the only languages that we will be considering. Thus, it is acceptable to say that a finitely generated group G has word problem in the class of languages $\mathcal{C}$, without referring to a specific generating set. Therefore we will use notation such as $WP(G) \in \mathcal{C}$. If $WP(G) \in \mathcal{C}$ for some class of languages $\mathcal{C}$, then we will refer to G as a "$\mathcal{C}$-group". For example, if $WP(G)$ is in the class of context free languages, we would say that "G is a $\mathcal{CF}$ group", or equivalently, "G is $\mathcal{CF}$".

1.2 Background

Regular and context free groups

As we previously mentioned, we can define a class of groups by the language class of the word problem. For example, a group G, for which $WP(G)$ is a context free language is in the class of *context free*, or $\mathcal{CF}$, groups. The question arises, for a class of languages $\mathcal{C}$, can the class of $\mathcal{C}$-groups be defined by purely group theoretic properties? In [1] Anatoly Anisimov proves the following theorem.

Theorem 1.2 (Anisimov)**.** *A finitely generated group G is a regular group if and only if it is finite.*

Put another way, this result states that the class of regular groups is exactly the class of finite groups. Subsequently, in [24] and [25], David Muller and Paul Schupp prove the following result for $\mathcal{CF}$ groups (relying on the work of Dunwoody in [18]).

Theorem 1.3 (Muller, Schupp)**.** *A finitely generated group G is a $\mathcal{CF}$ group if and only if it is virtually free.*

The co-context free groups

Let G be a group defined by the presentation $\langle X|R\rangle$. The co-word problem of G with respect to X, otherwise called $coWP(G, X)$, is the complement of $WP(G, X)$, formally defined as

$$coWP(G, X) := \{w : w \in X^{\pm} \text{ such that } w \neq_G 1_G\}.$$

As with $WP(G, X)$, the language class of $coWP(G, X)$ is independent of the generating set X.

Introduced by Holt, Rees, Röver and Thomas [20], the next class of groups we consider consists of all the groups whose *co-word* problem is context free, the $co\mathcal{CF}$ groups. The class of context free languages is not closed under complementation. Nevertheless, it is known that if a group G is virtually free (a group containing a finite index free subgroup) then $WP(G)$ is a *deterministic* context free language, (the definition of which we give in Chapter 2). Since the class of deterministic context free languages *is* closed under complementation, the class of $\mathcal{CF}$ groups is contained within the class of $co\mathcal{CF}$ groups. The containment is also strict, there do exist $co\mathcal{CF}$ groups which do not have a context free word problem, for example $\mathbb{Z} \times \mathbb{Z}$. From the perspective of formal language theory the class of $co\mathcal{CF}$ groups is a natural broadening of the class of $\mathcal{CF}$ groups, a "next step" in widening the class.

The authors of [20] prove a number of closure properties of the $co\mathcal{CF}$ groups,

1. passing to finitely generated subgroups,
2. passing to finite index overgroups,
3. taking finite direct products of $co\mathcal{CF}$ groups,
4. taking the restricted wreath product $G \wr H$, where G is a $co\mathcal{CF}$ group and H is a $\mathcal{CF}$ group.

However, there currently does not exist an algebraic classification of the whole class as there does for the classes of regular and $\mathcal{CF}$ groups. One of the motivations of this work was to explore a possible way in which these groups could be classified. To understand the potential classification we are referring to, we first need to introduce an infinite, finitely presented group, upon which the classification is based; R. Thompson's group V.

R. Thompson's group V

In 1965 [31] Richard Thompson introduces three infinite, finitely presented groups, which today are referred to by F, T and V. In unpublished notes [31], Thompson proves that T and V are simple, the first examples of infinite, finitely presented, simple groups. (F is *not* simple, however its derived subgroup $[F, F]$ is.) The R. Thompson groups have been widely researched since their introduction and used in a variety of different mathematical areas. For example, specific to V we can point the reader to publications [8, 9, 3, 4, 7, 17, 28], a by no means comprehensive list. The groups are often defined as groups of homeomorphisms of different spaces. Thompson's group F can be given as a group of homeomorphisms of the unit interval $[0, 1]$, T can be given as a group of homeomorphisms of the unit circle S^1, and V can be given as a group of homeomorphisms of the Cantor space $\mathfrak{C}_2$. Not apparent from these definitions though, it also happens that the groups are related to each other by natural containments, namely $F < T < V$. In [23] Lehnert and Schweitzer prove the following theorem.

Theorem 1 of [23] *The Higman-Thompson groups $G_{n,r}$ are co$\mathcal{CF}$.*

As Thompson's group V is isomorphic to the Higman-Thompson group $G_{2,1}$ a corollary of their result is that F, T and V are *co*$\mathcal{CF}$ groups. (Recall that the class of *co*$\mathcal{CF}$ groups are closed under passing to finitely generated subgroups.)

Lehnert's conjecture

Let $\mathcal{T}_{2,c}$ be the infinite binary 2-coloured tree, and let $QAut(\mathcal{T}_{2,c})$ be the group of all bijections on the vertices of $\mathcal{T}_{2,c}$ that respect adjacency and the edge-colour relation, except for, possibly, finitely many edges. In his dissertation [22], Lehnert proves the following theorem.

Theorem 1.4 (Lehnert)**.** *The group $QAut(\mathcal{T}_{2,c})$ is co$\mathcal{CF}$ and there exists an embedding of Thompson's group V into $QAut(\mathcal{T}_{2,c})$ as a subgroup.*

Also in [22], he conjectures that $QAut(\mathcal{T}_{2,c})$ is a **universal** *co*$\mathcal{CF}$ group. That is, he conjectured that a group G is *co*$\mathcal{CF}$ if and only if G is finitely generated and embeds in $QAut(\mathcal{T}_{2,c})$. If the conjecture were true then it would provide a purely algebraic classification of the class *co*$\mathcal{CF}$.

In [7], Bleak, Matucci and Neunhöffer prove the converse embedding result of Theorem 1.4.

Theorem 1.5 (Bleak, Matucci and Neunhöffer)**.** *The group $QAut(\mathcal{T}_{2,c})$ embeds into Thompson's group V as a subgroup.*

Thus, the authors of [7] prove that $QAut(\mathcal{T}_{2,c})$ and Thompson's group V are bi-embeddable. This means we can rephrase Lehnert's conjecture in the following way.

Conjecture 1.6 (Lehnert's conjecture)**.** *Thompson's group V is a universal co$\mathcal{CF}$ group.*

Lehnert's conjecture is still open at the time of writing, and the motivation for this thesis is to investigate potential counterexamples to the conjecture.

1.3 Results and Implications

Languages and dynamics

In R. Thompson's group V one can find embeddings of finite groups that have interesting dynamical properties. Let G be a finitely generated group. It was noted by Collin Bleak that the group G is finite if and only if it admits an embedding as a subgroup of V with a free action on the Cantor set [6]. By Theorem 1.2 the class of regular groups is exactly the class of finite groups. Thus we have a first correspondence between formal language theoretic properties of groups and dynamical properties of subgroups of Thompson's group V. In Chapter 3 we investigate this correspondence further. (Note that the work done in Chapter 3 is joint with Collin Bleak, and much of the content can be found in a co-authored paper [2].)

The initial motivation for the work in the third chapter was to answer a question posed by the authors of [3].

Question 1.2 of [3] *Does there exist a demonstrative embedding of the free group on two generators, F_2, into Thompson's group V?*

Informally, a finitely generated group G is a demonstrative subgroup if and only if it freely acts on an *open subset* of $\mathfrak{C}_2$. In this chapter we positively answer the question above and go on to prove a deeper result.

Theorem 1.7. *A finitely generated group G is $\mathcal{CF}$ if and only if it is a demonstrable group for Thompson's group V.*

This result provides another correspondence between formal language theoretic properties of groups and the dynamics of the subgroups of V. As we already mentioned, a finitely generated group G is regular if and only if it has an embedding into V which acts freely on the whole of $\mathfrak{C}_2$. By Theorem 1.7 a finitely generated group G is $\mathcal{CF}$ if and only if it has an embedding into V which acts freely on an *open subset* of $\mathfrak{C}_2$, a weaker condition. If Lehnert's conjecture is proven to be true then the correspondence would be taken further, as a group G will be *co*$\mathcal{CF}$ if and only if there exists an embedding of G into V with *no* conditions on the dynamics.

Potential counterexamples to Lehnert's conjecture

Given the implications that Lehnert's conjecture could have on the correspondence between group theory and formal language theory, establishing its resolution has been a major area of study which prompted the work done in this thesis. Currently there are a few notable groups that are considered potential counterexamples to Lehnert's conjecture. The first is $\mathbb{Z} * \mathbb{Z}^2$ which was shown by Bleak and Salazar-Díaz to have no embedding into V [9]. It was first mentioned in conjunction with $co\mathcal{CF}$ groups by Holt et al. [20], where they conjecture that $\mathbb{Z} * \mathbb{Z}^2$ is not a $co\mathcal{CF}$ group. However, this is still an open problem, and should their conjecture prove to be false then, by Bleak and Salazar-Díaz, $\mathbb{Z} * \mathbb{Z}^2$ would be a counterexample to Lehnert's conjecture. The second notable example is the Grigorchuk group (see [19]), a finitely generated, infinite torsion group. Since V is torsion locally finite by Röver [27], V cannot contain the Grigorchuk group as a subgroup. Therefore if one were able to prove that the Grigorchuk group was $co\mathcal{CF}$ then it would also be a counterexample to Lehnert's conjecture.

In both cases above, to prove the groups are counterexamples to Lehnert's conjecture, one has to show that they were in the class $co\mathcal{CF}$. In Chapters 4 and 5 we consider potential counterexamples that are already known to be $co\mathcal{CF}$. To prove that these groups are counterexamples requires showing that they cannot embed into V as subgroups. We are unsuccessful in providing such a proof, however we do prove that a subclass of these groups *does* embed into V.

The classes $\mathfrak{V}_{aug}$ and $\mathfrak{F}_{aug}$

In 2014 Rose Berns-Zieze, Dana Fry, Johnny Gillings, Hannah Hoganson and Heather Mathews introduce a class of groups, which we call $\mathfrak{V}_{aug}$, which they prove to be $co\mathcal{CF}$. The groups in $\mathfrak{V}_{aug}$ follow the construction of Stefan Witzel and Matthew Zaremsky [33], who introduce a method for constructing "Thompson-like" groups. We do not describe the construction here, however more details are given in Chapter 4, and for further reference see "A users guide to cloning systems" by Zaremsky [34]. The groups in $\mathfrak{V}_{aug}$ are a generalisation of those first observed by Slobodan Tanusevski in his PhD thesis [30], although written in different language to Witzel and Zaremsky.

A group $V_{(G,\theta)}$ from $\mathfrak{V}_{aug}$ can be constructed using Thompson's group V, a finite group G and an endomorphism θ of G, the details of which are given in Chapter 4. In seeking to find a counterexample to Lehnert's conjecture, we actually study a "simpler" class of groups, $\mathfrak{F}_{aug}$, where each group $F_{(G,\theta)}$ from $\mathfrak{F}_{aug}$ is a subgroup of some $V_{(G,\theta)}$ in $\mathfrak{V}_{aug}$. We study the groups from $\mathfrak{F}_{aug}$ first for the same reason one might study the properties of Thompson's group F before moving on to the related yet more complex properties of V. As the class of $co\mathcal{CF}$ groups is closed under taking finitely generated subgroups, the groups in $\mathfrak{F}_{aug}$ are also $co\mathcal{CF}$, and therefore are all potential counterexamples to Lehnert's conjecture. Thus, we narrow our search for a counterexample to the following question.

Question 1: *Does there exists a group $F_{(G,\theta)}$ in $\mathfrak{F}_{aug}$ that does not embed into Thompson's group V?*

Our approach to answer Question 1 is to first find a group presentation for each group in

$\mathfrak{F}_{aug}$, as often one can use presentations to find embeddings between groups. In Chapter 4 we construct two presentations for a group $F_{(G,\theta)}$. The first, $\mathcal{F}^{inf}_{(G,\theta)}$, is an infinite presentation with three infinite sets of generators, and twelve infinite sets of relations. The second is a finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$ which consists of $2(N+1)$ generators and 26 finite sets of relations, where N is the order of the finite group G. We have no doubt that these figures could be reduced, however, they complete the purpose of showing that each group in $\mathfrak{F}_{aug}$ is finitely presented. The method we use for creating $\mathcal{F}^{inf}_{(G,\theta)}$ follows a similar line to that used for Thompson's group T by Burillo, Cleary, Stein and Taback [11]. The finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$ then follows by using Tietze transformations on $\mathcal{F}^{inf}_{(G,\theta)}$.

In Chapter 5 we use the infinite presentation created in Chapter 4 to prove that the group $F_{(G,\theta)}$ does in fact embed into Thompson's group V under certain conditions on G and θ. This result follows from a surprising isomorphism theorem.

Theorem 1.8. *Let G is a finite abelian group and suppose θ and ϕ are two idempotent endomorphisms of G. Then $F_{(G,\theta)}$ and $F_{(G,\phi)}$ are isomorphic.*

Berns-Zieve et al. make the following observation.

Observation 1.9. *If θ is the identity endomorphism that maps every elements of G to itself, then $V_{(G,\theta)}$ embeds as a subgroup of Thompson's group V.*

As the identity endomorphism is clearly idempotent, this observation with Theorem 1.8 gives the main result of the chapter.

Theorem 1.10. *Let G be a finite abelian group and θ an idempotent endomorphism of G. Then the group $F_{(G,\theta)}$ embeds into Thompson's group V.*

The proof of Theorem 1.8 relies on the presentation of $F_{(G,\theta)}$. Therefore, we do not know if a similar theorem to Theorem 1.10 exists for groups in $\mathfrak{V}_{aug}$, and leave it as an open question.

Chapter 2

Background and Definitions

2.1 Formal Languages

In this section we give a brief introduction to formal languages, reinforcing and elaborating on the definitions we introduced in the previous chapter. Much of the notation that we introduce below will be used throughout the thesis.

Let Λ be a finite set of symbols which we call an **alphabet**. A **word** over Λ is a finite sequence or "string" of symbols from the alphabet. The set Λ^* is the set of all finite words over the alphabet Λ. Any subset $L \subseteq \Lambda^*$ we call a **language** over the alphabet Λ. We use the symbol ε to denote the empty string consisting of zero symbols.

Example 2.1. Let $\Lambda_1 := \{0, 1\}$ be the alphabet consisting of the two formal symbols "0" and "1". The set $\Lambda_1^* = \{0, 1\}^*$ is a language in its own right, the language of all finite binary strings. The subset $\{0^n1^n \mid n \in \mathbb{N}_0\} \subseteq \Lambda_1^*$ is the language over the alphabet Λ_1 consisting of all the binary strings that are of the form "n zeroes followed by n ones".

Example 2.2. The empty set, denoted by $\emptyset$ which consists of no words (not even the empty word), is a language over any alphabet.

Example 2.3. Let $\Lambda_2 := \{a, b, c\}$. The finite subset $\{a, ab, abc\} \subset \Lambda_2^*$ is a language over the alphabet Λ_2.

Given an alphabet Λ and a language $L \subseteq \Lambda^*$ then the **complement** of L is the language $L^c = \{w \in \Lambda^* | w \notin L\}$.

Example 2.4. Given any alphabet Λ the complement of the language Λ^* is $\emptyset$.

Finite state automata

Formal languages can also be defined by theoretical machines called **automata**. Informally, an automaton M is a theoretical computation device that reads a word over a predefined alphabet and either accepts it or rejects it. The **language defined by the automaton**, which we denote by $L(M)$, is the set of all words that the automaton accepts. There are many different types of automata, two of which will appear in this work, **finite state automata** and **pushdown automata**. We direct the interested reader to the seminal introduction of this topic by Hopcroft and Ullman in [21] which goes into greater depth and breadth than we will be able to here.

Definition 2.5 (Finite state automata)**.** *A finite state automaton (or* ***FSA****) is a 5-tuple, $M = (\Lambda, Q, \delta, I, F)$, where Λ is a finite alphabet, Q is a finite set of states, $\delta \subseteq (Q \times \Lambda) \times Q$ is the transition relation, I is the set of start states of M and $F \subseteq Q$ is the subset of final/accept states.*

Let $M = (\Lambda, Q, \delta, I, F)$ be a finite state automaton as defined above. A **valid path** through M for a word $w = a_1a_2 \ldots a_n$ is a finite sequence of states $q_1, q_2, \ldots, q_{n+1}$ with $((q_i, a_i), q_{i+1}) \in \delta$ for all $1 \leq 1 \leq n$, and $q_1 \in I$. A finite word $w = a_1a_2 \ldots a_n$ is **accepted** by M if and only if there exists a valid path $q_1, q_2, \ldots, q_{n+1}$ for w where $q_{n+1} \in F$. The empty string ε is accepted if and only if there exists $q \in I \cap F$. The set of all words over Λ that are accepted by the automaton M forms a language which we call $L(M)$. We say that the automaton M accepts the language L if $L = L(M)$. If for any finite string w over the alphabet λ there exists exactly one valid path for w through the automaton we say that M is **deterministic**. Otherwise M is called **non-deterministic**.

Informally we speak of an FSA "reading" a word. We can picture an FSA M as a machine which is in some initial state $q_0 \in I$, reading a sequence of symbols $w = a_1a_2 \ldots a_n$ from Λ^* written on a tape. In one move, the automaton reads the first symbol a_1, transitions to some state q_1 such that $((q_0, a_1), q_1) \in \delta$, and advances the tape on by one symbol to a_2. The automaton then repeats this move for the next symbol of w and so on, until either there is no possible transition to take, and the machine stalls, or it reaches the end of the tape/string. A word w is accepted if the automaton can read the whole word and finish in a state end from F. If M is non-deterministic then it may have a choice of transitions to take after reading a letter.

Finite state automata can be represented by labelled directed graphs sometimes called state diagrams. Suppose $M = (\Lambda, Q, \delta, I, F)$ is a FSA. A state diagram $\mathcal{G}_M = (V, E)$ for M consists of a vertex set $V = Q$ and a set E of directed edges of the form (q_i, q_j) where $q_i, q_j \in Q$. There exists a directed edge $(q_i, q_j) \in E$ if and only if there exists $a \in \Lambda$ such that $((q_i, a), q_j) \in \delta$. Therefore, a directed edge may appear more than once in E. When we draw $\mathcal{G}_M$ we label the edge $(q_i, q_j) \in E$ associated to the relation $((q_i, a), q_j) \in \delta$. with an "$a$".

Example 2.6. Consider the automaton $M_1 = (\Lambda, Q, \delta, I, F)$ where

- $\Lambda = \{a, b\}$
- $Q = \{q_0, q_1\}$
- $\delta = \{((q_0, a), q_0), ((q_0, b), q_1), ((q_1, a), q_1), ((q_1, b), q_0)\}$
- $I = \{q_0\}$
- $F = \{q_1\}$.

Figure 2.1 gives the state diagram for M_1.

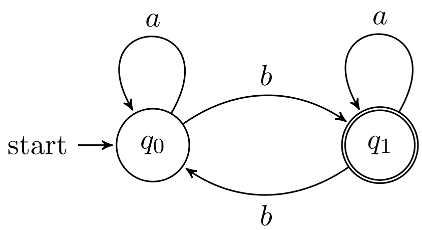


Figure 2.1: The state diagram for the finite state automaton M_1.

The state diagram in Figure 2.1 illustrates certain properties that will be true for all the automaton diagrams that we will draw. We indicate the final/accept states by drawing a "double circle" around them, in the example above q_1 is the only accept state. We also use a single arrow with no origin to indicate a start state, in this case the state q_0.

The language $L(M_1)$ accepted by the automaton consists of all the finite strings of a's and b's that contain an even number of b's. To see this visualise the process of M reading a word w. From either state, if M reads an a it remains in that state. If M reads a b then it moves to the alternate state. Reading one b will take the automaton to the state q_1. Therefore, if a word w has an odd number of b's it will end in state q_1, the accept state. If w has an even number of b's it will end in the state q_0, a reject state. Note that the automaton does not count the number of b's, it merely tracks the parity.

Definition 2.7 (Regular languages)**.** *A language L is called **regular** if and only if there exists a finite state automaton M that accepts L.*

There are other equivalent definitions of regular languages that use objects called formal grammars. However, we do not touch on these here.

The automaton M_1 in Example 2.6 is a deterministic FSA or DFSA. Given a *non-deterministic* FSA M there exists a construction (which can be found in [21]), called the **powerset construction**, by which one can create a DFSA N such that $L(M) = L(N)$.

Thus for every regular language L there exists a DFSA N such that $L = L(N)$. This leads to the following well known theorem concerning regular languages (see [21]).

Theorem 2.8. *The class of regular languages is closed under complementation.*

A sketch of the proof proceeds as follows. If L is a language accepted by a deterministic finite state automaton M then one can create an automaton M' that accepts the complement L^c by changing set set of accept states F to the complement set F^c. The automaton M' then rejects all the words accepted by M and must accept every word rejected by M.

Pushdown Automata

In Chapter 1, we mentioned that there exists a hierarchy for formal languages called the **Chomsky hierarchy**.

$$\text{Regular} \quad \subset \quad \text{Context Free} \quad \subset \quad \text{Context Sensitive} \quad \subset \quad \text{Recursively Enumerable}$$

The Chomsky hierarchy does not contain all languages, nor does it contain every complexity class that is known today, however for our purposes we are only interested in Regular and Context Free languages. As we go from left to right in the hierarchy above, the automata which define the languages of each type increase in their complexity. The context free languages, which contain the regular languages, are defined using **pushdown automata**.

Definition 2.9 (Pushdown Automata)**.** *A pushdown automaton (or **PDA**) is a 7-tuple, $M = (\Lambda, \Gamma, Q, \delta, I, \#, F)$, where Λ is a finite alphabet, Γ is a finite alphabet called the **stack alphabet**, Q is a finite set of states, $\delta \subseteq (Q \times (\Lambda \cup \varepsilon) \times \Gamma^*) \times (Q \times \Gamma^*)$ is the transition relation, $I \subseteq Q$ is the set of start states of M, $\# \in \Gamma$ is the initial stack symbol, and $F \subseteq Q$ is the set of final/accept states.*

One can see many similarities between the definition of a PDA and the definition of an FSA. However, a PDA introduces the concept of a **stack**, alluded to above by Γ the **stack alphabet**. A stack is a memory device that allows the automaton to store information. However, it can only access this information in a first-in-last-out basis. Formally it is a string of symbols from Γ where we define the top of the stack to be the left of the string and the bottom of the stack to be at right. The automaton can edit the stack by removing a finite string from the top of the stack, which we call a "pop", and placing a finite string onto the stack, which we call a "push".

We base the following from Chapter 5 of [21]. Let $M = (\Lambda, \Gamma, Q, \delta, I, \#, F)$ be a PDA as defined above. To define the way in which M computes a finite string w over the alphabet

Λ, we introduce notation that represents the current situation of the automaton. An instantaneous description, or ID, of M is a three-tuple $(q, v, \eta) \in Q \times \Lambda^* \times \Gamma^*$, where q is the current state of M, $v \in \Lambda^*$ is the remainder of the string w yet to be read by M, and $\eta \in \Gamma^*$ is the current stack. A **valid move**, or $\vdash_M$, is a relation on the set of ID's. Suppose $((q_i, a, \gamma_i), (q_j, \gamma_j)) \in \delta$. Then for all strings $v \in \Lambda^*$ and $\eta \in \Gamma$:

$$(q_i, av, \gamma_i\eta) \vdash_M (q_j, v, \gamma_j\eta).$$

If $a = \varepsilon$ then it is also called an **ε-move**, and none of the input string is read. We use the symbol $\vdash_M^*$ to denote the reflexive, transitive closure of $\vdash_M$, that is, a finite sequence of zero or more valid moves. We drop the subscript from $\vdash_M$ and $\vdash_M^*$ when the automaton M is understood.

The language $L(M)$ of a pushdown automaton is as follows.

$$L(M) = \{w : (q_0, w, \#) \vdash^* (p, \varepsilon, \gamma) \text{ where } q_0 \in I,\ p \in F \text{ and } \gamma \in \Gamma^*\}$$

The language $L(M)$ above accepts by final state, the same as a FSA. The contents of the stack at the end of the computation is irrelevant. However we could define M to *accept by empty stack*. We define the language $N(M)$ accepted by empty stack to be the following.

$$N(M) = \{w : (q_0, w, \#) \vdash^* (p, \varepsilon, \varepsilon) \text{ where } q_0 \in I \text{ and } p \in Q\}$$

There is a well-known theorem (see [21]) that the languages defined by PDAs that accept by final state and those defined by PDAs that accept by empty stack are the same. For our purposes in this thesis we will be creating PDAs that accept by final state.

The concept of determinism also exists for PDA's. A PDA M is deterministic if for every word $w \in \Lambda^*$ and every $q_0 \in I$ there is a unique sequence of valid moves $(q_0, w, \#) \vdash^* (p, \epsilon, \gamma)$, where $p \in Q$ and $\gamma \in \Gamma^*$.

We can represent pushdown automata graphically. Let $M = (\Lambda, \Gamma, Q, \delta, I, \#, F)$ be a pushdown automaton. A state diagram $\mathcal{G}_M = (V, E)$ for M consists of a vertex set $V = Q$ and a set E of labelled edges of the form (q_i, q_j) where $q_i, q_j \in Q$. An edge (q_i, q_j) is in E if and only if there exists $((q_i, a, \gamma_i), (q_j, \gamma_j)) \in \delta$. We label an edge (q_i, q_j) in $\mathcal{G}_M$ associated to the relation $((q_i, a, \gamma_i), (q_j, \gamma_j)) \in \delta$ with (a, γ_i, γ_j).

Example 2.10. Consider the PDA M_2 defined by the state diagram in Figure 2.2. The automaton M_2 has alphabet $\Lambda = \{a, b\}$ and stack alphabet $\Gamma = \{\#, \alpha\}$ where $\#$ is the bottom-of-the-stack symbol. The acceptance criteria for this automaton is by final state, although incidentally, when in the final state the automaton always has an empty stack.

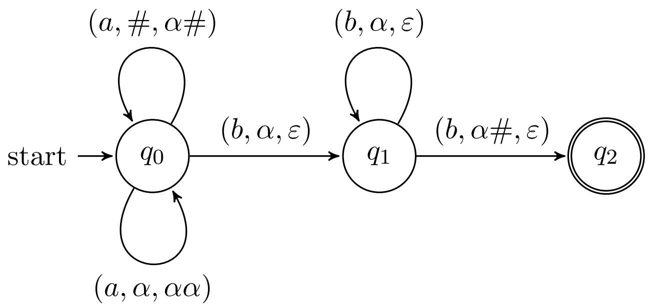


Figure 2.2: The state diagram for the finite state automaton M_1.

Consider the computation of a string w from Λ^*.

Reading the *a*'s: At q_0 the automaton starts to read the word w. The purpose of this stage is to count how many a's the word w begins with. For each a that is read the automaton pushes a symbol α onto the top of the stack. This stage ends once the first b has been read. If there is an α at the top of the stack then the automaton removes it and transitions to state q_2. If the very first letter of w is b then the automaton stalls as there is no transition of the form $\delta(q_1, b, \#)$. Thus any word beginning with a b is immediately rejected. If the word w does not contain any b's then the automaton remains in state q_0 and the word is rejected.

Reading the *b*'s: Once in state q_1 if any a's are read from w then the automaton stalls and the string is rejected. When b is read the transition depends on the current symbol at the top of the stack. If the top of the stack reads an α then the automaton pops it and remains in state q_1. If the stack reads $\#$ then we have reached the bottom of the stack, $\#$ is removed and we move to the accept state q_2. If we finish reading the string before the bottom-of-the-stack symbol is revealed then we end in q_1 and the word is rejected.

Acceptance: If the automaton reaches q_2 it means that every α that was put onto the stack at q_0 has been removed. Therefore the number of b's that have been read must be exactly the number of a's read at q_0. The state q_2 has no outgoing transitions thus if there is still more of w to read the automaton will stall and the word will be rejected. Thus for w to be accepted it must be of the form $a^n b^n$ for some natural number $n > 1$. Thus $L(M) \subseteq \{a^n b^n | n \in \mathbb{N}_1\}$. Furthermore, if an input string is of the form $w = a^n b^n$ then there exists a path through the automaton for w that ends in the state q_2. Thus $\{a^n b^n | n \in \mathbb{N}_1\} \subseteq L(M)$. Hence $L(M) = \{a^n b^n | n \in \mathbb{N}_1\}$.

Definition 2.11 (Context free languages)**.** *A language L is called context free if and only if there exists a PDA M such that $L = L(M)$.*

We saw in Theorem 2.8 that the class of regular languages is closed under complementation. This is *not* the case for context free languages. Given a context free language L its complement L^c need not be context free. However, there are an important subclass of the context free languages that are closed under complementation.

Theorem 2.12. *Suppose L is a language accepted by a* deterministic *pushdown automaton. Then L^c is also a context free language.*

The idea of the proof is the same as the case for regular languages, one simply swaps the accept and reject states of the original automaton accepting the language L.

2.2 Group presentations

A group $(G, \circ)$ is a set G and a binary operation $\circ : G \times G \to G$, that together satisfy the axioms of associativity, identity and invertibility. Commonly the group $(G, \circ)$ is simply denoted by G. Another way of defining a group G is by using a **group presentation**. In what follows we expand on our definition of a group presentation given in Chapter 1.

Free groups

To define a group presentation one begins with a fundamental object in combinatorial group theory, a free group. The theory of free groups is well known and in what follows we will not provide proofs for the statements that are made. We base much of what follows on Chapter 3 of [10].

Let X be an arbitrary set of symbols. Set $X^{-1} := \{x^{-1} : x \in X\}$, where x^{-1} is a distinct symbol corresponding to the symbol x from X. Thus X^{-1} is in bijective correspondence with X and $X \cap X^{-1} = \emptyset$. Set $X^{\pm} := X \sqcup X^{-1}$ and let W be the set of all finite words over $X^{\pm}$, that is, $W = X^{\pm *}$. We now introduce an equivalence relation between words in W. Two words u and v are equivalent if there exists a sequence of words $u = w_1, w_2, w_3, \ldots, w_n = v$ such that w_i differs from w_{i+1} by either the insertion or deletion of a subword of the form $x^{-1}x$ or xx^{-1} for some $x \in X$.

For example, suppose $X = \{x, y\}$, $u = xyx^{-1}xy$ and $v = yy^{-1}xyy$. Then the following sequence of words will take u to v.

$$u = xyx^{-1}xy \to xyy \to yy^{-1}xyy = v.$$

Thus u is equivalent to v under the equivalence relation defined above.

We denote the equivalence class of a string u by $[u]$, and use $[W]$ to represent the set of all equivalence classes of W. A word is called **freely reduced** if it contains no subwords of the form xx^{-1} or $x^{-1}x$ for any $x \in X$. It is a well known result that each equivalence class $[u]$ in $[W]$ contains exactly one reduced word. One can define a multiplication on W by $[u][v] = [uv]$, where uv is concatenation of strings. It is an exercise that this multiplication is well defined (see [10]). The free group F_X is defined as the set $[W]$ together with this multiplication. Practically, it is often more convenient to treat elements of F_X as words over the alphabet $X^{\pm}$, assuming two words are equal if and only if their reduced words are equal.

Presentations of groups

Definition 2.13 (Generating sets)**.** *Let G be a group and A a set of elements from G. We denote by $\langle A\rangle$ the set of all elements generated by A, i.e. every element in G created by a product of elements from A and their inverses A^{-1}. The set A is a* ***generating set*** *for G if $\langle A\rangle = G$. Thus A is a generating set if and only if the smallest subgroup of G containing A is G itself.*

Suppose G is a group with generating set A. Let X be a set of symbols such that the map $X \to A$ is a bijection of sets. Then the map $X \to A$ extends to a surjective homomorphism of groups, $\phi : F_X \to G$, that maps the generators of F_X onto the generators of G. As ϕ is surjective, the First Isomorphism Theorem gives $G \cong F_X/ker(\phi)$. The kernel $ker(\phi)$ consists of every word in F_X that gets sent to the identity of G under ϕ, we call such words **relators**. Suppose there exists a subset R of F_X such that the smallest normal subgroup containing R in F_X is $ker(\phi)$. Then we call the set R a **set of defining relators** for G. The smallest normal subgroup containing some set R is called its **normal closure** and is given by

$$\langle\langle R\rangle\rangle = \left\{ \prod_i^n w_i^{-1} r_i w_i : r_i \in R, w_i \in F_X \right\}. \tag{2.1}$$

Thus every relator can be written as a product of conjugates of elements from R. Therefore, even though there are an infinite number of relators, the set of *defining relators* may be finite.

Thus, the expression $\langle X|R\rangle$, determines the group G up to isomorphism and is referred to as **the presentation of G**. The presentation $\langle X|R\rangle$ is called finite if both X and R are finite.

Example 2.14 (Integers)**.** Consider the integers under addition $(\mathbb{Z}, +)$. The group has one generator and no relations. Therefore a presentation for $(\mathbb{Z}, +)$ is $\langle x|-\rangle$.

Example 2.15 (Cyclic groups)**.** Consider the group $\mathbb{Z}_n$, the integers modulo the natural number n. The group has one generator, call it x, and any product of generators x^m, where $m \bmod n = 0$, is equivalent to the identity and thus is a relator. If x^m is a relator then $m = np$ for some integer p and $x^m = (x^n)^p$. Thus the set consisting of the single relator x^n is a set of defining relators for $\mathbb{Z}_n$. Hence a presentation of $\mathbb{Z}_n$ is $\langle x | x^n \rangle$.

A presentation can also be defined using **relations** instead of relators. Let G be a group defined by a presentation $\langle X|R \rangle$. A relation is an identity of the form $u = v$ where u and v are words over the alphabet $X^{\pm}$ and $uv^{-1} \in \langle\langle R \rangle\rangle$. Given a presentation $\langle X|R \rangle$, where R is a set of relators, the set $R' := \{r = 1 : r \in R\}$ is called the **set of defining relations** and the group presentation $\langle X|R' \rangle$ represents the same group as $\langle X|R \rangle$. Conversely, suppose G is a group defined by the presentation $\langle X|S \rangle$ where S is a set of relations. Then one can define the **set of defining relators** as $S' := \{uv^{-1} : \text{if } u = v \text{ is in } S\}$ and $\langle X|S' \rangle$ represents the same group. Thus one can choose to define a group presentation with either relations or relators.

2.3 R. Thompson's groups

For the rest of the chapter, we give a brief introduction to the R. Thompson groups F, T and V, in particular, how one can construct the groups as homeomorphisms of the Cantor set. A helpful way to describe elements of F, T and V is to use *binary trees*. In what follows, we introduce the theory behind binary trees, describe how they relate to the Cantor set and then go on to define the R. Thompson groups using the notation that we have established.

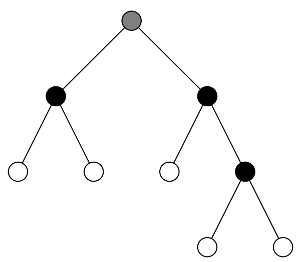
2.3.1 Binary trees

An undirected **graph** $\mathcal{G}$ is a an ordered pair $\mathcal{G} = (V, E)$ where V is set of **vertices** or **nodes** and E is a set of **edges** between the vertices of V. We denote edges by unordered pairs of vertices of the form $\{v, w\}$, where v and w are vertices in V. Two vertices v and w are **adjacent** if there exists an edge $\{v, w\}$ between them. A **loop** is an edge of the form $\{v, v\}$ from a vertex v to itself. An undirected graph $\mathcal{G}$ is **finite** if it contains a finite number of vertices and edges, and **simple** if it does not contain multiple edges between vertices or loops. A **path** in $\mathcal{G}$ is a sequence of vertices $v_1, v_2, v_3, \ldots, v_k$ such that there exists an edge $\{v_i, v_{i+1}\}$ in E for all $1 \leq i < k$. The **length** of a path is the number of edges the path contains, thus the path $v_1, v_2, v_3, \ldots, v_k$ has length $k - 1$. A **cycle** is a path $v_1, v_2, v_3, \ldots, v_k$ such that $v_1 = v_k$. A graph $\mathcal{G}$ is **connected** if for any two vertices v and w in $\mathcal{G}$ there exists a path $v = v_1, v_2, v_3, \ldots, v_k = w$ connecting them.

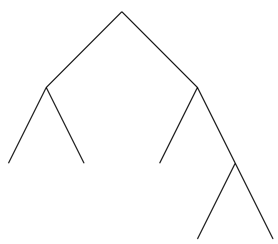
A **tree** is a simple, connected graph with no cycles. A tree is **rooted** if it has an assigned vertex which we call the **root**. We define a **descending path** for a node n in a rooted tree to be a path P beginning at the root and ending at n such that p does not contain any repeated vertices i.e. a path of minimal length from the root to n. The descending path for the root itself is the trivial path of length zero consisting solely of the root. The properties of a tree give us the following lemma.

Given a vertex v in a rooted tree, its **parent** p is the adjacent vertex that is contained in the unique descending path from the root to v. Conversely a **child** c of a vertex v is an adjacent node for which the v is the parent of c. From the definition of a rooted tree one can see that a non-root vertex must have exactly one parent but can have any number of children, including none. A vertex that has no children is called a **leaf**. A vertex in a tree that is not the root or a leaf we call an **internal node.** A **binary tree** is a rooted tree such that every vertex has at most two children. Every binary tree throughout this work will in fact be **strongly binary**, in that every vertex will have either two children or none. As this applies to all the binary trees that follow we will drop the descriptor "strongly" and assume it throughout. We will draw rooted trees so that parent vertices are drawn above their children, hence all the rooted trees will be drawn with the root at the top and all other vertices below it. As all the trees we will be considering will be rooted, we will drop the descriptor and refer to *rooted trees* simply as *trees*.

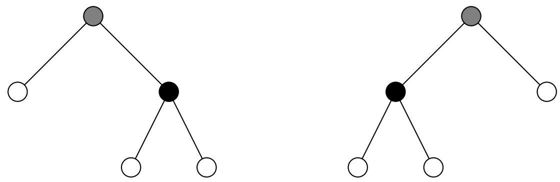
The tree below is an example of a finite binary tree. The grey vertex at the top is the **root** and the white vertices at the bottom are the **leaves**. Every vertex that isn't white we call an **internal node**.



Often we will not explicitly draw the circular vertices on the binary trees. For example, the tree above could instead be drawn as the following.



The way in which the nodes of a binary tree are drawn on the page are important. For example the two trees

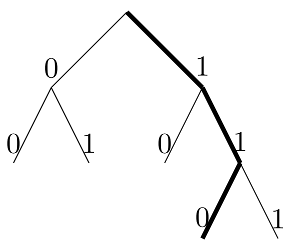


are not the same even though they are isomorphic as graphs. Given an internal node n of a binary tree we distinguish the two children of n by either **left** or **right**, depending on what side of the parent they were drawn. We say that the left child is connected by a **left edge** and the right child by a **right edge**.

Lemma 2.16. *Let T be a binary tree. Then there exists a bijection between the set of vertices of T and the set of descending paths.*

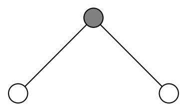
Proof. Let T be a binary tree. To prove the result we must show that for each node n in T, there exists a unique descending path from the root to n. As a tree is connected, for each node n there must exist a descending path from the root. Suppose $P_1 = v_1, v_2, \ldots, v_k$ and $P_2 = w_1, w_2, \ldots, w_k$ are two distinct descending paths for n, each of the minimal length $k-1$ where by definition $v_1 = w_1$ is the root of T and $v_k = w_k = n$. As P_1 and P_2 are different there must exist some minimal integer $i < k-1$ such that $v_i = w_i$ and $v_{i+1} \neq w_{i+1}$. As both paths terminate at the same node n, there must also exists a minimal integer $j > i$ such that $v_j \neq w_j$ and $v_{j+1} = w_{j+1}$. Thus there exists a path in the tree $P_3 = v_i, v_{i+1}, \ldots, v_j, \ldots, v_i$ which is a cycle in T. This is a contradiction and hence $P_1 = P_2$ and the descending path from the root to n must be unique. □

For each node n in a binary tree T we introduce an **address** by which to identify n in T. We begin by labelling all left edges of T with a "0" and all right edges by a "1". Suppose P is the descending path through the binary tree T for a node n. Starting at the root the sequence of left edges and right edges we follow in P through T will give a sequence of zeroes and ones. A sequence of zeroes and ones is called a **binary string** and we denote the set of all finite binary strings by $\{0,1\}^*$. As each node n in T has a unique descending path by Lemma 2.16, there exists a unique binary string for n, which we call its address in T. For example the bold path in the tree below corresponds to the string "110". Thus we say that the node n on which the path terminates has address 110.

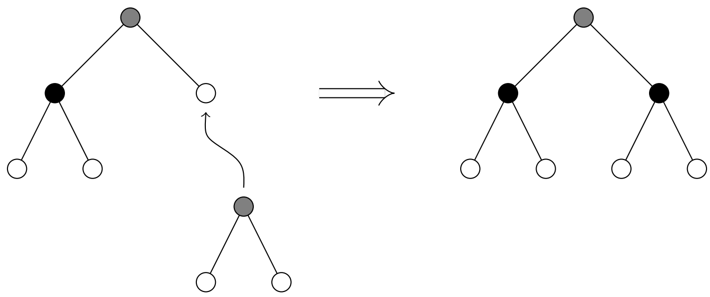


The descending path of the root of T is the trivial path of length zero consisting solely of the root and thus its address in $\{0,1\}^*$ is the empty string ε. For the remainder of the thesis we will not distinguish between a node n and its address, thus in the example above we would say that $n = 110$.

A **caret** is a binary tree consisting of just the root and two leaves.



Given a binary tree T we can "attach" a caret c to one of its leaves l by associating the root of c with the leaf l in T. As an example, we attach a caret to the final leaf of the binary tree below.



Therefore, beginning with the root we can construct any finite binary tree by progressively attaching carets. This gives us the following result.

Lemma 2.17. *If a binary tree is constructed from n carets then it has $n+1$ leaves.*

Proof. Let $\mathcal{P}(n)$ be the statement that all trees constructed from n caret have $n+1$ leaves for some integer $n \geq 0$. The proof proceeds by induction on n. The tree consisting of 0 carets contains only the root, thus has only one leaf and therefore satisfies the lemma. Suppose $\mathcal{P}(m)$ is true for some $m \geq 0$. Let T be some tree constructed from $m+1$ carets and suppose it has k leaves. Let T' be the tree created by removing a caret from T. Removing a caret from T removes two leaves and exposes an internal node, which becomes a leaf in T'. Thus T' has m carets and $k-1$ leaves. By $\mathcal{P}(m)$ the tree T' must have $m+1$ leaves and thus $k = m+2$. Hence $\mathcal{P}(m+1)$ is also true, and by induction $\mathcal{P}(n)$ is true for all $n \geq 0$. $\square$

The infinite binary tree $\mathcal{T}_2$

The **infinite binary tree** $\mathcal{T}_2$ is formally defined as the binary tree in which every node has two children. Thus $\mathcal{T}_2$ has a (countably) infinite number of nodes and zero number of leaves. Figure 2.3 gives the first four levels of $\mathcal{T}_2$.

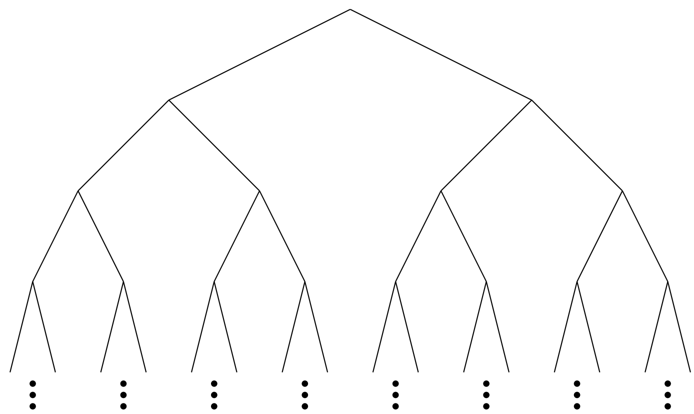


Figure 2.3: $\mathcal{T}_2$, the infinite binary tree

We can build addresses for nodes in $\mathcal{T}_2$ in the same way as finite binary trees, assigning a "0" to every left edge and a "1" to every right edge. From Lemma 2.16 we have the following corollary.

Corollary 2.18. *There exists a bijection between the set of nodes of $\mathcal{T}_2$ and the set of all binary strings $\{0,1\}^*$.*

We define an **infinite descending path** in $\mathcal{T}_2$ to be a path from the root that contains infinitely many nodes and never crosses an edge twice. By using the binary address system on $\mathcal{T}_2$ one can observe that there exists a bijection between the set of all infinite descending paths and the set of all infinite binary strings, which we denote by $\{0,1\}^\omega$. The set $\{0,1\}^\omega$ can be thought of as the "boundary" of the tree $\mathcal{T}_2$. We will see in the next section that the set $\{0,1\}^\omega$ is in fact in bijection with the **Cantor set**.

2.3.2 The Cantor set

The Cantor set, although named after Georg Cantor for his work in 1883 [13], was thought to have first been discovered by Henry J S Smith in 1874 [29] in his paper "On the Integration of Discontinuous Functions". The construction of the standard ternary Cantor set proceeds as follows. One begins with the unit interval $[0,1]$ which we call $\mathcal{C}_0$. One then deletes the open "middle third" interval $\left(\frac{1}{3},\frac{2}{3}\right)$ from $\mathcal{C}_0$ to produce $\mathcal{C}_1 = \left[0,\frac{1}{3}\right] \cup \left[\frac{2}{3},1\right]$. The procedure is then repeated on the two remaining intervals in $\mathcal{C}_1$, where we delete the middle thirds from both to give $\mathcal{C}_2 = \left[0,\frac{1}{9}\right] \cup \left[\frac{2}{9},\frac{1}{3}\right] \cup \left[\frac{2}{3},\frac{7}{9}\right] \cup \left[\frac{8}{9},1\right]$. This process continues ad infinitum, where the set $\mathcal{C}_{k+1}$ is created by deleting the open middle thirds from all the intervals comprising $\mathcal{C}_k$. Thus for any natural number n the set $\mathcal{C}_n$ is defined as

$$\mathcal{C}_n := \frac{\mathcal{C}_{n-1}}{3} \cup \left(\frac{2}{3} + \frac{\mathcal{C}_{n-1}}{3}\right). \tag{2.2}$$

The process is illustrated by the figure below.

Figure 2.4: The first five iterations of the Cantor set construction

The Cantor set is then defined to be all the points in the unit interval that are not removed at any step in the process.

One can set up a binary labelling system on the construction process similar to that found on binary trees. Each interval that is created in the construction process is contained in an interval from the previous stage. We call this its **parent interval**, similar to the parent of a node in a binary tree. If the interval contains the left limit point of its parent we label it with a "0", if it instead contains the right limit point of its parent we label it with a "1". For example, the first three stages are labelled as below.

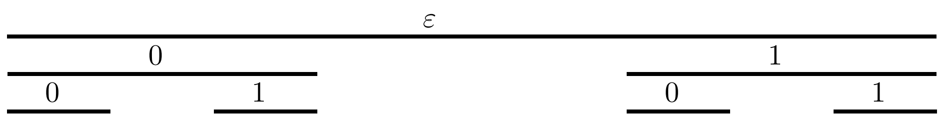


Figure 2.5: The binary labelling on the first two stages of the Cantor set construction

Thus each interval in the construction can be uniquely identified by a finite binary sequence, defined as we descend down the stages of the iterated process. For example the interval $\left[\frac{2}{9}, \frac{1}{3}\right]$ which appears after the second iteration would be given by the binary sequence "01" as we descend first to the interval $\left[0, \frac{1}{3}\right]$ which is labelled with a "0" and then to the interval $\left[\frac{2}{9}, \frac{1}{3}\right]$ itself which is labelled with a "1". If x is a point in the Cantor set then at each stage in the construction process x is contained within a unique interval. Thus x corresponds to an infinite sequence of intervals which in turn corresponds to an infinite binary sequence by the labelling we have introduced. Therefore the Cantor set is in bijective correspondence to the set $\mathfrak{C}_2 := \{0,1\}^\omega$, the set of countably infinite binary strings, sometimes seen as $\{0,1\}^\mathbb{N}$ or $2^\mathbb{N}$ in some of the literature. We call $\mathfrak{C}_2$ a **Cantor space**. Although we normally use lower case roman letters, such as w, to represent strings from $\{0,1\}^\omega$, occasionally we may use "arrowed" notation, such as $\vec{w}$, to distinguish from finite strings if the situation arose.

Topology of the Cantor space

The Cantor space $\mathfrak{C}_2$ can also be equipped with a topology. The set $\{0,1\}^\omega$ can be interpreted as the product of countably many copies of the the binary set $\{0,1\}$,

$$\{0,1\}^\omega = \prod_{n=1}^{\infty} \{0,1\}. \tag{2.3}$$

By giving each copy of $\{0,1\}$ the discrete topology we can induce a topology on $\{0,1\}^\omega$, namely the product topology. If u is a finite binary string from $\{0,1\}^*$ then the set

$$\lfloor u \rfloor = \{u\vec{w} | \vec{w} \in \{0,1\}^\omega\}$$

is called a cylinder set or **cone**. An alternative way to describe a cone $\lfloor u \rfloor$ is as the set of all infinite binary strings that have the finite string u as a **prefix**. The set of all cones forms a basis for the product topology on $\{0,1\}^\omega$. Notice that for each cone $\lfloor u \rfloor = \{u\vec{w} | \vec{w} \in \{0,1\}^\omega\}$ there exists a bijective map $\varphi : \lfloor u \rfloor \to \mathfrak{C}_2$ given by $(u\vec{w})\varphi = \vec{w}$. Thus each cone is itself a Cantor space. This property of a space containing copies of itself is called **self-similarity** and is a property found in some fractals, of which the Cantor space would be one.

Metric on the Cantor space

We can also introduce a metric $d : \mathfrak{C}_2 \times \mathfrak{C}_2 \to \mathbb{N}_0$ on $\mathfrak{C}_2$. If $u = a_1a_2 \cdots a_n$ is a finite binary string, $a_i \in \{0,1\}$, then we define the **length** of u to be n, sometimes written $length(u) = n$. Suppose x and y are two points in $\mathfrak{C}_2$. A **common prefix** u of x and y is a prefix u that is both a prefix of x and a prefix of y. Then the metric d defines the distance between x and y as

$$d(x,y) = inf\Big\{2^{-length(u)} : u \text{ is a common prefix of } x \text{ and } y\Big\}. \tag{2.4}$$

Proposition 2.19. *The set $\mathfrak{C}_2$ with the metric $d : \mathfrak{C}_2 \times \mathfrak{C}_2 \to \mathbb{N}_0$ defined in (2.4) is a metric space.*

Proof. Suppose $x = a_1a_2a_3\ldots$, $y = b_1b_2b_3\ldots$ and $z = c_1c_2c_3\ldots$ are three points in $\mathfrak{C}_2$, $a_i, b_i, c_i \in \{0,1\}$. There are four conditions to check.

1. (Positivity) By definition $d(x,y) > 0$ for any x and y.

2. (Identity) First note that if there exists a common prefix of x and y of length n, then there exists a common prefix of x and y for all natural numbers $m \leq n$. As $2^{-n} > 0$ for all $n \in \mathbb{N}_1$, if $d(x,y) = 0$ then for each natural number n there must exist a common prefix of x and y of length n. Thus $x = y$.

3. (Symmetry) The metric is commutative thus the symmetry condition holds.

4. (Triangle inequality) For any x, y, z the triangle inequality states that $d(x, y) \leq d(x, z) + d(y, z)$. If $x = y = z$ then the triangle inequality holds. If $x = y$ then $d(x, y) = 0$ and the inequality will always hold as the metric is always positive. If $x = z$ when $y \neq z$ then $d(x, y) = d(z, y)$ as required. Suppose x, y and z are all distinct from each other. Let $a \in \{0, 1\}^*$ be the longest prefix shared by x and z and $b \in \{0, 1\}^*$ be the longest prefix shared by y and z. If $length(a) \leq length(b)$ then a must be a prefix of b. Thus a is a shared prefix of x and y and $d(x, y) \leq length(a) = d(x, z)$. If instead $length(a) > length(b)$ then b must be a strict prefix of a. In this case b is a shared prefix of x and y and $d(x, y) \leq length(b) = d(y, z)$. Thus in either case $d(x, y) \leq d(x, z) + d(y, z)$ for any x, y and z which means the triangle inequality holds.

□

Cantor space and the infinite binary tree $\mathcal{T}_2$

There exists a natural correspondence between $\mathfrak{C}_2 = \{0, 1\}^\omega$ and the infinite binary tree $\mathcal{T}_2$. As we mentioned in our discussion of $\mathcal{T}_2$, the boundary of $\mathcal{T}_2$ which is the set of all infinite descending paths is in bijective correspondence to the set $\{0, 1\}^\omega$. A cone $\lfloor u \rfloor$ is a subset of this boundary that exists below a certain node in $\mathcal{T}_2$ given by the address u. The figure below illustrates the concept and one can see from the shape of the shaded section why we call $\lfloor u \rfloor$ a cone.

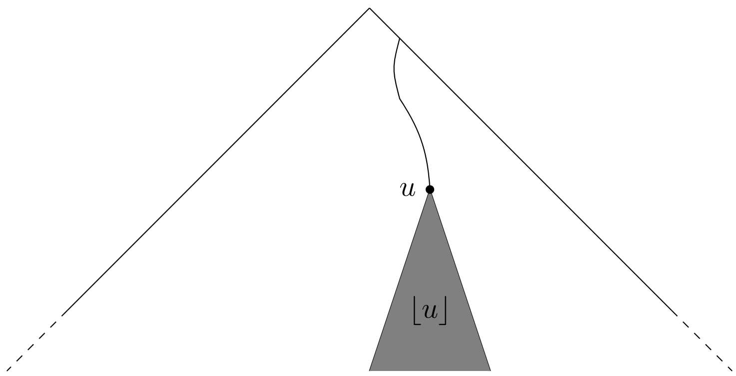


Figure 2.6: An illustration of a cone $\lfloor u \rfloor$ in $\mathcal{T}_2$.

2.3.3 R.Thompson's group V

R. Thompson's group V_2 is a group of self homeomorphisms of the Cantor space $\mathfrak{C}_2$ under composition. A **homeomorphism** is a continuous, bijective function between topological spaces that maps open sets to open sets. For the remainder of the work we will exclusively refer to V_2 as V and drop the subscript as we will never change the alphabet size of the Cantor space we are considering.

Prefix replacement maps

In this section we construct homeomorphisms of $\mathfrak{C}_2$ which are called **prefix replacement maps**. These will form our elements of Thompson's group V. The process of defining a prefix replacement map begins with a binary relation $\preceq$ on $\{0,1\}^*$ defined as

$u \preceq v$ if and only if the string v is of the form $v = uw$ for some string $w \in \{0,1\}^*$.

In other words $u \preceq v$ if and only if u is a **prefix** of v. The following lemma proves that this relation is in fact a partial order on the set of all finite binary strings.

Lemma 2.20. *The binary relation $\preceq$ is a partial order on the set $\{0,1\}^*$.*

Proof. To prove that $\preceq$ is a partial order we have to show that it is reflexive, anti-symmetric and transitive. Reflexivity is obvious as any string is its own prefix. Suppose u and v are strings in $\{0,1\}^*$ such that $u \preceq v$ and $v \preceq u$. The there exists strings a and b in $\{0,1\}^*$ such that $u = va$ and $v = ub$. By substitution we have $u = uba$ and $v = vab$ and thus a and b must both be the empty string ε. Hence $u = v$ and $\preceq$ is anti-symmetric. Finally suppose that u, v and w are strings in $\{0,1\}^*$ such that $u \preceq v$ and $v \preceq w$. Then there exists strings c and d in $\{0,1\}^*$ such that $v = uc$ and $w = vd$. By substituting for v we have $w = ucd$ and therefore $u \preceq w$, which proves transitivity. $\square$

Given two strings u and v if $u \preceq v$ or $v \preceq u$ then we say u and v are **comparable**. Else u and v are called **incomparable**, a property we denote by $u \perp v$. Any subset of $\{0,1\}^*$ that consists solely of pairwise incomparable elements is called an **antichain**. Suppose $A = \{u_1, u_2, \ldots, u_n\}$ is a finite antichain where each u_i is a string from $\{0,1\}^*$. We say that A is **complete** if for every infinite string w in $\{0,1\}^\omega$ there exists a string u_i from A such that u_i is a prefix of w. Note that uniqueness is guaranteed by the fact that all the strings in A are incomparable. For example, the set $\{00, 010, 011, 1\}$ is a finite complete antichain, whereas the set $\{001, 01, 1\}$ does not qualify as it contains no prefix for points in the cone $\lfloor 000 \rfloor$. The theory of prefixes and antichains is well known and not restricted to the sets $\{0,1\}^*$ and $\{0,1\}^\omega$. Jean-Camille Birget in his 2004 paper [5] gives a more general treatment of the subject, also with Thompson's groups in mind. He uses the term *prefix codes* to denote antichains and the term *maximal* instead of complete.

We can use antichains to define homeomorphisms of the Cantor set. Suppose $A_1 = \{u_1, u_2, \ldots, u_n\}$ and $A_2 = \{v_1, v_2, \ldots, v_n\}$ are two complete antichains of order n. Let φ be a bijection between A_1 and A_2 that maps each string u_i from A_1 to a unique string

v_i from A_2. We call φ a **prefix replacement**. Each individual map $u_i \mapsto v_i$ from φ we call a **prefix replacement rule**. The bijection φ induces a map Φ on the space $\mathfrak{C}_2$ in the following way. Suppose w is a string from $\mathfrak{C}_2$. As A_1 is a finite complete antichain, w has some prefix u_i from A_1 such that $w = u_i\vec{w}$ where $\vec{w}$ is from $\{0,1\}^\omega$. The map Φ then acts on the string w by

$$\Phi(w) = \Phi(u_i\vec{w}) = \varphi(u_i)\vec{w} = v_i\vec{w}. \tag{2.5}$$

We call Φ a **prefix replacement map**. The following lemma proves that Φ is a homeomorphism of $\mathfrak{C}_2$.

Lemma 2.21. *Let $\varphi : A_1 \to A_2$ be a prefix replacement. The induced prefix replacement map Φ from φ is a homeomorphism of the Cantor set $\mathfrak{C}_2 = \{0,1\}^\omega$.*

Proof. Let Φ be the prefix replacement map induced by the prefix replacement $\varphi : A_1 \to A_2$. As A_1 and A_2 are complete antichains and φ is bijective then Φ must also be bijective. This leaves us to check continuity, to show that the preimage of every open subset of $\mathfrak{C}_2$ under Φ is open. Suppose U is some open subset of $\{0,1\}^\omega$. Then U can be written as a union of cones $U = \bigcup_i \lfloor u_i \rfloor$ where each $\lfloor u_i \rfloor$ is contained in some unique cone $\lfloor a_i \rfloor$ where $a_i \in A_2$. Therefore $u_i = a_i v$ for some $v \in \{0,1\}^*$ and the preimage of $\lfloor u_i \rfloor$ under Φ is the cone $\lfloor (a_i)\phi^{-1} v \rfloor$. Therefore the preimage of U under Φ is a union of cones and hence is open. Continuity of the inverse function is given by the same argument. $\square$

A prefix replacement rule $u_i \mapsto v_i$ defines a partial function $\lfloor u_i \rfloor \mapsto \lfloor v_i \rfloor$ on $\mathfrak{C}_2$ which we call a **cone map**. The cone map $\lfloor u_i \rfloor \mapsto \lfloor v_i \rfloor$ is defined on the cone $\lfloor u_i \rfloor$ and maps the point $u_i\vec{w}$ to the point $v_i\vec{w}$ in $\lfloor v_i \rfloor$. Therefore, given a prefix replacement $\varphi : A_1 \to A_2$ the induced prefix replacement map Φ can be defined by the collection of cone maps induced from the prefix replacement rules of φ.

Example 2.22. Define complete antichains $A_1 = \{0, 10, 11\}$ and $A_2 = \{00, 01, 1\}$. An example of a prefix replacement is the map $\varphi : A_1 \to A_2$ such that

$$0 \mapsto 01$$
$$10 \mapsto 1$$
$$11 \mapsto 00$$

The induced map $\Phi : \{0,1\}^\omega \to \{0,1\}^\omega$ can be then defined by the cone maps;

$$\lfloor 0 \rfloor \mapsto \lfloor 01 \rfloor$$

$$\lfloor 10 \rfloor \mapsto \lfloor 1 \rfloor$$
$$\lfloor 11 \rfloor \mapsto \lfloor 00 \rfloor.$$

As an example consider the infinite string of zeroes, denoted by $\bar{0} \in \mathfrak{C}_2$. Under Φ this point gets mapped to $(\bar{0})\Phi = 01\bar{0}$.

Expansions and reductions

Consider a prefix replacement $\varphi : A_1 \to A_2$ that contains the rule $u \mapsto v$ for some $u, v \in \{0,1\}^*$. Notice that the two prefix replacement rules $u0 \mapsto v0$ and $u1 \mapsto v1$ together induce the same partial map on $\mathfrak{C}_2$ as the single rule $u \mapsto v$ from φ. Thus the prefix replacement $\varphi' : A_1 \to A_2$ we create from φ by replacing the rule $u \mapsto v$ with the two rules $u0 \mapsto v0$ and $u1 \mapsto v1$, induces the same prefix replacement map as φ. We call such an operation an **expansion** of φ. The inverse operation is called a **reduction**. We have the following observation which follows from the discussion above.

Observation 2.23. *Suppose φ_1 and φ_2 are two prefix replacements such that there exists a finite sequence of prefix replacements $\varphi_1 = r_1, r_2, \ldots, r_n = \varphi_2$ such that for all $1 \leq i < n$, r_i differs from r_{i+1} by an expansion or a reduction. Then φ_1 and φ_2 induce the same prefix replacement map.*

For example consider the two prefix replacements φ_1 and φ_2.

φ_1	φ_2
$0 \mapsto 1$	$00 \mapsto 10$
$1 \mapsto 0$	$01 \mapsto 11$
	$1 \mapsto 0$

The prefix replacement φ_2 is created from φ_1 by expanding the rule $0 \mapsto 1$ once. One can observe that any string from $\mathfrak{C}_2$ will get mapped to the same string under the prefix replacement map induced by φ_1 or φ_2.

Definition 2.24. *Suppose $\varphi : A_1 \to A_2$ is a prefix replacement such that one cannot apply a reduction. Then φ is called **irreducible**. Else it is called **reducible**.*

Lemma 2.25. *For each prefix replacement map Φ there exists a unique irreducible prefix replacement φ that induces Φ.*

Proof. Let $\Phi : \mathfrak{C}_2 \to \mathfrak{C}_2$ be some prefix replacement map induced from a prefix replacement $\varphi_1 : A_1 \to A_2$. Suppose φ_1 is reducible. Then φ_1 contains two prefix replacement rules $u0 \mapsto v0$ and $u1 \mapsto v1$ in φ_1 for some $u, v \in \{0,1\}^*$. Then by Observation 2.23 the prefix replacement φ_2 created from φ_1 by applying a reduction to the rules $u0 \mapsto v0$ and $u1 \mapsto v1$ induces the same prefix replacement map Φ. If φ_2 is itself reducible then we repeat the steps above. The process of reduction continues until one reaches a prefix replacement which is irreducible. As the number of rules in φ_{i+1} is one less that φ_i the process must terminate in finite time.

Suppose there exists two distinct prefix replacements $\varphi_a : A_1 \to A_2$ and $\varphi_b : B_1 \to B_2$ that are both irreducible and both induce the same prefix replacement map Φ. Suppose $A_1 = B_1$ and $A_2 = B_2$. Then as φ_a and φ_b are not the same they must defined different bijections between the antichains and thus induce two distinct homeomorphisms of $\mathfrak{C}_2$, a contradiction. Suppose $A_1 = B_1$ and $A_2 \neq B_2$. Then there exists u, w_1, w_2 such that $w_1 \neq w_2$ and φ_a contains the rule $u \mapsto w_1$ and φ_b contains the rules $u \mapsto w_2$. The cone maps defined by these two rules are different and thus the homeomorphisms induced by φ_a and φ_b must also be different, another contradiction. The parallel case when $A_1 \neq B_1$ and $A_2 = B_2$ follows likewise. Suppose then that $A_1 \neq B_1$ and $A_2 \neq B_2$. As the antichains are complete there must exist $u_1 \in A_1$ and $v_1 \in B_1$ such that $u_1 \neq v_1$ and either $u_1 \preceq v_1$ or $v_1 \preceq u_1$. Without loss of generality assume that $u_1 \preceq v_1$, that is, $v_1 = u_1 w$ for some $w \in \{0,1\}^*$. As B_1 is complete we choose v_1 such that $v_1 = v0$ for some v and such that the string $v_1' = v1$ is also in B_1. If this were not possible then the infinite string $v\bar{1}$ would not have a prefix in B_1. Note that u_1 must also be a prefix of v_1'. Suppose $u_1 \mapsto u_2$ is a rule in φ_a and $v_1 \mapsto v_2$ and $v_1' \mapsto v_2'$ are rules in φ_b. Then these rules induce the cones maps $\lfloor u_1 \rfloor \mapsto \lfloor u_2 \rfloor$, $\lfloor v_1 \rfloor \mapsto \lfloor v_2 \rfloor$ and $\lfloor v_1' \rfloor \mapsto \lfloor v_2' \rfloor$ respectively. As both φ_a and φ_b induce the same homeomorphism of $\mathfrak{C}_2$, if $\lfloor u_1 \rfloor \mapsto \lfloor u_2 \rfloor$ then $\lfloor u_1 w \rfloor \mapsto \lfloor u_2 w \rfloor$ and hence u_2 is a prefix of v_2. The same argument applies to v_2' which must also have u_2 as a prefix. Therefore the cone map $\lfloor v_1 \rfloor \mapsto \lfloor v_2 \rfloor$ can be rewritten as $\lfloor u_1 w'0 \rfloor \mapsto \lfloor u_2 w'0 \rfloor$ for some w', and the cone map $\lfloor v_1' \rfloor \mapsto \lfloor v_2' \rfloor$ can be rewritten as $\lfloor u_1 w'1 \rfloor \mapsto \lfloor u_2 w'1 \rfloor$. Therefore φ_b contains the rules $u_1 w'0 \mapsto u_2 w'0$ and $u_1 w'1 \mapsto u_2 w'1$ and hence must be reducible, a contradiction. $\square$

Theorem 2.26. *The set of all prefix replacement maps under composition is a group.*

Proof. Let $\mathfrak{V}$ be the set of all prefix replacement maps and let the binary product $*$ represent composition of functions. Function composition is known to be associative. The identity homeomorphism is given by the prefix replacement map Φ_ε which maps every string in $\{0,1\}^\omega$ to itself. Suppose Φ is a prefix replacement map induced by a prefix replacement $\varphi : A_1 \to A_2$ which contains n number of rules $u_i \mapsto v_i$. As φ is a bijection its inverse exists $\varphi^{-1} : A_2 \to A_1$ from which one can build a prefix replacement map Π. The map formed by the composition $\Phi * \Pi$ acts on the cones $\lfloor u_i \rfloor$ by $\lfloor u_i \rfloor \overset{\Phi}{\mapsto} \lfloor v_i \rfloor \overset{\Pi}{\mapsto} \lfloor u_i \rfloor$, and likewise the composition $\Pi * \Phi$ acts on the cones $\lfloor v_i \rfloor$ by $\lfloor v_i \rfloor \overset{\Pi}{\mapsto} \lfloor u_i \rfloor \overset{\Phi}{\mapsto} \lfloor v_i \rfloor$. Thus $\Pi = \Phi^{-1}$. Therefore $(\mathfrak{V}, *)$ is a group. $\square$

The group $(\mathfrak{V}, *)$ is one way of describing R. Thompson's group V. Although the theorem above proves that $(\mathfrak{V}, *)$ is a group, it does not describe how to compose two prefix replacement maps when given as a collection of cone maps. In the next section we describe a useful way of describing the elements of $(\mathfrak{V}, *)$, using binary trees, which will provide an effective way to represent the group multiplication.

Group Actions

A group G is said to act on a set X if there exists a map $\psi : G \times X \to G$ such that the following two conditions hold for all $x \in X$;

1. $\psi(1_G, x) = x$ and,
2. $\psi(g, \psi(h, x)) = \psi(gh, x)$.

Lemma 2.27. *Thompson's group G acts on the cantor set $\mathfrak{C}_2$.*

Proof. The proof follows from the fact that elements of V are homeomorphisms of $\mathfrak{C}_2$. Therefore, there exists a natural group action $(\Phi, x) \mapsto \Phi(x)$. □

This natural action will appear again in Chapter 3 when we consider the dynamics of the action of Thompson's group V on $\mathfrak{C}_2$.

Tree pairs

Definition 2.28 (Tree Pairs)**.** *Suppose $\mathcal{D}$ and $\mathcal{R}$ are binary trees, both with n leaves. Let σ be a permutation from the group S_n. We call the triple $(\mathcal{D}, \sigma, \mathcal{R})$ a **tree pair**. The tree $\mathcal{D}$ we call the **domain tree** and the tree $\mathcal{R}$ the **range tree**. One interprets σ as a map between the leaves of domain and range trees, where the i^{th} leaf of $\mathcal{D}$ is being mapped to the $(i)\sigma^{th}$ leaf of $\mathcal{R}$.*

One draws tree pairs in the following way. Let $(\mathcal{D}, \sigma, \mathcal{R})$ be a tree pair. One first draws $\mathcal{D}$ on the left and $\mathcal{R}$ on the right. Then one sequentially labels the leaves of $\mathcal{D}$ from left to right with the numbers 1 through n. Finally, one labels the i^{th} leaf of $\mathcal{R}$ with the number $(i)\sigma^{-1}$. Consider the following example below where $\sigma = (1234)$. (We will explain the label on the middle arrow in the discussion below.)

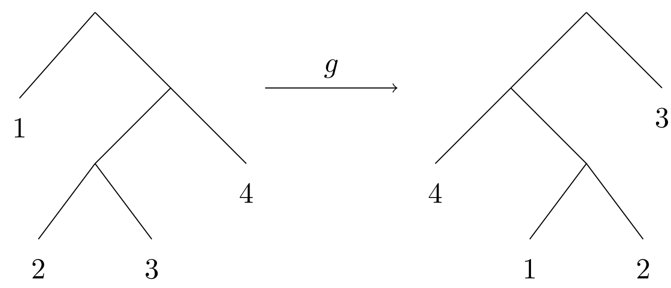


Figure 2.7: An example of a tree pair where $\sigma = (1234)$.

Tree pairs are an alternative way to represent prefix replacements. To prove this, we need the following lemma.

Lemma 2.29. *There exists a bijection between the set of all finite complete antichains and the set of all finite binary trees.*

Proof. Suppose T is a finite binary tree with n leaves. Each leaf in the tree has a unique binary string associated to it, defined by its descending path from the root. Let $A_T = \{l_1, l_2, \ldots, l_n\}$ be the set of the n binary strings associated to the n leaves of T. Suppose $l_i \preceq l_j$ for some i and j. Then the ith leaf of T must be the parent of the jth leaf which contradicts the definition of a leaf. Thus A_T must be an antichain. By our definition every binary tree is strongly binary, therefore every infinite descending path in $\mathcal{T}_2$ must pass through one of the nodes defined by the n leaves of T. Thus every infinite string in $\{0,1\}^\omega$ must have a unique prefix from A_T and hence A_T must also be complete.

Conversely, suppose A is a finite complete antichain of the poset $(\{0,1\}^*, \preceq)$ containing n strings. Each string in A uniquely defines a node in the infinite binary tree $\mathcal{T}_2$. As all these strings are pairwise incomparable, each node must lie in a unique path from the root. Therefore we can define a finite tree T_A as the rooted subtree in $\mathcal{T}_2$ whose leaves are exactly the nodes defined by A. To finish the proof we have to show that the tree T_A is strongly binary. As A is complete every infinite binary string has a unique prefix from A, this is equivalent to saying that every infinite descending path from $\mathcal{T}_2$ passes through exactly one of the nodes defined by A. Suppose T_A was not strongly binary. Then there would exists a node u in T_A that would only have one child. All infinite descending paths in $\mathcal{T}_2$ that passed through the child of u that was not in T_A would have no prefix from A and thus is a contradiction. Hence T_A must be strongly binary and the proof is complete. $\square$

Let $(\mathcal{D}, \sigma, \mathcal{R})$ be a tree pair. By Lemma 2.29, the trees $\mathcal{D}$ and $\mathcal{R}$ represent two finite complete antichains $A_1 = \{u_1, u_2, \ldots, u_n\}$ and $A_2 = \{v_1, v_2, \ldots, v_n\}$. Therefore the tree pair induces the prefix replacement $\varphi : A_1 \mapsto A_2$, where $(u_i)\varphi = v_{(i)\sigma}$.

Consider the tree pair in Figure 2.7. This tree pair gives rise to the following prefix replacement;

$$\begin{aligned} 0 &\mapsto 010 \\ 100 &\mapsto 011 \\ 101 &\mapsto 1 \\ 11 &\mapsto 00. \end{aligned}$$

In a tree pair we commonly label the arrow between the two binary trees with the element in V that the respective prefix replacement represents. In the example above the tree pair represents the following element g from V;

$$\begin{aligned} \lfloor 0 \rfloor &\mapsto \lfloor 010 \rfloor \\ \lfloor 100 \rfloor &\mapsto \lfloor 011 \rfloor \\ \lfloor 101 \rfloor &\mapsto \lfloor 1 \rfloor \\ \lfloor 11 \rfloor &\mapsto \lfloor 00 \rfloor. \end{aligned}$$

When considering the tree pair representative of g in Figure 2.7 we may use language such as "the element g takes the node 0 to the node 010". This language could imply that g is describing an automorphism of the infinite binary tree $\mathcal{T}_2$, however, this is not the case. What is meant by the sentence is that g defines a prefix replacement rule $0 \mapsto 010$, and nothing more. There is a sense in which an element v of V does induce a partial map from $\mathcal{T}_2$ to itself. However, it is only partial as there may be finitely many vertices at the top of the tree for which the map induced by v is not well defined. For example, consider the vertex 10 in $\mathcal{T}_2$. There is no well defined way in which the element g acts on the vertex 10.

In the same way that there are many prefix replacements that all induce the same prefix replacement map, so too there are many tree pairs that also represent a single element of V. Reducible prefix replacements can be easily identified from their equivalent tree pairs. Consider the tree pair below that has the same domain and range trees as Figure 2.7 but a different permutation on the leaves, in this case the identity permutation.

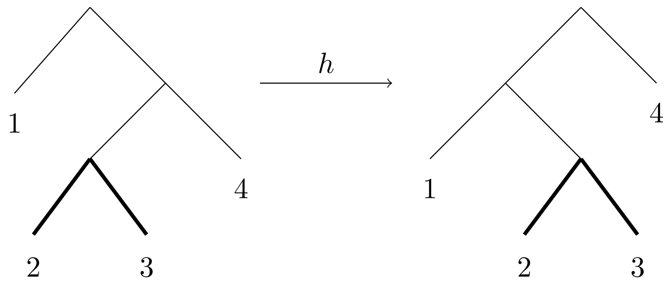


Figure 2.8: An example of a reducible tree pair with an exposed caret (in bold) consisting of the second and third leaves.

Notice how the leaves labelled 2 and 3 are both contained in a single caret in both the domain and range trees of the tree pair. We call this an **exposed caret**. In the language of prefix replacements, these represent the rules $100 \mapsto 010$ and $101 \mapsto 011$ which satisfy the conditions for reducible prefix replacement rules and thus can be replaced with the one rule $10 \mapsto 01$. This reduction is represented in the tree pair diagram by removing the exposed caret from both the domain and range trees, and adjusting the permutation to reflect the lower number of leaves. Any tree with an exposed caret we call **reducible**. In the figure below we have removed the exposed caret from the tree pair in Figure 2.8, notice that the tree pair that remains is in fact irreducible.

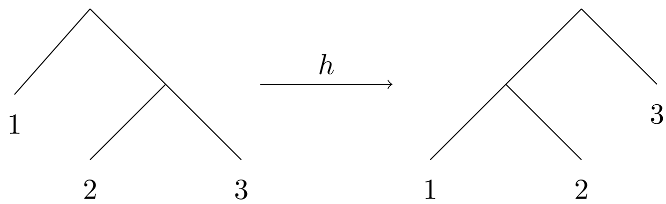


Figure 2.9: The unique irreducible tree pair representing the element h previously defined in Figure 2.8. Notice how we have removed the exposed caret and adjusted the permutation on the leaves accordingly.

As well as reducing a tree pair by removing an exposed caret, one can also **expand** a tree pair by adding one. To any tree pair one can expand the nth leaf by adding a caret to the nth leaf of the domain tree and to the $\sigma(n)$th leaf of the range tree. In the example above, we can create the tree pair in Figure 2.8 by adding caret to the second leaves of both the domain and range tree of Figure 2.9. Notice that as well as adding the extra carets one must also adjust the permutation accordingly too, so that the leaves that were being mapped to one another in the original tree are still getting mapped to one another in the new tree.

Tree pair multiplication

As well as representing the individual elements of V, one can use tree pairs to effectively multiply elements together. Recall that multiplication in V is composition of homeomorphisms, thus the method we use to multiply tree pairs must be well-defined in accordance with the group multiplication. We will begin by illustrating tree pair multiplication with an example. Consider the two elements g and h from Figure 2.7 and Figure 2.9 respectively. The order of multiplication matters, for this example we will be creating the product $\lambda = g * h$, where we use right actions. Thus this is equivalent to saying, "apply the homeomorphism g first, then follow it with h".

Written explicitly the two homeomorphisms are given in the table below by the canonical cone maps;

$\boldsymbol{g}$	$\boldsymbol{h}$
$\lfloor 0 \rfloor \mapsto \lfloor 010 \rfloor$	$\lfloor 0 \rfloor \mapsto \lfloor 00 \rfloor$
$\lfloor 100 \rfloor \mapsto \lfloor 011 \rfloor$	$\lfloor 10 \rfloor \mapsto \lfloor 01 \rfloor$
$\lfloor 101 \rfloor \mapsto \lfloor 1 \rfloor$	$\lfloor 11 \rfloor \mapsto \lfloor 1 \rfloor$
$\lfloor 11 \rfloor \mapsto \lfloor 00 \rfloor.$	

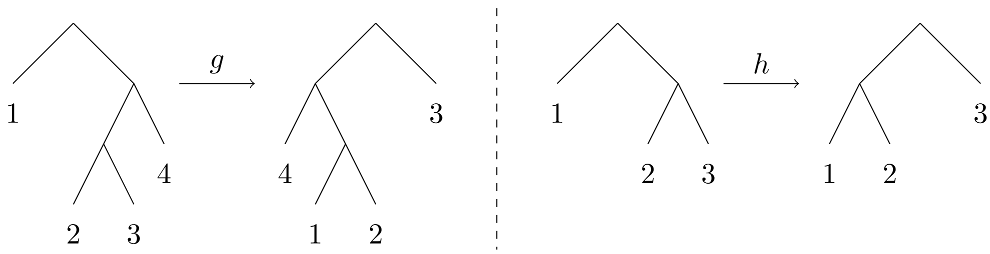
To compose two cone maps one requires that the range of the first map matches the range of the second, if this does not hold then the function composition is not well defined. For example the element g written above contains the cone map $\lfloor 0 \rfloor \mapsto \lfloor 010 \rfloor$, however the element h has no explicit map on the cone $\lfloor 010 \rfloor$. Instead we notice h contains the map $\lfloor 0 \rfloor \mapsto \lfloor 00 \rfloor$ which maps all these strings in $\mathfrak{C}_2$ with prefix "010" to those with prefix "0010". Thus implicit within h is the cone map $\lfloor 010 \rfloor \mapsto \lfloor 0010 \rfloor$. The first stage in creating the element $\lambda = g*h$ is then to rewrite the cone maps of g and h so that function composition is well defined between the cone maps, by which we mean for each cone in the range of g there exists the same cone in the domain of h. The rewriting occurs by repeatedly expanding the cone maps of the form $\lfloor u \rfloor \mapsto \lfloor v \rfloor$ to two of the form $\lfloor u0 \rfloor \mapsto \lfloor v0 \rfloor$ and $\lfloor u1 \rfloor \mapsto \lfloor v1 \rfloor$ until all the cone maps are as required. Below are the cone maps for g and h once this process has taken place, we have arranged the maps of h so that by reading across each row of the table one reads the composition of cone maps. All the cone maps in bold are those that we have created by expanding previous maps.

$$
\begin{array}{cc}
\boldsymbol{g} & \boldsymbol{h} \\
\lfloor 0 \rfloor \mapsto \lfloor 010 \rfloor & \boldsymbol{\lfloor \mathbf{010} \rfloor \mapsto \lfloor \mathbf{0010} \rfloor} \\
\lfloor 100 \rfloor \mapsto \lfloor 011 \rfloor & \boldsymbol{\lfloor \mathbf{011} \rfloor \mapsto \lfloor \mathbf{0011} \rfloor} \\
\boldsymbol{\lfloor \mathbf{1010} \rfloor \mapsto \lfloor \mathbf{10} \rfloor} & \lfloor 10 \rfloor \mapsto \lfloor 01 \rfloor \\
\boldsymbol{\lfloor \mathbf{1011} \rfloor \mapsto \lfloor \mathbf{11} \rfloor} & \lfloor 11 \rfloor \mapsto \lfloor 1 \rfloor \\
\lfloor 11 \rfloor \mapsto \lfloor 00 \rfloor & \boldsymbol{\lfloor \mathbf{00} \rfloor \mapsto \lfloor \mathbf{000} \rfloor}.
\end{array}
$$

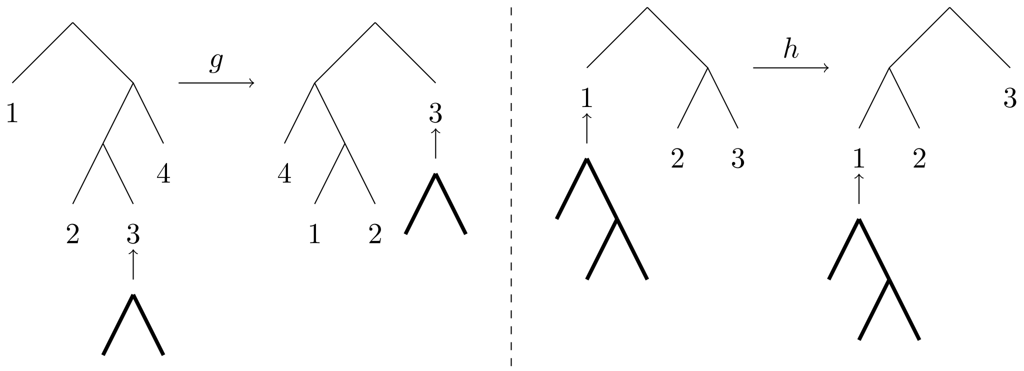
The element $\lambda = g * h$ is then given by the union of the cone maps created by composing the maps above;

$$
\begin{aligned}
&\boldsymbol{g * h} \\
\lfloor 0 \rfloor &\mapsto \lfloor 0010 \rfloor \\
\lfloor 100 \rfloor &\mapsto \lfloor 0011 \rfloor \\
\lfloor 1010 \rfloor &\mapsto \lfloor 01 \rfloor \\
\lfloor 1011 \rfloor &\mapsto \lfloor 1 \rfloor \\
\lfloor 11 \rfloor &\mapsto \lfloor 000 \rfloor.
\end{aligned}
$$

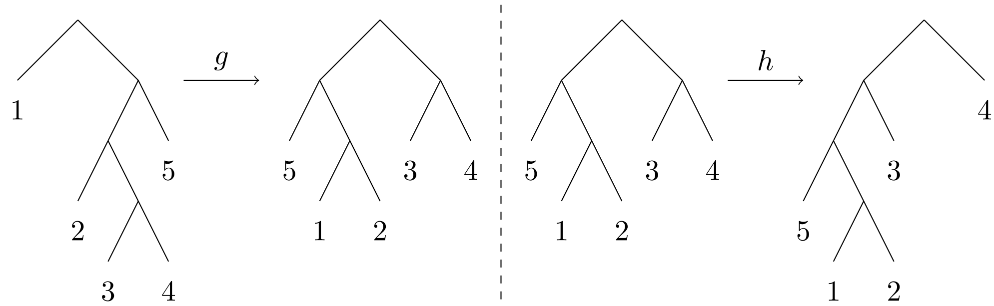
This process is more easily replicated using tree pairs. The two tree pairs representing g and h are given below.



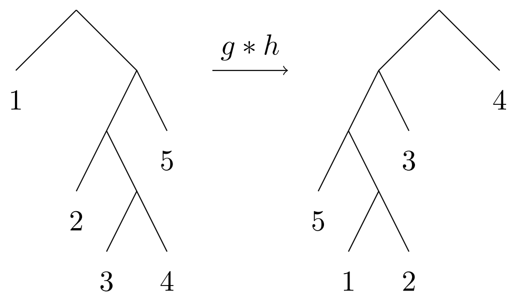
The process of rewriting the cone maps is the same as expanding the binary trees. To the tree pairs that represent g and h we add carets until the range tree of g matches the domain tree of h. Notice that any caret we add to the range tree of g we must also add to its domain tree, respecting the bijection between the leaves. Thus by adding a caret to the leaf labelled “3” in the range we must also add a caret to the leaf labelled “3” in the domain. Similarly any carets we attach to the domain tree of h we must also add to its range tree. In the case of h we are actually attaching a subtree made from two carets.



We must also adjust the permutation on the leaves to compensate for the number of leaves we add when we expand the trees. Each tree pair now has five leaves on each tree so the permutation is adjusted to reflect that. Notice below that we have also relabelled the domain tree of h so that the labels exactly match the labels on the range tree of g. Accordingly we have also adjusted the labels on the range tree of h so that leaves of the domain are being mapped to the appropriate leaves in the range.



The final step is to remove the two middle trees which are now identical and construct a new tree pair from the domain tree of g and the range tree of h. This new tree pair now represents the product $\lambda = g * h$.



We can summarise the process used in the example above in five steps.

1. Choose two tree pair representatives for g and h,

2. Add carets to each tree pair until the range tree of g matches the range tree of h (excluding the labels on the leaves),

3. Adjust the permutation on the leaves of each tree pair to account for the extra leaves added by the extra carets,

4. Rewrite the labels on the leaves of h so that its domain tree becomes identical to the range tree of g,

5. The product $g * h$ is then given by the domain tree of g and the range tree of h with the bijection on the leaves given by the pre-existing labels.

This process is the same for any elements g and h in V and provides a very visual way of multiplying elements of the group. The careful reader will have noticed that the first step involves a choice of tree pair representative being made for each element g and h. We do not give a proof here that multiplication is well defined regardless of what representative are chosen but the result is well known and follows from an application of Observation 2.23.

Parting a node of $\mathcal{T}_2$

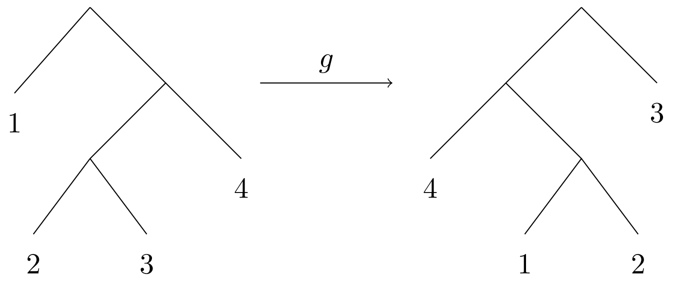
Given an element g in V one can define a natural partial bijection on the nodes of the infinite binary tree. Let $a_i \mapsto b_i$ be the unique set of reduced prefix replacement rules for g. Then there exists a bijection between the subsets $\{a_i w | \text{for all } a_i \text{ and } w \in \{0,1\}^*\} \subset \mathcal{T}_2$ and $\{b_i w | \text{for all } b_i \text{ and } w \in \{0,1\}^*\} \subset \mathcal{T}_2$ by $a_i w \mapsto b_i w$.

Notice that the bijection on the nodes of $\mathcal{T}_2$ induced by g is only partial, it is not defined on any node which is a *prefix* of some a_i or b_i. If we consider the unique reduced binary tree pair representative of g, it is exactly the set of *internal* nodes in either the domain or range tree on which the bijection is not defined. Suppose n is an internal node of the domain tree of the unique reduced tree pair representative T_g of g. Let $\{l_1, \ldots, l_m\}$ be the leaves of T_g underlying n. Then g acts on the subset $\lfloor n \rfloor$ by

$$g\lfloor n \rfloor = g\lfloor l_1 \rfloor \sqcup \ldots \sqcup g\lfloor l_m \rfloor.$$

As T_g is reduced each $g\lfloor l_i \rfloor$ must be disjoint, by which we mean there exists no $j \neq i$ such that $g\lfloor l_i \rfloor \sqcup g\lfloor l_j \rfloor$ is a cone. The subset of $\mathfrak{C}_2$ under each leaf has been separated from its neighbour under g. Therefore, we say that the node n has been **parted** under g. (Note, in [9] they use the term *splitting* instead. We reserve the word *splitting* for another definition in Chapters 4 and 5).

For example, consider the element g from Figure 2.7 which we have redrawn below.



The binary tree pair for g defines a prefix replacement $0 \mapsto 010$, therefore the induced bijection would map the node **0**110 to **010**110. However, for all the nodes in the binary tree pair that lie above the leaves of the domain and range trees g defines no bijection. For example, consider the node '10' in the domain tree of Figure 2.7. Under g its children get mapped to the following; $100 \mapsto 011$ and $101 \mapsto 1$, but g induces no well defined action on 10 itself. Therefore 10 has been parted under g, or we could say that g **parts** the node 10.

2.3.4 Thompson's groups F and T

Two subgroups of V are of particular importance as groups in their own right. The first is denoted by $\widehat{\mathbb{P}}$ by Thompson in his original work [31] but now is more commonly seen as F (see [12]). The group F is more commonly defined as a group of piecewise linear homeomorphisms of the closed unit interval $[0, 1]$. We define a **dyadic number** to be a number of the form $\frac{m}{2^k}$ for some integers k and m. We then define a **dyadic interval** to be an interval of the form

$$\left[\frac{m}{2^k}, \frac{m+1}{2^k}\right].$$

The closed unit interval $[0, 1]$ can be subdivided into unions of dyadic intervals, for example

$$[0, 1] = \left[0, \frac{1}{4}\right] \cup \left[\frac{1}{4}, \frac{1}{2}\right] \cup \left[\frac{1}{2}, 1\right].$$

Suppose

$$I_1 = d_1 \cup d_2 \cup \cdots \cup d_n \qquad I_2 = r_1 \cup r_2 \cup \cdots \cup r_n$$

are two subdivisions of $[0, 1]$ into n dyadic intervals. Then one can define a homeomorphism $f : [0, 1] \to [0, 1]$ that affinely maps each dyadic interval d_i in I_1 to the corresponding interval r_i in I_2 for every $1 \leq i \leq n$. This is an element of Thompson's group F.

Definition 2.30. *F is the group of piecewise linear homeomorphisms of the closed unit interval $[0, 1]$ that satisfy the following conditions;*

- *break points only ever appear on dyadic rationals,*

- *the derivative on any unbroken interval is a power of 2.*

As mentioned above, F appears as a subgroup of V. Consider the subgroup $\widehat{F}$ formed by taking all the elements of V which have a tree pair representative with the identity permutation on the leaves. Then $F \cong \widehat{F}$. A proof is given in [12]. We give some intuition behind the result. Consider the infinite binary tree $\mathcal{T}_2$. Associate to each node n in the tree a dyadic interval d_n such that $d_n = d_{n0} \cup d_{n1}$. Then the root is representative of the whole unit interval and the leaves of each finite binary tree represent a dyadic subdivision of $[0, 1]$. The action of each element of F on the unit interval then induces an action on the infinite binary tree as described by elements of $\widehat{F}$.

When we draw tree pairs representing element of F, we will forgo writing the permutation on the leaves of the diagram, leaving it implicit that the permutation is the identity. For the example in Figure 2.10 below it is implicit that the i^{th} leaf of the domain is getting mapped to the i^{th} leaf of the range.

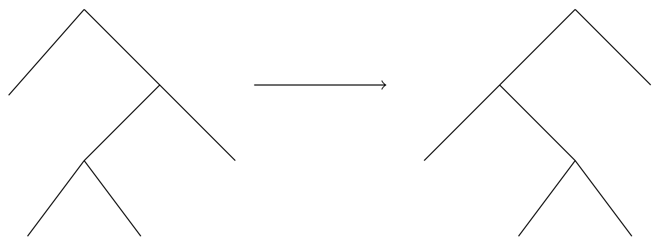


Figure 2.10: An example element of Thompson's group F

Thompson's group T, originally called $\widehat{\mathcal{C}}$ by Thompson in [31], is a group of piecewise linear homeomorphisms of the unit circle S^1. Consider S^1 as the interval $[0, 1]$ with the endpoints identified. Similarly to F, elements of T are homeomorphisms which map images of dyadic rationals to images of dyadic rationals and have finitely many break points. The derivatives on all unbroken intervals are powers of 2. As a subgroup in V it contains all the elements whose tree pair representations have a cyclic permutation on their leaves.

Chapter 3

The Demonstrable Groups for *V*

The work in this chapter is joint with Collin Bleak and much of the content appears in [2].

In this chapter we explore the relationship between groups defined by language theoretic properties and the dynamics of certain subgroups in Thompson's group V. As we have already discussed in Chapters 1 and 2, one can classify a group using the language type of its word problem. In 1977 Anatoly Anisimov provided an algebraic classification for regular groups [1].

Theorem 3.1 (Anisimov)**.** *A finitely generated group G is a regular group if and only if it is finite.*

It is a well known fact that Thompson's group V contains all the finite groups. However, there exist particular embeddings of finite groups into V whose natural action on the Cantor space $\mathfrak{C}_2$ exhibit interesting dynamical properties. To understand the result one first needs to know the definition of a **free action**.

Definition 3.2. *An group action $\psi(G \times X) \to X$ is* ***free*** *if and only if $\psi(g, x) \neq x$ for all non-trivial $g \in G$ and all $x \in X$. Thus, for any $x \in X$, if $(g, x) = x$ then $g = 1_G$.*

The following lemma was pointed out by Collin Bleak [6].

Lemma 3.3. *A finitely generated group G is finite if and only if there exists an embedding of G into V whose natural action is free on the whole Cantor space $\mathfrak{C}_2$.*

The subgroups described in Lemma 3.3 are dynamically interesting because by the definition of a free action, each non-trivial element moves every point of $\mathfrak{C}_2$. Therefore, by Theorem 3.1, we can characterise regular groups using a dynamical property of V. That is, a group is regular if and only if it is isomorphic to a finitely generated subgroup of V whose non-trivial elements move every point in $\mathfrak{C}_2$.

In this chapter we take the association between language theoretic properties of groups and the dynamics of subgroups of V a step further. The context-free groups, or $\mathcal{CF}$ groups, were classified by Muller and Schupp [24],[25].

Theorem 3.4 (Muller, Schupp)**.** *A finitely generated group G is a $\mathcal{CF}$ group if and only if it is virtually free.*

In this chapter we will provide a characterisation for the $\mathcal{CF}$ groups using dynamical properties of subgroups of V. The particular subgroups of V were first introduced by Bleak and Salazar-Diaz [9] and can informally be described as subgroups whose non-trivial elements act freely on an *open set* of $\mathfrak{C}_2$. These subgroups are called **demonstrative subgroups of** V, and the set of demonstrable subgroups is denoted by $\dot{\mathfrak{D}}_{V,\mathfrak{C}_2}$. Any group isomorphic to a demonstrative subgroup of V, is called **a demonstrable group for** V. We denote the class of demonstrable groups for V by $\mathfrak{D}_{V,\mathfrak{C}_2}$.

The main result of this chapter is as follows.

Theorem 3.5. *A group G is a $\mathcal{CF}$ group if and only if it is finitely generated and in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$.*

Theorem 3.5 is a parallel result to Lemma 3.3, where the condition placed upon the group G has been weakened so that its corresponding subgroup in V only has to act freely on an open set of $\mathfrak{C}_2$ rather than the whole space. Thus Theorem 3.5 continues the relationship between groups defined by language theoretic properties and groups defined by dynamics in V. If Lehnert's conjecture is true, the relationship would then include the *co*$\mathcal{CF}$ groups, which would be characterised by subgroups of V without any restriction on their dynamics.

3.1 Statement of results

The work in this chapter answers two questions proposed separately in two different papers. The first question arose in [3], in which the authors investigate a class of groups which were suggested as potential counterexamples to Lehnert's conjecture that V is a universal *co*$\mathcal{CF}$ group. They proved these groups were *co*$\mathcal{CF}$ and thus if any are shown to be non-embeddable into V then Lehnert's conjecture will have been proven to be false. We investigate these groups in far more detail in the final two chapters of this work.

In the course of [3] the authors ask a question regarding demonstrable groups for V.

Question 1.2 of [3]. *Does there exist a demonstrative embedding of F_2 into V?*

We answer this question here in the affirmative. Indeed, by using a further result of theirs, we show that all the countable *virtually* free groups are also demonstrable for V.

Theorem 3.6. *The countable virtually free groups are demonstrable for V.*

Some ramifications of Theorem 3.6 can understood by considering the work done by Holt, Rees, Röver and Thomas in [20]. In [20] they show that class of $co\mathcal{CF}$ groups is closed under four operations.

1. Passing to a finitely generated subgroup
2. Passing to a finite index overgroup
3. Taking finite direct products
4. Taking the restricted wreath product of a $co\mathcal{CF}$ group with a $\mathcal{CF}$ top-group.

If V is a universal $co\mathcal{CF}$ group then its set of finitely generated subgroups should also be closed under the same operations. The first three operations were already known to be satisfied by the set of finitely generated subgroups V. A theorem of Bleak and Salazar-Díaz in [9] states the following.

Theorem 1.2 [9] *If $G \leq V$ and H is a demonstrable group from $\mathfrak{D}_{V,\mathfrak{C}_2}$ then the restricted wreath product $G \wr H$ embeds into V.*

Recall that by Theorem 3.4 a finitely generated group is $co\mathcal{CF}$ iff it is virtually free. Therefore, by Theorem 3.6 we have the following corollary.

Corollary 3.7. *The finitely generated subgroups of V are closed under taking the restricted wreath product of a $co\mathcal{CF}$ group with a $\mathcal{CF}$ top-group.*

The second question we (partially) answer in this chaper was raised by Bleak and Salazar-Díaz in [9]. In their introduction they pose four questions, the third of which asks;

Question 3 of [9] (paraphrased): *Can one find a universal description of the demonstrable groups of V?*

In the second half of this chapter we prove the following theorem which answers the question above for finitely generated demonstrable groups.

Theorem 3.8. *If G is finitely generated and in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$, then G is a $\mathcal{CF}$ group.*

Therefore, the main result of this chapter, Theorem 3.5, follows as a consequence of Theorem 3.6 and Theorem 3.8.

3.2 Demonstrable groups

Definition 3.9. *Suppose H is a group that acts on a space X. We say that G is a* ***demonstrative subgroup*** *of H with respect to X if and only if there exists a non-empty open subset U in X such that for any two elements g_1 and g_2 in G if $g_1 \neq g_2$ then*

$$Ug_1 \cap Ug_2 = \emptyset.$$

We define $\dot{\mathfrak{D}}_{H,X}$ to be the set of all demonstrative subgroups of H with respect to X. A group G is in the class of **demonstrable** groups for H with respect to X, denoted by $\mathfrak{D}_{H,X}$, if and only if G is isomorphic to a subgroup of H in $\dot{\mathfrak{D}}_{H,X}$. The open set U used in the definition above we call a **demonstration set**, and is often not unique.

Example 3.10. Consider the group $H = (\mathbb{R}, +)$ and its subgroup $G = (\mathbb{Z}, +)$. Both of these groups act on the real line $\mathbb{R}$ in the usual way. Let $U = (a_1, a_2)$ be some open interval of $\mathbb{R}$ such that $a_2 - a_1 < 1$. Then for any z in G, z acts on U by $Uz = (a_1 + z, a_2 + z)$. Thus for z_1 and z_2 in G such that $z_1 \neq z_2$ the intersection $Uz_1 \cap Uz_2$ must be empty. Hence G is a demonstrative subgroup of H and the group $(\mathbb{Z}, +)$ is a demonstrable group in the class $\mathfrak{D}_{H,\mathbb{R}}$.

Recall that Thompson's group V has a natural action on the Cantor space $\mathfrak{C}_2$ (see Chapter 2). When we refer to the demonstrative subgroups of V we will always mean with respect its natural action on the space $\mathfrak{C}_2$. In [9] the authors give some isomorphism types of the demonstrable groups of V.

Lemma 1.1 of [9] *Let $\mathcal{A}$ be the smallest class of groups such that*

1. *$\mathcal{A}$ contains all finite groups,*
2. *$\mathcal{A}$ contains $\mathbb{Z}$,*
3. *$\mathcal{A}$ contains $\mathbb{Q}/\mathbb{Z}$,*
4. *$\mathcal{A}$ is closed under*
   (a) *isomorphism,*
   (b) *passing to a subgroup,*
   (c) *taking the direct product of a finite member with any other member.*

Then if G is in $\mathcal{A}$ then G is in $\mathfrak{D}_{V,\mathfrak{C}_2}$.

They go on to prove three embedding results around demonstrable groups, one of which is important for the main result of this chapter. We begin however with a result that gives us an alternative description for demonstrative groups.

Lemma 3.11. *A subgroup $G \leq V$ is demonstrative if and only if there exists a basic open set (cone) $\lfloor u \rfloor$ in $\mathfrak{C}_2$ such that $\lfloor u \rfloor \cap \lfloor u \rfloor g = \emptyset$ for all nontrivial g in G.*

Proof. Suppose G is a demonstrative subgroup of V. Then there exists an open set U that satisfies the properties of a demonstration set. The set U can be decomposed into the union of basic open sets. Suppose $\lfloor u \rfloor \subset U$. Therefore, the set $\lfloor u \rfloor$ must also be a demonstration set for G, that is, $\lfloor u \rfloor g_1 \cap \lfloor u \rfloor g_2 = \emptyset$ for all $g_1 \neq g_2$ in G. Therefore $\lfloor u \rfloor \cap \lfloor u \rfloor g = \emptyset$ for all g in G.

Suppose instead that G is a subgroup of V and $\lfloor u \rfloor$ is a cone such that $\lfloor u \rfloor \cap \lfloor u \rfloor g = \emptyset$ for all nontrivial g in G. We claim that G is demonstrative with demonstrative set $\lfloor u \rfloor$. Suppose g_1 and g_2 are elements of G such that $g_1 \neq g_2$ and $\lfloor u \rfloor g_1 \cap \lfloor u \rfloor g_2 = U \neq \emptyset$. Therefore $\lfloor u \rfloor \cap \lfloor u \rfloor g_2 g_1^{-1} \neq \emptyset$ which is a contradiction. Therefore $\lfloor u \rfloor g_1 \cap \lfloor u \rfloor g_2 = \emptyset$ for all $g_1 \neq g_2$ and thus G is a demonstrative subgroup of V. $\square$

If G is a demonstrative group for V with a demonstrative open set $\lfloor u \rfloor$ then we call u a **demonstrative node** for G.

Using this lemma we now sketch a proof of Theorem 1.2 in [9] which is important in proving the fourth closure property of Corollary 3.7.

Theorem 1.2 [9] *If $G \leq V$ and H is a demonstrable group from $\mathfrak{D}_{V,\mathfrak{C}_2}$ then the restricted wreath product $G \wr H$ embeds into V.*

Proof. We give a sketch of the proof that was given in more generality in [9]. It is known that Thompson's group V is a group that acts with **local realisation** on $\mathfrak{C}_2$ (see [9]). That is, given any open set $U \subset \mathfrak{C}_2$, there exists a subgroup $V_U \leq V$ such that $V_U \cong V$ and the support V_U is contained within U. Therefore for every cone $\lfloor u \rfloor$ and every subgroup G of V, there exists an isomorphic copy of G in V whose support is contained entirely in $\lfloor u \rfloor$. We call this subgroup G_u.

Let G and H be subgroups of V such that H is demonstrative with a demonstration node u. Let G_u be the subgroup of V isomorphic to G whose support is completely contained within $\lfloor u \rfloor$. For each $h \in G$ define the subgroup $G_u^h := \{h^{-1}gh : g \in G\}$, which isomorphic to G. As H is a demonstrative subgroup $G_u \cap G_u^h = \emptyset$ for all non-trivial $h \in H$ and thus for all $h_1 \neq h_2$, $G_u^{h_1}$ and $G_u^{h_2}$ have disjoint support. Therefore $\langle G_u, H \rangle \cong G \wr H$. $\square$

3.3 The countable virtually free groups are in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$

By the end of this section we will have proven Theorem 3.6, that the countable virtually free groups are demonstrable for V. However the main content of the section will be taken up with proving a smaller result.

Lemma 3.12. *The modular group $\Gamma \cong C_2 * C_3$ is contained in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$.*

As the class of demonstrable groups is closed under passing to subgroups, the lemma implies that any countable free group admits a demonstrative embedding into V. In [3] the authors show that the class of demonstrable groups for V is closed under passage to finite index overgroups, which will then give us the main result.

Free products

We first formally define the free product of groups $A * B$, basing our definition on the one given in Chapter 2 of [10]. Suppose A and B are groups such that $A \cap B = \{1\}$. We define a **reduced word** over the set $A \cup B$, to be a string $g_1 g_2 g_3 \ldots g_n$, where $g_i \in (A \cup B) \setminus \{1\}$ such that for all $1 \leq i \leq n-1$ the elements g_i and g_{i+1} do not lie in the same group. The reduced word of the identity element is represented by the empty string ε.

For the reduced words $x = g_1 \ldots g_n$ and $y = h_1 \ldots h_m$ we define the product $x \cdot y$ by

$$x \cdot y = \begin{cases} g_1 \ldots g_n h_1 \ldots h_m & \text{if } g_n \text{ and } h_1 \text{ are in opposite groups} \\ g_1 \ldots g_{n-1} z h_2 \ldots h_m & \text{if } g_n \text{ and } h_1 \text{ are both in the same group} \\ & \text{and } g_n h_1 = z \neq 1 \\ (g_1 \ldots g_{n-1}) \cdot (h_2 \ldots h_m) & \text{if } g_n \text{ and } h_1 \text{ are both in the same group} \\ & \text{and } g_n h_1 = 1. \end{cases}$$

Under the multiplication above the set of reduced words form a group. The inverse of the reduced word $x = g_1 g_2 \ldots g_n$ is the reduced word $x^{-1} = g_n^{-1} \ldots g_2^{-1} g_1^{-1}$. We call this group the free product of A and B and denote it by $A * B$.

We introduce a function $len : A * B \to \mathbb{N}_0$ that takes a reduced word and outputs the number of elements in the string, that is, $len(g_1 \ldots g_n) = n$. Informally we call n the length of the reduced word.

There also exists an alternative definition for the free product using group presentations.

Lemma 3.13. *Suppose $A \cong \langle X | R \rangle$ and $B \cong \langle Y | S \rangle$ are two groups. Then the group*

$A * B$ has presentation

$$A * B \cong \langle X \sqcup Y | R \sqcup S \rangle.$$

3.3.1 Embedding Γ into V

The modular group Γ is the group of Möbius transformations of the upper half complex plane which have the form

$$z \mapsto \frac{az+b}{cz+d}$$

such that $ad - bc = 1$. For our purposes there are only two properties of Γ that are important. Firstly, that it is isomorphic to the free product of the cyclic group of order two and the cyclic group of order three, namely $\Gamma \cong C_2 * C_3$. Secondly, that it contains the free group on two generators as a subgroup.

The cyclic group of order m is given by the presentation $\langle x | x^m \rangle$. Therefore the group $C_2 * C_3$ has presentation $\langle \alpha, \beta | \alpha^2, \beta^3 \rangle$ where subgroups $\langle \alpha \rangle$ and $\langle \beta \rangle$ are isomorphic to C_2 and C_3 respectively. The purpose of this section is to find a subgroup of V that is isomorphic to $C_2 * C_3$. In the section following this we will show that this subgroup is demonstrative.

We begin with two elements from Thompson's group V, which we call a and b, defined by the two tree pairs below.

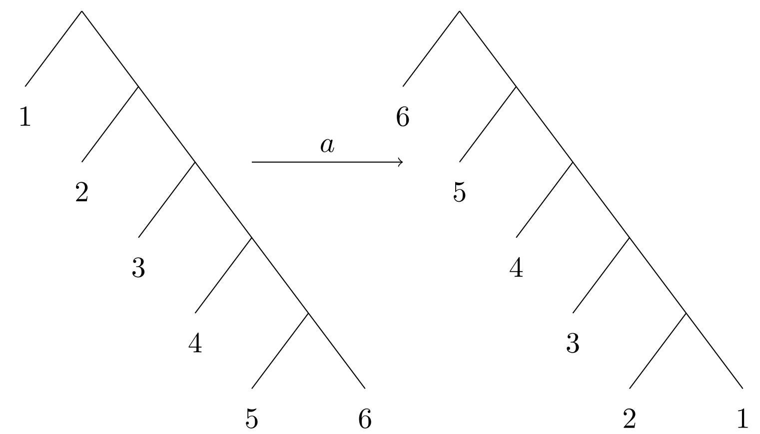


Figure 3.1: The tree pair representative $(\mathcal{D}_a, \sigma_a, \mathcal{R}_a)$ for $a \in V$

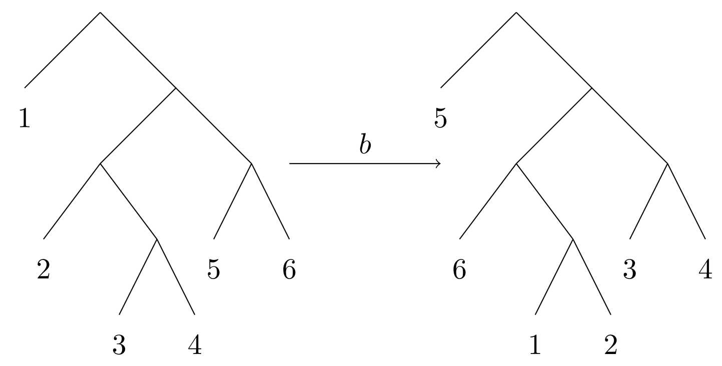


Figure 3.2: The tree pair representative $(\mathcal{D}_b, \sigma_b, \mathcal{R}_b)$ $b \in V$

We denote the tree pairs in Figure 3.1 and Figure 3.2 by $(\mathcal{D}_a, \sigma_a, \mathcal{R}_a)$ and $(\mathcal{D}_b, \sigma_b, \mathcal{R}_b)$ respectively. Notice that for each tree pair the domain tree is identical to the range tree, that is $\mathcal{D}_a = \mathcal{R}_a$ and $\mathcal{D}_b = \mathcal{R}_b$. Thus both a and b are of finite order, determined by their respective permutations. The permutation on the leaves of $(\mathcal{D}_a, \sigma_a, \mathcal{R}_a)$ is the product of transpositions $\sigma_a = (1,6)(2,5)(3,4)$, thus $\sigma_a^2 = 1$ and the order of a is two. The permutation on the leaves of $(\mathcal{D}_b, \sigma_b, \mathcal{R}_b)$ is the product of two disjoint three cycles $\sigma_b = (1,3,5)(2,4,6)$ and thus the order of b is three. Let $G = \langle a, b\rangle$ be the subgroup of V generated by the elements a and b.

Lemma 3.14. *The subgroup $G = \langle a, b\rangle \leq V$ factors as $\langle a\rangle * \langle b\rangle$.*

To prove Lemma 3.14 we use Fricke and Klein's well known criterion, the Ping-Pong Lemma. The version we give here is based on the one found in [15].

Lemma 3.15 (Ping-Pong Lemma)**.** *Let G be a group acting on a set X and let A and B be two subgroups of G such that $|A| \geq 2$ and $|B| \geq 3$. Suppose there exist two non-empty subsets X_A and X_B such that the following three conditions hold*

1. *$X_A \not\subset X_B$*
2. *for all non-trivial $a \in A$, $(X_B)a \subset X_A$*
3. *for all non-trivial $b \in B$, $(X_A)b \subset X_B$.*

Then $\langle A, B\rangle \cong A * B$.

Proof of Lemma 3.15. Let G be a group acting on a set X and let A and B be two subgroups of G that satisfy the properties described in the lemma. To prove the result we need to show that any reduced word in $\{A, B\}^*$ is non-trivial in $\langle A, B\rangle$. Consider the reduced word $w_1 = b_1a_1b_2a_2 \ldots b_n$ in $\{A, B\}^*$ where $a_i \in A$ and $b_i \in B$. Then w_1 acts on the set X_A and we have the following sequence of containments;

$$(X_A)b_1a_1b_2a_2 \ldots b_n \subset (X_B)a_1b_2a_2 \ldots b_n \subset (X_A)b_2a_2 \ldots b_n \subset \ldots \subset (X_A)b_n \subset (X_B).$$

As $X_A \not\subset X_B$ the element represented by w_1 must have acted on X_A in a non-trivial way and thus cannot be trivial.

Let $w_2 = a_1b_1a_2b_2 \ldots a_n$ be a reduced word and consider $b^{-1}w_2b$ where $b \in B \backslash \{1\}$. Then $b^{-1}w_2b$ is a word in the same form as w_1 and hence is non-trivial by the same argument. Therefore w_2 must also be non-trivial. Let $w_3 = a_1b_1a_2b_2 \ldots a_nb_n$ be a reduced word and consider $c^{-1}w_3c$ where $c \in B \setminus \{1, b_n^{-1}\}$. Then $c^{-1}w_3c$ reduces to a word in the same form as w_1 and thus by the same argument as before w_3 must be non-trivial. Finally let $w_4 = b_1a_1b_2a_2 \ldots b_na_n$ be a reduced word and consider $d^{-1}w_4d$ where $d \in B \setminus \{1, b_1\}$. Then $d^{-1}w_4d$ reduces to a word in the same form as w_1 and thus by the same argument as before w_4 must also be non-trivial. As every non-trivial reduced word in $\{A, B\}^*$ has the same form as either w_1, w_2, w_3 or w_4 we conclude that they all represent non-trivial elements in $\langle A, B\rangle$ and hence $\langle A, B\rangle \cong A * B$. $\square$

Proof of Lemma 3.14. We use Lemma 3.15 to prove the result. First we identify the subgroups $\langle a\rangle$ and $\langle b\rangle$ with the groups A and B from the lemma. Define the subsets $X_A = \lfloor 111\rfloor$ and $X_B = \lfloor 10\rfloor$.

Immediately we see that $X_A \not\subset X_B$ and the first condition of the lemma is satisfied. Now observe the action of a on the set X_B;

$$(X_B)a = (\lfloor 10\rfloor)a = \lfloor 11110\rfloor \subset \lfloor 111\rfloor = X_A.$$

This confirms the third requirement. In $\langle b\rangle$ there are two non-trivial elements, b and b^{-1}. Observe $(X_A)b = (\lfloor 111\rfloor)b = \lfloor 100\rfloor \subset \lfloor 10\rfloor = X_B$, and $(X_B)b^{-1} = (\lfloor 111\rfloor)b^{-1} = \lfloor 1011\rfloor \subset \lfloor 10\rfloor = X_B$. Thus the last of the three conditions in Lemma 3.15 is met and thus $\langle a, b\rangle$ factors as $\langle a\rangle * \langle b\rangle$. $\square$

By Lemma 3.14 there exists a copy of the group $C_2 * C_3 \cong \langle \alpha, \beta | \alpha^2, \beta^3\rangle$ in V given by the embedding $\alpha \mapsto a$ and $\beta \mapsto b$. We now go on to show that this subgroup in V is demonstrative.

3.3.2 Proving that $C_2 * C_3$ is demonstrable for V

In what follows we will prove that the subgroup G is demonstrative in V with demonstration set $\lfloor 0\rfloor$. As G is the free product $\langle a\rangle * \langle b\rangle$, each element $g \in G$ can be written by a unique reduced word. The proof that G is a demonstrative subgroup for V, begins with an induction on the length of the reduced words over $\{a\} \cup \{b, b^{-1}\}$.

Lemma 3.16. *For every non-trivial $g \in G$,*

$$\lfloor 0\rfloor \cap \lfloor 0\rfloor g = \emptyset$$

Proof. The proof will proceed on the length of reduced words in G.

Suppose g is a reduced word such that $len(g) = 1$. There are three options, namely a, b and b^{-1}. Suppose $g = a$, then $\lfloor 0 \rfloor g = \lfloor 11111 \rfloor$. Further, if $g = b$, then $\lfloor 0 \rfloor g = \lfloor 1010 \rfloor$. Finally suppose $g = b^{-1}$, then $\lfloor 0 \rfloor g = \lfloor 110 \rfloor$. Thus for all reduced words of length one in G the lemma holds.

We now consider all elements in G with reduced length greater than one. Let $\mathcal{P}(n)$ be the statement

$$\lfloor 0 \rfloor g \subseteq \begin{cases} \lfloor 111 \rfloor, & \text{if } g \text{ ends with generator } a \\ \lfloor 10 \rfloor, & \text{if } g \text{ ends with either of the generators } b \text{ or } b^{-1} \end{cases}$$

for all reduced words g over $\{a\} \cup \{b, b^{-1}\}$, such that $len(g) = n \geq 2$.

We will proceed to prove by induction that $P(n)$ holds for all $n \geq 2$. Suppose $g \in G$ such that $len(g) = 2$. There are four options, namely, ab, ab^{-1}, ba and $b^{-1}a$. Suppose $g = ab$, then $\lfloor 0 \rfloor g = \lfloor 10011 \rfloor \subset \lfloor 10 \rfloor$. Suppose $g = ab^{-1}$, then $\lfloor 0 \rfloor g = \lfloor 101111 \rfloor \subset \lfloor 10 \rfloor$. Suppose $g = ba$, then $\lfloor 0 \rfloor g = \lfloor 1111010 \rfloor \subset \lfloor 111 \rfloor$. Finally suppose $g = b^{-1}a$, then $\lfloor 0 \rfloor g = \lfloor 1110 \rfloor \subset \lfloor 111 \rfloor$. Therefore $\mathcal{P}(2)$ holds.

Suppose $\mathcal{P}(k)$ is true for all reduced words g with $len(g) = k$ such that $2 \leq k \leq n$. Now suppose h is a reduced word such that $len(h) = n + 1$. Let h' be the prefix of length n in h.

Suppose h' ends with a, then as h is a reduced word there are two possibilities, either $h = h'b$ or $h = h'b^{-1}$. As $len(h') = n$, by our inductive assumption $\lfloor 0 \rfloor h' \subseteq \lfloor 111 \rfloor$. Thus if $h = h'b$, then $\lfloor 0 \rfloor h = \lfloor 0 \rfloor h'b \subseteq \lfloor 111 \rfloor b = \lfloor 100 \rfloor$ and $\mathcal{P}(n+1)$ is true. Suppose $h = h'b^{-1}$, then $\lfloor 0 \rfloor h = \lfloor 0 \rfloor h'b^{-1} \subseteq \lfloor 111 \rfloor b^{-1} = \lfloor 1011 \rfloor$ and again $\mathcal{P}(n+1)$ is true.

Suppose instead that h' ends with either b or b^{-1}. Then there is only one possibility for h, namely $h = h'a$. As $len(h') = n$, by our inductive assumption $\lfloor 0 \rfloor h' \subseteq \lfloor 10 \rfloor$. Thus $\lfloor 0 \rfloor h = \lfloor 0 \rfloor h'a \subseteq \lfloor 10 \rfloor a = \lfloor 11110 \rfloor$ and $\mathcal{P}(n+1)$ is true.

Therefore for all reduced words h such that $len(h) = n + 1$, the statement $\mathcal{P}(n+1)$ is true, and therefore by induction $\mathcal{P}(n)$ must be true for all reduced words g such that $len(g) \geq 2$. Therefore, the lemma holds for all elements of G. $\square$

Therefore, by Lemma 3.11, we have the following corollary.

Corollary 3.17. *The subgroup G is a demonstrative subgroup of V*

Therefore this proves Lemma 3.12, that the modular group $\Gamma \cong C_2 * C_3$ is a demonstrable group in $\mathfrak{D}_{V,\mathfrak{C}_2}$.

3.3.3 Proof of Theorem 3.6

Bleak and Salazar-Díaz observe the following in Lemma 3.2 [9].

Observation 3.18 (Bleak, Salazar-Díaz)**.** *Suppose that G is a demonstrative group with m serving as a demonstration node. Then given any subgroup $H \leq G$, H is also demonstrative with demonstration node m.*

This observation, together with the fact that the free group on two generators F_2 is isomorphic to the subgroup $\langle [a, b], [a, b^{-1}] \rangle \leq G$ (where the bracket $[x, y]$ represents the commutator $x^{-1}y^{-1}xy$), implies the following corollary.

Corollary 3.19. *The free group on two generators, F_2, is in the class $\mathfrak{D}_{(V,\mathfrak{C}_2)}$*

Virtually free groups are groups that contain a free group as a finite index subgroup. While it is known (see [26, 9]) that if a group G embeds in V, then any finite index over-group of G also embeds into V, the paper of Berns-Zieve et al [3], extends this with Theorem 3.3, which we paraphrase below.

Theorem 3.20 (Theorem 3.3 in Berns-Zieve et al [3])**.** *Suppose G is a group which embeds in R. Thompson's group V. If $G \leq H$ where $[H : G] = m$, for some $m \in \mathbb{N}$ and G embeds as a demonstrative subgroup in V, then H also embeds as demonstrative subgroup of V.*

The theorem tells us that $\mathfrak{D}_{(V,\mathfrak{C}_2)}$ is closed under taking finite index overgroups. As all countable free groups embed into F_2 and since virtually free groups are, by definition, finite index overgroups of free groups, by Corollary 3.19 countable virtually free groups are contained within $\mathfrak{D}_{(V,\mathfrak{C}_2)}$. This proves Theorem 3.6.

3.4 Finitely generated groups in $\mathfrak{D}_{V,\mathfrak{C}_2}$ are virtually free

In this final section we will prove Theorem 3.8, stated at the beginning of the chapter but repeated here for convenience.

Theorem 3.8 *If G is finitely generated and in the class $\mathfrak{D}_{V,\mathfrak{C}_2}$, then G is a co$\mathcal{CF}$ group.*

To prove our theorem we must be able to construct a push-down automaton that accepts the word problem for a given finitely generated demonstrative group. (See Definition 2.9 in Chapter 2 for the definition of a push-down automaton.)

The section will proceed as follows. We begin with a motivating example which demonstrates how we use the property of being demonstrative in V to create a push down automaton accepting the group's word problem. Subsequent to the example, we then give the procedure for creating a push down automaton which accepts the word problem for an arbitrary demonstrative subgroup of V.

The example below is taken from [9] where they give a demonstrative embedding of $\mathbb{Z}$ generated by the element g given in Fig. 3.3. The demonstrative node n is at address 0. Recall that we will abuse notation slightly and say that $n = 0$.

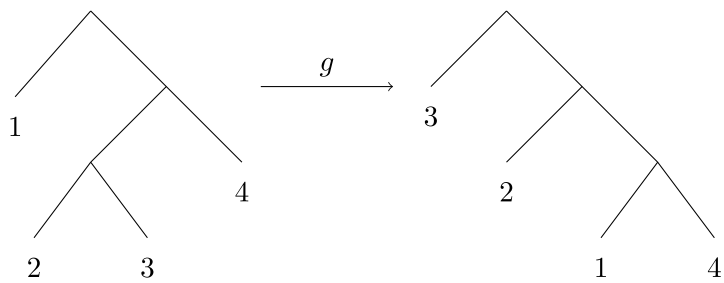


Figure 3.3: The generator of a demonstrative copy of $\mathbb{Z}$ inside V. The demonstrative node n has address 0.

The subgroup $G = \langle g \rangle$ is isomorphic to the group $\mathbb{Z}$ given by presentation $\langle g | \emptyset \rangle$. We abuse notation by using the g to represent the element of the subgroup G and the formal symbol in the presentation $\langle g | \emptyset \rangle$. Where confusion may arise we will always clarify which definition of g we are using.

The property that G is a demonstrative subgroup of V can be seen by considering where the elements of the group map the cone $\lfloor 0 \rfloor$. Notice that any element of the form g^p, for some positive integer p, takes the node 0 to the node 1^p10. Conversely, notice that any element of the form g^{-p} takes the node 0 to the node 10^p1. Thus $\lfloor 0 \rfloor \cap \lfloor 0 \rfloor z = \emptyset$ for all non-trivial z in G. It is already known that $\mathbb{Z}$ is a $\mathcal{CF}$-group so our example gives us no new result, but the method we use below to create the push-down automaton that accepts the word problem of $G = \langle g | \emptyset \rangle$, can be generalised for every finitely generated demonstrative subgroup of V.

Let $\mathcal{A}$ be our PDA that accepts the word problem of G. $\mathcal{A}$ has three states $\{q_0, q_r, q_a\}$ where q_0 is the start state and q_a is the only accept state. The alphabet Λ of the automaton is the set $\Lambda = \{g, g^{-1}\}$, consisting of the generator and its formal inverse symbol from the presentation $\langle g | \rangle$. Our stack alphabet is the set $\Gamma = \{\#, 0, 1\}$, where $\#$ is a special bottom-of-the-stack symbol. $\mathcal{A}$ is defined by the transition table given by Table 3.1 below.

Current State	Input	Stack Top	Stack Replacement	New State
q_0	ϵ	$\emptyset$	$0\#$	q_a
q_a	g	0	110	q_r
q_a	g^{-1}	0	101	q_r
q_r	g	0	110	q_r
q_r	g	100	10	q_r
q_r	g	11	111	q_r
q_r	g^{-1}	0	101	q_r
q_r	g^{-1}	10	100	q_r
q_r	g^{-1}	111	11	q_r
q_r	g	$101\#$	$0\#$	q_a
q_r	g^{-1}	$110\#$	$0\#$	q_a

Table 3.1: The transition table of the automaton accepting the word problem of $G \cong \mathbb{Z}$

We also provide a visual representation of $\mathcal{A}$ in Fig. 3.4.

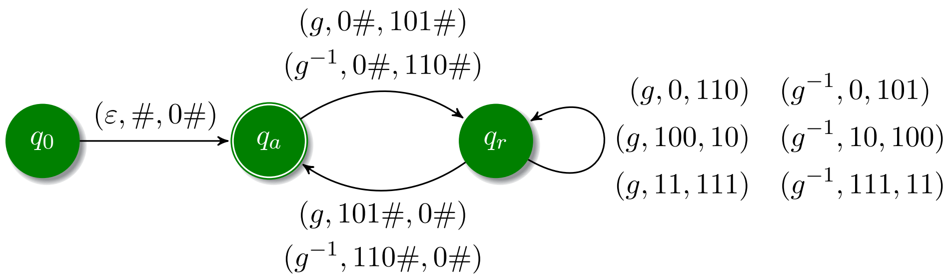


Figure 3.4: A graphical representation of the automata $\mathcal{A}$ that accepts the word problem of G.

The automaton attempts to model the action of the generators on the demonstrative node 0. The stack will represent the location of the demonstrative node under the action of the element defined by the word read by the automaton so far. Whenever the automaton $\mathcal{A}$ processes a letter g or g^{-1}, it amends the stack according to the prefix replacement rules defined for the appropriate element g or g^{-1} in G.

The automaton $\mathcal{A}$ begins with active state q_0 by loading the stack with the address of the demonstrative node $n = 0$, and moving the active state to the accept state q_a. Note that none of the input string is read at this time. Whenever the active state is q_a, if $\mathcal{A}$ has finished reading the input then it accepts the word. However, if the active state is q_a and there are still more letters to be read then $\mathcal{A}$ will process the next letter, the action of which will move the active state to q_r and modify the stack according to the prefix replacement rules. From the state q_r there are circumstances which allow the active state to return to q_a. Namely, whenever the active state is q_r and $\mathcal{A}$ processes a letter and the resultant stack is "0#", then the active state transitions to q_a.

By the definition of demonstration nodes, a demonstration node under the action of an element w of the demonstrative group is taken to itself if and only if w is the identity element. By construction, our automata has stack "0#" only when the previously processed word represents the trivial element. However, this is precisely at the times that the automaton's active state is q_a.

We now generalise this method into a proof of Theorem 3.8. Note that our choice of demonstrative node is important. It is possible to choose a demonstrative node that gets parted by elements of G. Recall from Chapter 2 that an element $g \in V$ **parts** a node p if and only if g does not induce a prefix replacement rule of the form $p_1 \mapsto q_1$ for some prefix p_1 of p and some $q \in \{0,1\}^*$. We will show that for the automaton to work we need to choose a node that will not get parted under any element of G. The following lemma shows that such a node will exist.

Lemma 3.21. *Let G be a finitely generated, demonstrative subgroup of V, with demonstrative node n. Then there exists a depth $d \in \mathbb{N}$ such that a cone $\lfloor m \rfloor \subset \lfloor n \rfloor$ gets parted if and only if depth$(m) \leq d$.*

Proof. Let $G = \langle X \rangle$ be a finitely generated, demonstrative subgroup of V, with demonstrative node n and generating set X. For each generator $x_i \in X^{\pm}$ there exist finitely many nodes which are parted by x_i. Only those in the interior of the domain tree of the irreducible representative of x_i have the potential to be parted by the element. Let N be the finite set of all nodes parted by the generators of G. Suppose the element $g \in G$ parts a cone $\lfloor m \rfloor$. Consider g decomposed as a product of generators $g = x_1 x_2 \ldots x_p$. Then as the product $x_1 x_2 \ldots x_p$ acts on the cone $\lfloor m \rfloor$, each x_{i+1} acts on the set $\lfloor m \rfloor x_1 x_2 \ldots x_i$ for all $1 \leq i < p$. As g parts $\lfloor m \rfloor$ there must exist a minimal $q \leq p$ such that x_q parts the cone $\lfloor m \rfloor x_1 \ldots x_{q-1}$. As x_q is a generator, the cone $[a] = \lfloor m \rfloor x_1 \ldots x_{q-1}$ must be in the set N. We say that the decomposition $g = x_1 x_2 \ldots x_p$ parts the cone $\lfloor m \rfloor$ at $[a]$ with the subproduct $g' = x_1 x_2 \ldots x_q$.

Let S_k be the set of all such cones $\lfloor m \rfloor \subset \lfloor n \rfloor$ of depth k that get parted by some element of G. Suppose for a contradiction that for every $k \in \mathbb{N}$ there exists $k' > k$ such that $S_{k'}$ is non-empty. Let $[m_1] \in S_k$ for some k. Then by our assumption there exists some $[m_2] \in S_{k'}$ where $k' > k$. Suppose g and h are elements of G that part $[m_1]$ and $[m_2]$ respectively for the decompositions $g = x_1 x_2 \ldots x_s$ and $h = y_1 y_2 \ldots y_t$, where $x_i, y_i \in X$. As N is finite, by the pigeon-hole principle it must be possible to find such $[m_1], [m_2], g$ and h such that g and h part $[m_1]$ and $[m_2]$ respectively at the same

$[a] \in N$ for the decompositions given above. Thus there exists $q \leq s$ and $r \leq t$ such that $g' = x_1x_2 \dots x_q$, $h' = y_1y_2 \dots y_r$ and $[m_1]g' = [m_2]h'$. However, as $[m_1], [m_2] \subset \lfloor n \rfloor$, this implies $\lfloor n \rfloor g \cap \lfloor n \rfloor h \neq \emptyset$. Observe that g and h must be distinct as $[m_1] \neq [m_2]$. Therefore, as $\lfloor n \rfloor$ is a demonstrative node this is a contradiction by Definition 3.9. $\square$

We can now go on to describe how to construct a PDA that accepts the word problem of a demonstrable group of V.

Proof of Theorem 3.8. Suppose $\widehat{G}$ is a finitely generated, demonstrable group, isomorphic to a demonstrative subgroup G of Thompson's Group V where G is finitely generated with set of generators $X = \{g_1, g_2, \dots, g_m\}$. Suppose $n \in \{0,1\}^*$ is a demonstrative node for G that does not get parted by any element of G. We describe and construct our automaton $\mathcal{A}$ below.

Set the input alphabet to be $\Lambda := X^{\pm}$, the set of generators for G and their inverses. We note, as we did for the example with $\mathbb{Z}$, that the symbols $g_i \in \Lambda$ are formal symbols from an alphabet, not group elements, and we will make the distinction where confusion may arise. Set $\Gamma = \{\#, 0, 1\}^*$ to be the stack alphabet. The new automaton $\mathcal{A}$ will also have three states q_0, q_a and q_r, where q_a is the automaton's only accept state. We will describe the transitions from each of these states.

Loading phase: The automaton $\mathcal{A}$ begins in the state q_0. That state admits one transition, which loads the stack with the string $n\# \in \Gamma^*$, and transfers the active state to q_a, without reading any of the input. After this transition, the stack will contain the address of the demonstration node n and the bottom-of-the-stack symbol, with n written from top to bottom on the stack. (For example if $n = 100$ then 1 would be at the top of the stack followed by two 0's and finally #.) We call this the *loading phase.*

Reading phase: After the loading phase, $\mathcal{A}$ enters the *reading phase*, where it begins to read the input string from Λ^*. Each g in the input alphabet represents a generator from the group. That generator has a unique minimal tree pair representative g_T which defines the homeomorphism. The domain and range trees of g_T define two antichains, $\{s_1, s_2, \dots, s_j\}$ and $\{t_1, t_2, \dots, t_j\}$ where $(s_i)g_T = t_i$. These will define our transitions in the automaton.

Transitions from q_0 and q_a: When the automaton leaves the state q_0 after the loading phase, the symbols on the stack will be exactly "$n\#$". For each input letter g there exists a unique prefix n' of n contained within the domain antichain. Therefore from q_a we define transitions to the state q_r by the tuples of the form *(g, n', $(n')g_T$)* for each g in the input alphabet. For each input letter g the automaton removes the string n' from the top of the stack and replaces it with the string $(n)g_T$ and moves the active state to q_r. This will always be well defined as n is not parted by any element of the group G.

Transitions from q_r All transitions from q_r take the active state to either q_r or to q_a. We first add transitions from q_r to q_r given by the tuples $(g, s_i, (s_i)g_T)$. We then add transitions from q_r to q_a of the form $(g, s_i w\#, (s_i)g_T w\#)$ if $(s_i)g_T w = n$. This means that the automaton will only be in the state q_a if and only if the stack consists exactly of the string $n\#$. Note that as n is a demonstrative node the stack *cannot* contain a string of the form nv for some finite string v. If it did this would imply that a non-identity element of G had mapped the cone $\lfloor n \rfloor$ to the cone $[nv]$ which is not possible as $\lfloor n \rfloor$ is a demonstrative set.

We have introduced non-determinism into the automaton. If the stack consists of the string $s_i w\#$ then the automaton must choose between the two transitions $(g, s_i, (s_i)g_T)$ and $(g, s_i w\#, (s_i)g_T w\#)$.

When reading a string, the transitions defined above are sufficient to make the automaton well defined on all possible instantaneous descriptions that could arise. That is, there cannot exist a situation in which for a given input letter g and current stack m, a transition cannot be found. If that were the case then the stack m would identify a node which is parted by the element g, which is not possible by our choice of n.

Termination. When we reach the end of the input string, if the automaton is in q_a, then the stack must be $n\#$ by construction, and the input string is equivalent to the identity in our group (by the definition of a demonstrative embedding). If the active state at end of input is q_r then the word is rejected.

The automaton is successful in accepting the word problem by the following reasoning. The stack initially contains the location of the demonstrative node n. Let $w = g_1 g_2 \ldots g_k$ be a string of generators from Λ^*. Each time an input letter g_i is read the stack is updated with the location of the node n after the element $g_1 g_2 \ldots g_i$ has acted on it. Therefore if the automaton reads the whole string and the stack consists only of $n\#$ then the equivalent element $w \in G$ has fixed the node n. By the previous paragraph, if the stack ever consists solely of $n\#$ then there will have existed a transition that would have taken the automaton to the active state q_a. Therefore, if $w = 1_G$ then w will fix n and there will be a sequence of transitions that will take the automaton to the accept state. Thus every string equivalent to the identity in G will be accepted. Suppose $w \neq 1_G$. Then because n is a demonstrative node it cannot be fixed by w. Hence, the automaton cannot be in the active state q_a when it has finished reading w as the stack will not solely consist of the string $n\#$. Thus $\mathcal{A}$ accepts the word problem of G and rejects all other strings. $\square$

Chapter 4

Presentations for Groups in $\mathfrak{F}_{aug}$

In this chapter we change our attention to a class of groups first introduced in [3], which are suggested by the authors as potential counterexamples to Lehnert's conjecture. The groups have their origin in the work by Stefan Witzel and Matthew Zaremsky [33]. In [33] the authors describe a construction that creates "Thompson-like" groups. These groups are defined from *cloning systems* on a family of groups, the details of which we briefly discuss further down the chapter, but direct the interested reader to the recent survey by Matthew Zaremsky in [34] for a more thorough description. Examples of well known groups which arise from such a procedure include the Thompson groups themselves and the braided Thompson groups F_{br} and V_{br}.

In addition to the groups stated above, the technology of cloning systems also brings to light groups previously unstudied. Using the contstruction in [33], Berns-Zieze, Fry, Gillings, Hoganson and Mathews in [3] introduce a group $V_{(G,\theta)}$, where G is a finite group and $\theta : G \to G$ is an endomorphism of G. They go on to prove that $V_{(G,\theta)}$ is $co\mathcal{CF}$ for all possible pairs of G and θ. We use $\mathfrak{V}_{aug}$ to denote the class of all groups isomorphic to a group of the form $V_{(G,\theta)}$, (where 'aug' is short for **augmentation**, a term we will define further in the chapter). That is, a group H is in $\mathfrak{V}_{aug}$ if and only if there exists some finite group G and endomorphism θ such that $H \cong V_{(G,\theta)}$. The work in this chapter and was motivated by the following question raised in [3].

Question 4.1. Does there exist some $H \in \mathfrak{V}_{aug}$ such that H cannot embed into Thompson's group V?

If such a group could be found then it would be a counterexample to Lehnert's conjecture. However, in what follows we do not study groups from $\mathfrak{V}_{aug}$, but instead consider a closely related class of groups, $\mathfrak{F}_{aug}$. Given a group $V_{(G,\theta)}$ we will construct a finitely generated subgroup which we call $F_{(G,\theta)}$. Let $\mathfrak{F}_{aug}$ denote the class of groups where $H \in \mathfrak{F}_{aug}$ if and only if there exists some finite group G and endomorphism θ such that $H \cong$

$F_{(G,\theta)}$. By the properties of *co*$\mathcal{CF}$ groups, as $V_{(G,\theta)}$ is *co*$\mathcal{CF}$ its finitely generated subgroup $F_{(G,\theta)}$ must also be *co*$\mathcal{CF}$. Therefore $\mathfrak{F}_{aug}$ is also a class of potential counterexamples to Lehnert's conjecture. In ordinary Thompson groups theory, one often establishes properties of F before exploring the related and more complicated properties in T or V. In the same manner we focus on the class $\mathfrak{F}_{aug}$ instead of $\mathfrak{V}_{aug}$ as the natural first stop for exploration.

In what follows we construct two group presentations which are isomorphic to $F_{(G,\theta)}$, for all pairs of G and θ. The first presentation we create is an infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$, the second a finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$. One will note that the infinite presentation provides a clearer and more natural way of interpreting the group than the finite presentation. This follows a well known property of group presentations, also seen, for example, in Thompson's group V (see [8]), where a larger, but perhaps more regular, presentation provides more clarity regarding the nature of the group than a finite presentation. Although we have been unable to use either of the presentations $\mathcal{F}^{inf}_{(G,\theta)}$ or $\mathcal{F}^{fin}_{(G,\theta)}$ to find a group $F_{(G,\theta)}$ which does not embed into V, in Chapter 5 we use a variant of $\mathcal{F}^{inf}_{(G,\theta)}$ to show the perhaps surprising result that a subclass of groups from $\mathfrak{F}_{aug}$ *do* embed into V.

Standard definitions and notation

Throughout this chapter, and also Chapter 5, a bold letter $\boldsymbol{n}$ represents the subset $\{1, 2, \ldots, n\}$ of the natural numbers. We also use $\mathbb{N}_i$ to refer to the set of natural numbers greater than or equal to i. If $\gamma = (x_1, x_2, \ldots, x_n)$ is an n-tuple, then we use the notation $\gamma(i) = x_i$ to reference the i^{th} entry of γ.

For the rest of the chapter we fix a finite group G of order N with elements g_i where $i \in \boldsymbol{N}$ and $g_1 = 1_G$. We also fix an endomorphism $\theta : G \to G$. This allows us to refer to the groups $V_{(G,\theta)}$ and $F_{(G,\theta)}$ as existing objects in our discourse. Additionally we define two maps $\delta : \boldsymbol{N} \to \boldsymbol{N}$ and $\Delta : \boldsymbol{N} \times \boldsymbol{N} \to \boldsymbol{N}$ such that for all $i, j \in \boldsymbol{N}$,

$$\begin{aligned} (i)\delta &= s \qquad \text{if } (g_i)\theta = g_s \\ (i,j)\Delta &= t \qquad \text{if } g_i \cdot g_j = g_t. \end{aligned}$$

Also common throughout the chapter is the binary tree with $n+1$ leaves where all but one of the leaves is a left child. We call this a **right vine** binary tree and denote it by $\mathcal{T}_n$.

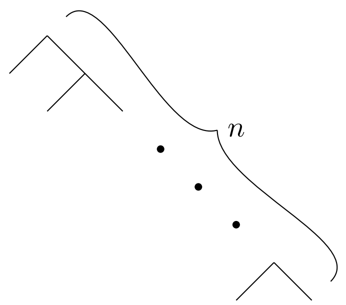


Figure 4.1: The right vine tree T_n

4.1 Statement of results

In this chapter we create two presentations for the group $F_{(G,\theta)}$, where G is a finite group and θ is any endomorphism of G. The first presentation we create is infinite.

Theorem 4.2. *Let G be a finite group of order N, and $\theta : G \to G$ be an endomorphism of G. Then there exists a infinite presentation $\mathcal{F}^{inf}_{(G,\theta)} := \langle X|R\rangle$ for the group $F_{(G,\theta)}$ consisting of three infinite sets of generators and twelve infinite sets of relations.*

In the second half of the chapter we use the infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$ to create a finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$ by using Tietze transformations. Thus we will prove the following theorem.

Theorem 4.3. *Suppose G is a finite group of order N and θ is an endomorphism of G. There exists a finite presentation $\mathcal{F}^{fin}_{(G,\theta)} = \langle X_{fin}|R_{fin}\rangle$ for the group $F_{(G,\theta)}$ consisting of $2(N+1)$ generators and 26 finite sets of relations.*

4.2 The groups $\mathfrak{V}_{aug}$

Witzel and Zaremsky in [33] introduced the notion of a **cloning system** on a family of groups $(G_n)_{n\in\mathbb{N}}$. From a cloning system they described a procedure for creating a group, called the **generalised Thompson group** for the cloning system, or "Thompson-like group". In this section we define the cloning system from which the group $V_{(G,\theta)}$ is derived, and prove that it satisfies the conditions necessary for a cloning system given in [33]. However, we are content to show such a cloning system exists and do not construct $V_{(G,\theta)}$ using the method given [33] as it is detailed and not required to understand the main results of this chapter. (For more information on the Brin-Zappa-Szép product of monoids upon which the formal construction of these groups are based see [33] or [34].) Instead we define $V_{(G,\theta)}$ in the same way given in [3], using binary trees in a similar manner to Thompson's group V.

The literature for cloning systems and Thompson-like groups ([33],[34],[3]) is based on left actions. Our work with the groups $V_{(G,\theta)}$ and $F_{(G,\theta)}$ uses right actions and thus we have translated the standard definitions in the literature from left to right actions. In what follows, the maps $\iota_{m,n}$ and κ_k^n act on the left and ρ_n acts on the right. This is consistent with our translation of the literature where the respective maps above instead act on the opposite sides.

Cloning systems

Let $(H_n)_{i\in\mathbb{N}}$ be a direct system of groups with injective maps $\iota_{m,n} : H_m \to H_n$ for all $m \leq n$. This means that $\iota_{n,n} : H_n \to H_n$ is the identity map for all n, and for all $l \leq m \leq n$ the composition $\iota_{m,n} \circ \iota_{l,m} = \iota_{l,n}$ holds. Note that the way we have defined $\iota_{m,n}$ means it acts on the left, that is, $\iota_{m,n}(h) \in H_n$ for some $h \in H_m$. We fix $(H_n)_{i\in\mathbb{N}}$ and the injective maps $\iota_{m,n} : H_m \to H_n$ for the rest of this section. To define a cloning system we need two more families of maps.

A **representation map** is a homomorphism $\rho_n : H_n \to S_n$ for some $n \in \mathbb{N}$. One can think that a representation map for H_n defines how H_n acts on the set $\boldsymbol{n} := \{1, 2, \ldots, n\}$. For ρ_n to be a representation map we require it to satisfy

$$(\iota_{m,n}(g))\rho_n = \iota_{m,n}((g)\rho_m)$$

for all $m < n$, and all $g \in H_m$. Note that we abuse notation by using $\iota_{m,n}$ for both $H_m \hookrightarrow H_n$ and $S_m \hookrightarrow S_n$. We fix a family of representation maps $(\rho_n)_{n\in\mathbb{N}}$ for our direct system of groups $(H_n)_{i\in\mathbb{N}}$.

A **cloning map** $\kappa_k^n : H_n \to H_{n+1}$ is an injective function from the *set* H_n to the set H_{n+1} where $1 \leq k \leq n$. Note that it need not be a homomorphism of groups. To be cloning maps the functions κ_k^n must satisfy the three cloning axioms which we outline below, plus the additional requirement

$$\kappa_k^n \circ \iota_{m,n} = \iota_{m+1,n+1} \circ \kappa_k^m$$

for all $k \leq m \leq n$. We now fix a family of cloning maps $(\kappa_k^n)_{k\leq n}$ for a direct system of groups $(H_n)_{i\in\mathbb{N}}$.

Definition 4.4 (Cloning system)**.** *The tuple*

$$((H_n)_{n\in\mathbb{N}}, (\iota_{m,n})_{m\leq n}, (\rho_n)_{n\in\mathbb{N}}, (\kappa_k^n)_{k\leq n})$$

is called a ***cloning system*** *if for all $1 \leq k < m \leq n$ and $g, h \in H_n$ it satisfies the three cloning axioms, given by;*

C1*: (Cloning products) $\kappa_k^n(gh) = \kappa_k^n(g) \cdot \kappa_{(k)(g)\rho_n}^n(h)$*

***C2**: (Products of cloning maps)* $\kappa_k^{n+1} \circ \kappa_m^n = \kappa_{m+1}^{n+1} \circ \kappa_k^n$

***C3**: (Compatibility)* $(i)(\kappa_k^n(g))\rho_{n+1} = (i)(\varsigma_k^n((g)\rho_n))$

where $(\varsigma_k^n)_{k \leq n}$ *are the set of cloning maps for* S_n*, such that for all* $\sigma \in S_n$

$$(i)(\varsigma_k^n \sigma) = \begin{cases} (i)\sigma & \text{if } i \leq k \text{ and } (i)\sigma \leq (k)\sigma \\ (i)\sigma + 1 & \text{if } i \leq k \text{ and } (i)\sigma > (k)\sigma \\ (i-1)\sigma & \text{if } i > k \text{ and } (i-1)\sigma < (k)\sigma \\ (i-1)\sigma + 1 & \text{if } i > k \text{ and } (i-1)\sigma \geq (k)\sigma. \end{cases} \tag{4.1}$$

The cloning system for $V_{(G,\theta)}$

We will now introduce a particular class of groups around which we will construct a cloning system. From this cloning system we will define the class of groups $\mathfrak{V}_{aug}$ that is first defined in [3]. (Although the cloning system is mentioned in [3] they do not provide a proof. In what follows below we prove that the system we create satisfies the conditions for a cloning system given in Definition 4.4.) Let G be a finite group and let $\theta : G \to G$ be an arbitrary endomorphism of G. Define H_n to be the permutation wreath product

$$H_n = \bigoplus_{i=1}^{n} G_i \rtimes_\phi S_n,$$

where $G_i \cong G$ for all $1 \leq i \leq n$, S_n is the symmetric group on n points, and $\phi : S_n \to Aut(\bigoplus_{i=1}^n G_i)$ such that $(g_1, \ldots, g_n)\phi(\sigma) = (g_{(1)\sigma}, \ldots, g_{(n)\sigma})$. The map ϕ defines an action of S_n on $\bigoplus_{i=1}^n G_i$ where $\sigma \in S_n$ acts on an element $(g_1, \ldots, g_n)$ by permuting the entries by σ^{-1}. Thus if $((a_1, a_2, \ldots, a_n), \sigma_1)$ and $((b_1, b_2, \ldots, b_n), \sigma_2)$ are two elements of H_n then their product

$$((a_1, a_2, \ldots, a_n), \sigma_1)((b_1, b_2, \ldots, b_n), \sigma_2)$$

is equal to

$$((a_1 b_{(1)\sigma_1}, a_2 b_{(2)\sigma_1}, \ldots, a_n b_{(n)\sigma_1}), \sigma_1 \cdot \sigma_2).$$

For integers $m \leq n$ we define an injective map $\iota_{m,n} : H_m \to H_n$ by

$$\iota_{m,n}((g_1, g_2, \ldots, g_m), \sigma) = ((g_1, g_2, \ldots, g_m, 1_G, 1_G, \ldots, 1_G), \sigma'), \tag{4.2}$$

where

$$(i)\sigma' = \begin{cases} (i)\sigma & \text{if } i \leq m \\ i & \text{otherwise.} \end{cases}$$

The groups $(H_n)_{n\in\mathbb{N}}$ together with the set of maps $(i_{m,n})_{m\leq n}$, form a direct system of groups and we set $H := \varinjlim(H_n)$. We define the representation map $\rho_n : H_n \to S_n$ as

$$(h)\rho_n = \sigma$$

for each $h = ((g_1, g_2, \ldots, g_n), \sigma)$ in H_n.

The cloning maps $\kappa_k^n : H_n \to H_{n+1}$ we define as

$$\kappa_k^n((g_1, \ldots, g_k, g_{k+1}, \ldots, g_n), \sigma) = ((g_1, \ldots, g_k, (g_k)\theta, g_{k+1}, \ldots, g_n), \varsigma_k^n \sigma) \tag{4.3}$$

where $\varsigma_k^n : S_n \to S_{n+1}$ is the cloning map defined in (4.1). We fix the definitions of the representation maps and the cloning maps for the rest of the chapter.

We now go on to prove that the tuple $\mathcal{C}_{(G,\theta)} := ((H_n)_{n\in\mathbb{N}}, (\iota_{m,n})_{m\leq n}, (\rho_n)_{n\in\mathbb{N}}, (\kappa_k^n)_{k\leq n})$ is a cloning system. We first prove the requirements on the maps $(\rho_n)_{n\in\mathbb{N}}$ and $(\kappa_k^n)_{k\leq n}$ with regards to the injective maps $(\iota_{m,n})_{m\leq n}$.

Proposition 4.5. *The representation map $\rho_n : H_n \to S_n$ satisfies the condition*

$$(\iota_{m,n}(h))\rho_n = \iota_{m,n}((h)\rho_m)$$

for all $m < n$ and $h \in H_m$, where we use $\iota_{m,n}$ for both maps $H_m \hookrightarrow H_n$ and $S_m \hookrightarrow S_n$.

Proof. Let $h = ((g_1, \ldots, g_m), \sigma)$ be an element of H_m. Then by (4.2) we have

$$\iota_{m,n}(h) = ((g_1, \ldots, g_m, 1_G, 1_G, \ldots, 1_G), \sigma'),$$

where

$$(i)\sigma' = \begin{cases} (i)\sigma & \text{if } i \leq m \\ i & \text{otherwise.} \end{cases}$$

Therefore $(\iota_{m,n}(h))\rho_n = \sigma'$.

Consider instead $\iota_{m,n}((h)\rho_m)$. By definition, $(h)\rho_m = \sigma$, and the canonical injection of σ from S_m to S_n is σ', therefore $(\iota_{m,n}(h))\rho_n = \iota_{m,n}((h)\rho_m)$ holds. $\square$

Proposition 4.6. *The cloning maps $(\kappa_k^n)_{k\leq n}$ and the injective maps $(\iota_{m,n})_{m\leq n}$ satisfy the rule*

$$\kappa_k^n \circ \iota_{m,n} = \iota_{m+1,n+1} \circ \kappa_k^m$$

for all $k \leq m \leq n$.

Proof. Let $h = ((g_1, \ldots, g_k, g_{k+1}, \ldots, g_m), \sigma)$ be an element of H_m. Then $\iota_{m,n}(h) = ((g_1, \ldots, g_k, g_{k+1}, \ldots, g_m, 1_G, 1_G, \ldots, 1_G), \sigma')$, where $(i)\sigma' = (i)\sigma$ if $1 \leq i \leq m$ and $(i)\sigma =$

i otherwise. Then $\kappa_k^n(\iota_{m,n}(h)) = ((g_1, \ldots, g_k, (g_k)\theta, g_{k+1}, \ldots, g_m, 1_G, 1_G, \ldots, 1_G), \varsigma_k^n \sigma')$.

As $(i)\sigma' = i$ for all $i > m$, $k \leq m$, and $\sigma(k) \leq m$, the permutation $(i)(\varsigma_k^n(\sigma'))$ could be written as

$$(i)(\varsigma_k^n \sigma') = \begin{cases} (i)\sigma & \text{if } i \leq k \text{ and } (i)\sigma \leq (k)\sigma \\ (i)\sigma + 1 & \text{if } i \leq k \text{ and } (i)\sigma > (k)\sigma \\ (i-1)\sigma & \text{if } i > k \text{ and } (i-1)\sigma < (k)\sigma \\ (i-1)\sigma + 1 & \text{if } m+1 \geq i > k \text{ and } (i-1)\sigma \geq (k)\sigma \\ i & \text{if } i > m+1 \end{cases}$$

Suppose instead we act on h with κ_k^m first. Then $\kappa_k^m((g_1, \ldots, g_k, g_{k+1}, \ldots, g_m), \sigma) = ((g_1, \ldots, g_k, (g_k)\theta, g_{k+1}, \ldots, g_m), \varsigma_k^m \sigma)$, where $\varsigma_k^m \sigma$ is the permutation defined in (4.1). Applying the map $\iota_{m+1,n+1}$ to $((g_1, \ldots, g_k, (g_k)\theta, g_{k+1}, \ldots, g_m), \varsigma_k^m \sigma)$ gives

$$((g_1, \ldots, g_k, (g_k)\theta, g_{k+1}, \ldots, g_m, 1_G, \ldots, 1_G), (\varsigma_k^m \sigma)')$$

where $(\varsigma_k^m \sigma)'(i)$ is $\varsigma_k^m \sigma(i)$ if $1 \leq i \leq m+1$ and just i otherwise. Observe that $(\varsigma_k^m \sigma)' = \varsigma_k^n \sigma'$ and therefore $(\kappa_k^n \circ \iota_{m,n})(h) = (\iota_{m+1,n+1} \circ \kappa_k^m)(h)$. $\square$

We can now go on to prove that $\mathcal{C}_{(G,\theta)}$ is a cloning system.

Proposition 4.7. *The tuple* $\mathcal{C}_{(G,\theta)} := ((H_n)_{n\in\mathbb{N}}, (\iota_{m,n})_{m\leq n}, (\rho_n)_{n\in\mathbb{N}}, (\kappa_k^n)_{k\leq n})$ *is a cloning system.*

Proof. To prove that $\mathcal{C}_{(G,\theta)} := ((H_n)_{n\in\mathbb{N}}, (\iota_{m,n})_{m\leq n}, (\rho_n)_{n\in\mathbb{N}}, (\kappa_k^n)_{k\leq n})$ is a cloning system we have to show that $\mathcal{C}_{(G,\theta)}$ satisfies the three cloning conditions given in Definition 4.4. In the proof below we use the result from [33] that the tuple

$$((S_n)_{n\in\mathbb{N}}, (\iota_{m,n})_{m\leq n}, (\rho_n)_{n\in\mathbb{N}}, (\varsigma_k^n)_{k\leq n}),$$

where the cloning maps $(\varsigma_k^n)_{k\leq n}$ are those defined in (4.1), is a cloning system. (We note that the maps $(\iota_{m,n})_{m\leq n}$ and $(\rho_n)_{n\in\mathbb{N}}$ are here defined for the family of groups $(S_n)_{n\in\mathbb{N}}$, where ρ_n is just the identity map on S_n).

C1: Cloning a product

Let $h_1 = ((a_1, a_2, \ldots, a_n), \sigma_1)$ and $h_2 = ((b_1, b_2, \ldots, b_n), \sigma_2)$ for $a_i, b_i \in G$ and $h_1, h_2 \in H_n$. Then $h_1 \cdot h_2 = ((a_1 b_{(1)\sigma_1}, a_2 b_{(2)\sigma_1}, \ldots, a_n b_{(n)\sigma_1}), \sigma_1 \cdot \sigma_2)$. Cloning the product with κ_k^n gives

$$\kappa_k^n(h_1 \cdot h_2) = ((a_1 b_{(1)\sigma_1}, \ldots, a_k b_{(k)\sigma_1}, (a_k b_{(k)\sigma_1})\theta, \ldots, a_n b_{(n)\sigma_1}), \varsigma_k^n(\sigma_1 \cdot \sigma_2)). \tag{4.4}$$

We now consider the product $\kappa^n_k(h_1) \cdot \kappa^n_{(k)\sigma_1}(h_2)$ and show that it is equal to (4.4). We begin by cloning the individual elements.

$$\begin{aligned}
\kappa^n_k(h_1) &= ((a_1, \ldots, a_k, (a_k)\theta, \ldots, a_n), \varsigma^n_k \sigma_1) \\
\kappa^n_{(k)\sigma_1}(h_2) &= ((b_1, \ldots, b_{(k)\sigma_1}, (b_{(k)\sigma_1})\theta, \ldots, b_n), \varsigma^n_{(k)\sigma_1} \sigma_2)
\end{aligned}$$

For ease of notation we introduce the following temporary definitions.

$$\begin{aligned}
\sigma_1' &:= \varsigma^n_k \sigma_1 \\
\sigma_2' &:= \varsigma^n_{(k)\sigma_1} \sigma_2 \\
(a_i')_{1 \le i \le n+1} &:= (a_1, \ldots, a_k, (a_k)\theta, \ldots, a_n) \\
(b_i')_{1 \le i \le n+1} &:= (b_1, \ldots, b_{(k)\sigma_1}, (b_{(k)\sigma_1})\theta, \ldots, b_n)
\end{aligned}$$

where

$$a_i' = \begin{cases} a_i & \text{if } i \le k \\ a_{i-1} & \text{if } i > k+1 \\ (a_k)\theta & \text{if } i = k+1 \end{cases} \tag{4.5}$$

$$b_i' = \begin{cases} b_i & \text{if } i \le (k)\sigma_1 \\ b_{i-1} & \text{if } i > (k)\sigma_1 + 1 \\ (b_{(k)\sigma_1})\theta & \text{if } i = (k)\sigma_1 + 1 \end{cases} \tag{4.6}$$

Therefore the product $\kappa^n_k(h_1) \cdot \kappa^n_{(k)\sigma_1}(h_2)$ becomes $((a_i' b'_{(i)\sigma_1'})_{1 \le i \le n+1}, \sigma_1' \cdot \sigma_2')$. We immediately note that $\sigma_1' \cdot \sigma_2' = \varsigma^n_k \sigma_1 \cdot \varsigma^n_{(k)\sigma_1} \sigma_2 = \varsigma^n_k (\sigma_1 \cdot \sigma_2)$ by Example 2.9 in [33].

To determine each $a_i' b'_{(i)\sigma_1'}$ there are five cases to consider.

If $i \le k$ and $(i)\sigma_1 \le (k)\sigma_1$ then

$$\begin{aligned}
a_i' b'_{(i)\sigma_1'} &= a_i b'_{(i)\sigma_1'} && \text{by (4.5)} \\
&= a_i b'_{(i)\sigma_1} && \text{by (4.1)} \\
&= a_i b_{(i)\sigma_1} && \text{by (4.6)}
\end{aligned}$$

If $i \le k$ and $(i)\sigma_1 > (k)\sigma_1$ then

$$\begin{aligned}
a_i' b'_{(i)\sigma_1'} &= a_i b'_{(i)\sigma_1'} && \text{by (4.5)} \\
&= a_i b'_{(i)\sigma_1+1} && \text{by (4.1)}
\end{aligned}$$

$$= a_i b_{(i)\sigma_1} \qquad \text{by (4.6)}$$

If $i > k+1$ and $(i-1)\sigma_1 < (k)\sigma_1$ then

$$\begin{aligned}
a'_i b'_{(i)\sigma'_1} &= a_{i-1} b'_{(i)\sigma'_1} && \text{by (4.5)}\\
&= a_{i-1} b'_{(i-1)\sigma_1} && \text{by (4.1)}\\
&= a_{i-1} b_{(i-1)\sigma_1} && \text{by (4.6)}
\end{aligned}$$

If $i > k+1$ and $(i-1)\sigma_1 > (k)\sigma_1$ then

$$\begin{aligned}
a'_i b'_{(i)\sigma'_1} &= a_{i-1} b'_{(i)\sigma'_1} && \text{by (4.5)}\\
&= a_{i-1} b'_{(i-1)\sigma_1+1} && \text{by (4.1)}\\
&= a_{i-1} b_{(i-1)\sigma_1} && \text{by (4.6)}
\end{aligned}$$

If $i = k+1$ then

$$\begin{aligned}
a'_i b'_{(i)\sigma'_1} &= (a_k)\theta b'_{(i)\sigma'_1} && \text{by (4.5)}\\
&= (a_k)\theta b'_{(k)\sigma_1+1} && \text{by (4.1)}\\
&= (a_k)(b_{(k)\sigma_1})\theta && \text{by (4.6)}
\end{aligned}$$

Therefore $(a'_i b'_{(i)\sigma'_1})_{1\le i\le n+1} = (a_1 b_{(1)\sigma_1}, \ldots, a_k b_{(k)\sigma_1}, (a_k b_{(k)\sigma_1})\theta, \ldots, a_n b_{(n)\sigma_1})$. Hence $\kappa^n_k(h_1) \cdot \kappa^n_{(k)\sigma_1}(h_2) = \kappa^n_k(h_1 \cdot h_2)$ and **C1** is verified.

C2: A product of cloning maps

Let $h = ((g_1, \ldots, g_n), \sigma)$ be an element of H_n and let k and m be integers such that $1 \le k < m \le n$. First consider the product of cloning maps $\kappa^{n+1}_k \circ \kappa^n_m$ acting on h. The maps act on the left so we first act on h with κ^n_m to give $\kappa^n_m(h) = ((g_1, \ldots, g_m, (g_m)\theta, \ldots, g_n), \varsigma^n_m\sigma)$. Recall that $m > k$, thus by applying κ^{n+1}_k to $\kappa^n_m(h)$ we get the following,

$$\kappa^{n+1}_k(\kappa^n_m(h)) = \begin{cases} ((g_1, \ldots, g_m, (g_m)\theta, \theta^2(g_m), \ldots, g_n), \varsigma^{n+1}_{m+1}(\varsigma^n_m\sigma)) & \text{if } m = k+1 \\ ((g_1, \ldots, g_k, (g_k)\theta, \ldots, g_m, (g_m)\theta, \ldots, g_n), \varsigma^{n+1}_k(\varsigma^n_m\sigma)) & \text{otherwise.} \end{cases}$$

Suppose instead we begin by acting on h with κ^n_k to produce the element $\kappa^n_k(h) = ((g_1, \ldots, g_k, (g_k)\theta, \ldots, g_n), \varsigma^n_k\sigma)$. Therefore, for all $i > k+1$, the i^{th} component of $(g_1, \ldots, g_k, (g_k)\theta, \ldots, g_n)$ will be equal to the $(i-1)^{th}$ component of $(g_1, \ldots, g_k, \ldots, g_n)$

from h. Therefore applying κ_{m+1}^{n+1} to $\kappa_k^n(h)$ produces the element;

$$\kappa_{m+1}^{n+1}(\kappa_k^n(h)) = \begin{cases} ((g_1, \ldots, g_k, (g_k)\theta, \theta^2(g_k), \ldots, g_n), \varsigma_{k+1}^{n+1}(\varsigma_k^n \sigma)) & \text{if } m = k+1 \\ ((g_1, \ldots, g_k, (g_k)\theta, \ldots, g_m, (g_m)\theta, \ldots, g_n), \varsigma_{m+1}^{n+1}(\varsigma_k^n \sigma)) & \text{otherwise.} \end{cases}$$

By Example 2.9 in [33], the cloning maps $(\varsigma_k^n)_{k \leq n}$ all satisfy $\varsigma_k^{n+1} \circ \varsigma_m^n = \varsigma_{m+1}^{n+1} \circ \varsigma_k^n$ for $k < m$. Thus $\kappa_k^{n+1} \circ \kappa_m^n = \kappa_{m+1}^{n+1} \circ \kappa_k^n$ and **C2** is verified.

C3: Compatibility

Let $h = ((g_1, \ldots, g_n), \sigma)$. Then $(h)\rho_n = \sigma$ by definition. Applying the cloning map κ_k^n to h gives $((g_1, \ldots, g_k, (g_k)\theta, \ldots, g_n), \varsigma_k^n \sigma)$. Then $(\kappa_k^n(h))\rho_{n+1} = \varsigma_k^n \sigma = \varsigma_k^n((h)\rho_n)$ and **C3** is verified. $\square$

4.2.1 Definition of $V_{(G,\theta)}$

Having shown that $\mathcal{C}_{(G,\theta)}$ is a cloning system, we now go on to define the Thompson-like group derived from this cloning system. Recall that G is a finite group, θ is an endomorphism of G and the group H_n is the symmetric wreath product of G and S_n for all $n \in \mathbb{N}$. For each G and θ we denote by $V_{(G,\theta)}$ the group defined by the cloning system $\mathcal{C}_{(G,\theta)}$. As mentioned in the introduction to the section, we do not use the construction of $V_{(G,\theta)}$ described in [33] and instead use the definition given in [3].

We begin by defining the elements of $V_{(G,\theta)}$ then go on to introduce multiplication between elements and finally prove that $V_{(G,\theta)}$ satisfies the conditions of a group. To define the elements of $V_{(G,\theta)}$ we begin by defining what we call an **augmented tree pair**.

Definition 4.8 (Augmented tree pair)**.** *An **augmented tree pair** with n leaves is a tuple $A = (\mathcal{D}, h, \mathcal{R})$ where $\mathcal{D}$ and $\mathcal{R}$ are n-leaved (rooted) binary trees and $h = (\gamma, \sigma)$ is an element of H_n such that $\gamma \in G^n$ and $\sigma \in S_n$. We call $\mathcal{D}$ and $\mathcal{R}$ the **domain** and **range** trees of A respectively.*

We can draw augmented tree pairs in much the same way that we draw binary tree pair representatives for V. Let $A = (\mathcal{D}, (\gamma, \sigma), \mathcal{R})$ be an augmented tree pair with n leaves. We represent A by first drawing the domain tree on the left and the range tree on the right. We then label the leaves of $\mathcal{D}$ from left to right with the numbers 1 up to n. The leaves of $\mathcal{R}$ are correspondingly labelled such that the i^{th} leaf is labelled with the number $(i)\sigma^{-1}$. The element γ is represented by labelling the i^{th} leaf of $\mathcal{D}$ and the $(i)\sigma^{th}$ leaf of $\mathcal{R}$ with the element $(i)\gamma$ from G. As such, we call γ the **decoration** of the augmented tree pair, as the elements of γ decorate the leaves of the trees. Below we give an example of an augmented tree pair A with three leaves, where $\sigma = (1\ 2\ 3)$ and $\gamma = (g_1, g_2, g_3)$.

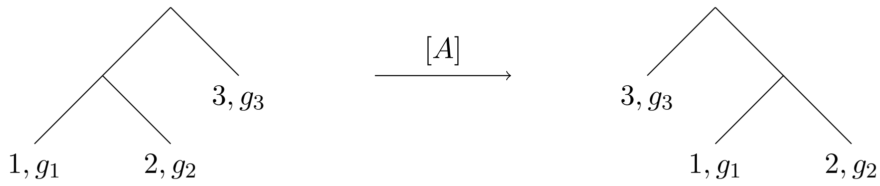


Figure 4.2

Note the difference between our augmented tree pairs and those used by the authors of [3], in which they only place the decoration on *either* the domain or the range tree, and not both. The difference will mean a slight change in the way we multiply elements using augmented tree pairs. It is also common in the literature ([33],[34],[3]) to put the domain tree on the right and the range tree on the left, as these articles all use left actions. In our work we will use the opposite convention, our domain trees will always be drawn on the left, and the range trees on the right, as we consider our elements to act on the right which often occurs in literature around Thompson's groups.

Leaf maps

Let $A = (\mathcal{D}, (\gamma, \sigma), \mathcal{R})$ be an n-leaved augmented tree pair. It is sometimes more convenient to represent A using a collection of maps, similar to prefix replacements in Thompson's group V. Let $L_1 := \{d_1, d_2, \ldots, d_n\}$ be the antichain represented by $\mathcal{D}$ and $L_2 := \{r_1, r_2, \ldots, r_n\}$ be the antichain represented by $\mathcal{R}$. The **leaf map** for the i^{th} leaf of A is the map $(d_i, \gamma(i)) \mapsto (r_{(i)\sigma}, \gamma(i))$. An augmented tree pair can be completely described by the complete set of leaf maps for its leaves.

Splitting operation

Let $\mathfrak{A}$ be the set of all augmented tree pairs. An element of $V_{(G,\theta)}$ is an equivalence class of an equivalence relation we will place on $\mathfrak{A}$. To define the equivalence relation we need the following operation.

Definition 4.9 (Splitting operation for augmented tree pairs)**.** *Let $A = (\mathcal{D}, h, \mathcal{R})$ be an augmented tree pair. Applying the splitting operation to the k^{th} leaf of A produces the augmented tree pair $A' = (\mathcal{D}', h', \mathcal{R}')$ where $\mathcal{D}'$ is the binary tree created by adding a caret to the k^{th} leaf of $\mathcal{D}$, $\mathcal{R}'$ is the binary tree created by adding a caret to the $(k)(\sigma^{-1})^{th}$ leaf of $\mathcal{R}$, and $h' = \kappa^n_k(h)$.*

The splitting operation becomes clearer by considering an example. Let $A = (\mathcal{D}, (\gamma, \sigma), \mathcal{R})$ be the augmented tree pair given in Figure 4.2. Applying the splitting operation to the second leaf of A gives augmented tree pair A' below.

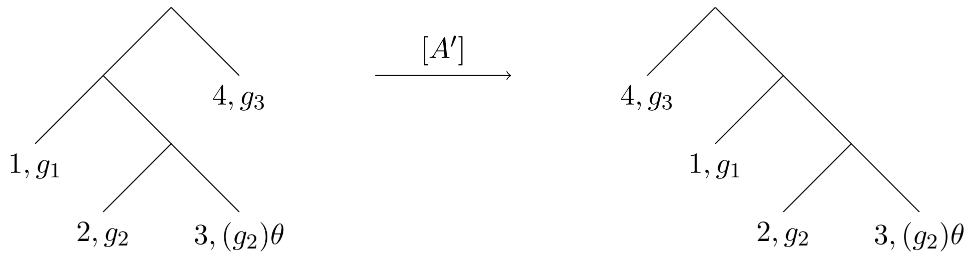


Figure 4.3: An example of the splitting operation. The tree pair above is created from $[A]$ in Figure 4.2 by splitting the second leaf.

The augmented tree pair A' is constructed by adding a caret to the second leaf of the domain tree of A and to the third leaf of the range tree. The permutation $\sigma = (1\ 2\ 3)$ has been replaced with $\sigma' = \varsigma_k^n(\sigma) = (1\ 2\ 3\ 4)$ and the decoration has been replaced with $\gamma' = (g_1, g_2, (g_2)\theta, g_3)$. Notice that the bijection on the leaves of A' defined by σ' matches the bijection on the leaves of A for all shared leaves between the two tree pairs. This is due to the way we defined the cloning map, and this property will hold any time we use the splitting operation.

We define an equivalence relation, " $\sim$ ", on the set of all augmented tree pairs $\mathfrak{A}$, to be the symmetric transitive closure of the splitting operation. The reflexive condition is satisfied by the empty splitting which splits no leaf and returns the original tree pair. The inverse operation we call **shrinking**, in which a decorated caret is removed from both trees of an augmented tree pair. We cannot simply remove any decorated caret however, it must be one which we could recover using the splitting operation. We call this type of caret an **exposed caret** and drop the moniker *decorated*. Suppose $B = (\mathcal{D}, (\gamma, \sigma), \mathcal{R})$ is an augmented tree pair such that for some i there exists a caret of $\mathcal{D}$ consisting of the i^{th} and $(i+1)^{th}$ leaves and a caret of $\mathcal{R}$ consisting of the $(i)\sigma^{th}$ and $(i+1)\sigma^{th}$ leaves with the property that $(i+1)\gamma = ((i)\gamma)\theta$. We would then say that B has an exposed caret consisting of the i^{th} and $(i+1)^{th}$ leaves of B. Notice that if we shrink this caret and produce the tree pair B' then we can recover B by splitting the i^{th} leaf of B'. Consider the previous example in Figure 4.3, one can recover the augmented tree pair A in Figure 4.2 by shrinking the caret of A' consisting of the second and third leaves.

For each augmented tree pair $A = (\mathcal{D}, h, \mathcal{R})$ we denote its equivalence class by $[A] = [\mathcal{D}, h, \mathcal{R}]$. We denote the set of all equivalence classes by $[\mathfrak{A}] := \mathfrak{A}/\sim$. For each equivalence class in $\mathfrak{A}$ under $\sim$ we will now prove that there exists a canonical representative which which has no exposed carets, which we call an **irreducible augmented tree pair**.

Proposition 4.10 (Existence)**.** *In each equivalence class $[A]$ in $[\mathfrak{A}]$ there exists an irreducible augmented tree pair.*

Proof. Let $A = (\mathcal{D}, \gamma, \mathcal{R})$ be an n-leaved augmented tree pair in an equivalence class $[A]$ of $[\mathfrak{A}]$. If A has no exposed carets then we are done. Suppose A has an exposed caret consisting of the (i)th and $(i+1)$th leaves of $\mathcal{D}$ and $\mathcal{R}$. We thus apply the shrinking operation to this caret and produce a new augmented tree pair A_1 which is also in $[A]$ and has $(n-1)$-leaves. We then repeat the process with A_1 and all subsequent augmented tree pairs that are created by shrinking exposed carets. Eventually we must reach the augmented tree pair $A_n \in [A]$ for some n that has no exposed carets. The process must terminate as the number of leaves of the augmented tree pair decreases by one after every iteration of the process. Note that it is possible to end with the augmented tree pair whose domain or range trees are nothing but the single leaved binary tree also known as the *root*, at this point the augmented tree pair contains no carets and thus the process must terminate. □

We will now prove that this irreducible tree pair created from an arbitrary element in the equivalence class is unique.

Proposition 4.11 (Uniqueness)**.** *Suppose A is an irreducible augmented tree pair in an equivalence class $[A]$ of $[\mathfrak{A}]$. If $B \in [A]$ such that B is irreducible then $A = B$.*

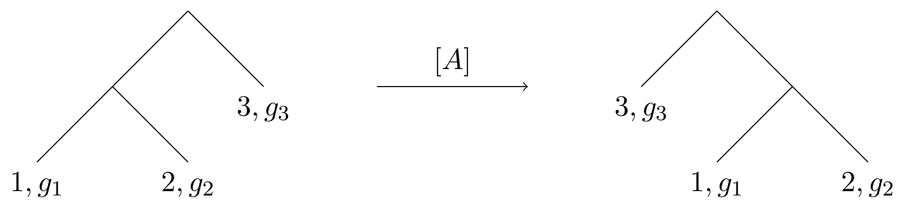
Proof. Suppose A and B are two irreducible augmented tree pairs in an equivalence class $[A]$ of $[\mathfrak{A}]$ such that $A \neq B$. Let the number of leaves in A be l_A and the number of leaves in B to be l_B and without any loss of generality suppose that $l_A \geq l_B$. Let $m > 0$ be the minimum number of splitting and shrinking operations required to take A to B and suppose $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_m$ is one such sequence of operations where each $\mathcal{O}_i$ is either the splitting of a single leaf or the shrinking of an exposed caret. Let $A = A_0, A_1, A_2, \ldots, A_m = B$ be the sequence of augmented tree pairs such that each A_i is created by applying $\mathcal{O}_i$ to A_{i-1}. As A is irreducible $\mathcal{O}_1$ must be a splitting operation and thus A_1 must have $l_A + 1$ leaves. However, as $l_B \leq l_A$ there must exist some minimal natural number k such that A_k has fewer leaves that A_{k-1}, and thus $\mathcal{O}_k$ must be a shrinking operation. However, the only carets that are exposed in A_k are those which have been created by the splitting operations $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_{k-1}$ that preceded it. Suppose the exposed caret deleted by $\mathcal{O}_k$ was first created by $\mathcal{O}_j$ for some $j < k$. Call this caret c. As $\mathcal{O}_k$ is the first instance of a shrinking operation, there can be no operation $\mathcal{O}_l$ prior to $\mathcal{O}_k$ that splits $\mathcal{O}_j$. If there were, then we would require the caret added by $\mathcal{O}_l$ to be removed by another operation before the action of $\mathcal{O}_k$, a contradiction on the minimality of k. Therefore the sequence of operations sequence of operations $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_{j-1}, \mathcal{O}_{j+1}, \ldots, \mathcal{O}_{k-1}$ is well defined, but note that it will also take A to A_k, a contradiction to the original claim as this sequence is shorter than $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_m$. Thus $A = B$. □

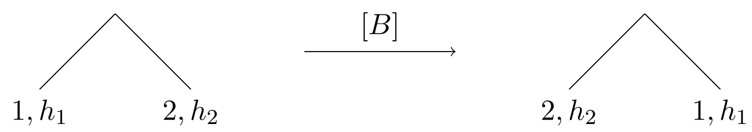
Multiplication on $[\mathfrak{A}]$

We now define a binary operation "$\bullet$" : $[\mathfrak{A}] \times [\mathfrak{A}] \to [\mathfrak{A}]$. Suppose $[A]$ and $[B]$ are in $[\mathfrak{A}]$ such that $A = (\mathcal{D}_A, h_A, \mathcal{R}_A)$ and $B = (\mathcal{D}_B, h_B, \mathcal{R}_B)$ are (not necessarily irreducible) augmented tree pairs. The product $[A] \bullet [B]$ is defined by the following process. First find augmented tree pairs $U = (\mathcal{D}_U, h_U, \mathcal{R}_U)$ and $V = (\mathcal{D}_V, h_V, \mathcal{R}_V)$ such that $[A] = [U]$, $[B] = [V]$ and $\mathcal{R}_U = \mathcal{D}_V$. Thus U and V are chosen so that the range tree of U is the same as the domain tree of V. This can always be done, for example let $\mathcal{R}_U$ be equal to the union of the trees $\mathcal{R}_A$ and $\mathcal{D}_B$. By performing the operations on A that create $\mathcal{R}_U$ from $\mathcal{R}_A$ will in turn create U, and by performing the operations on B that create $\mathcal{R}_U$ from $\mathcal{D}_B$ will in turn create V. The product $[A] \bullet [B]$ is equal to $[\mathcal{D}_U, h_U \cdot h_V, \mathcal{R}_V]$.

Before we prove that the binary product is well defined we provide an example.

Example 4.12. Let A and B be the two augmented tree pairs drawn below.





To create the product $[A] \cdot [B]$ we must find augmented tree pairs $U = (\mathcal{D}_U, h_U, \mathcal{R}_U)$ and $V = (\mathcal{D}_V, h_V, \mathcal{R}_V)$ that are equivalent to A and B such that $\mathcal{R}_U = \mathcal{D}_V$. One such instance is to set $U = A$ and V to be the augmented tree pair that is produced by splitting the second leaf of B. Figure 4.4 gives the augmented tree pairs U and V explicitly.

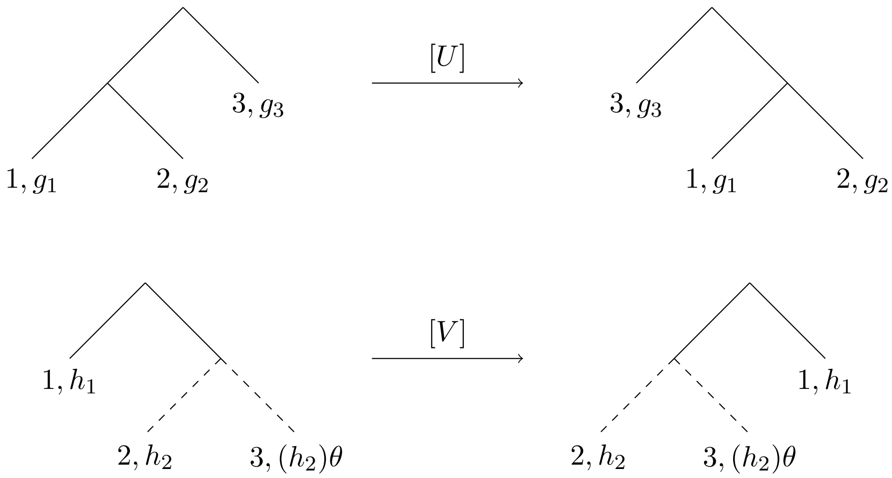


Figure 4.4

Following the definition of multiplication the product $[A] \bullet [B]$ is thus represented by the following augmented tree pair.

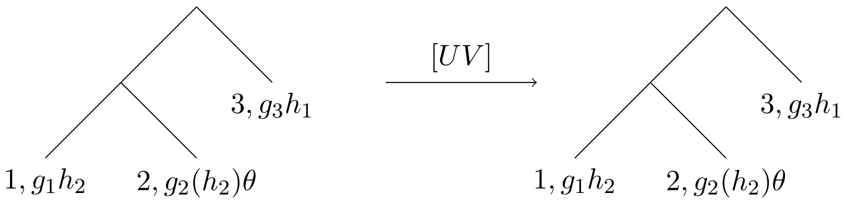


Figure 4.5

It is worth noting at this stage a divergence in notation from the mainstream of Thompson group literature, particularly in the work of Cannon, Floyd and Parry ([12]) to which we refer to many times. In [12] it is the convention to write multiplication of (non-augmented) tree pairs from right to left, and thus the product VU of two binary tree pairs $U = (\mathcal{D}_U, \mathcal{R}_U)$ and $V = (\mathcal{D}_V, \mathcal{R}_V)$ would correspond to our tree pair multiplication $(\mathcal{D}_U, \mathcal{R}_U) \cdot (\mathcal{D}_V, \mathcal{R}_V)$ which is done from left to right.

Proposition 4.13. *The binary operation "$\bullet$" on $[\mathfrak{A}]$ is well defined.*

Proof. Let $[A]$ and $[B]$ be equivalence classes from $[\mathfrak{A}]$, where A and B are tree pairs from $\mathfrak{A}$. There are two ways in which the binary operation $[A] \cdot [B]$, as described above, may not be well defined. It firstly may depend on our choice of the tree pairs U and V, secondly it may depend on the initial choice of representatives of $[A]$ and $[B]$.

We first check our choice of U and V. Let $U = (\mathcal{D}_U, h_U, \mathcal{R}_U)$, $V = (\mathcal{D}_V, h_V, \mathcal{R}_V)$, $U' = (\mathcal{D}_{U'}, h_{U'}, \mathcal{R}_{U'})$ and $V' = (\mathcal{D}_{V'}, h_{V'}, \mathcal{R}_{V'})$ where $h_j = (\gamma_j, \sigma_j)$ for all $j \in \{U, V, U', V'\}$. Suppose U, V, U', V' are such that $[U] = [U']$, $[V] = [V']$, $\mathcal{R}_U = \mathcal{D}_V$ and $\mathcal{R}_{U'} = \mathcal{D}_{V'}$. Let

$UV = (\mathcal{D}_U, h_U \cdot h_V, \mathcal{R}_V)$ and $U'V' = (\mathcal{D}'_U, h'_U \cdot h'_V, \mathcal{R}'_V)$. To show that the binary product is well defined we must show that $[UV] = [U'V']$. Suppose that U and U' differ only by one application of the splitting operation and, without loss of generality, let us assume that we have split the i^{th} leaf of U to produce U'. Then $U' = (\mathcal{D}_{U'}, \kappa_i^n(h_U), \mathcal{R}_{U'})$. Notice that $\mathcal{R}_{U'}$ is produced by adding a caret to the $(i)\sigma_U^{th}$ leaf of $\mathcal{R}_U$. Therefore, as $\mathcal{R}_U = \mathcal{D}_V$ and $\mathcal{R}_{U'} = \mathcal{D}_{V'}$, if we apply the splitting operation to the $(i)\sigma_U^{th}$ leaf of V the tree $\mathcal{D}_V$ will become $\mathcal{D}_{V'}$ and thus V we will become V'. Therefore $V' = (\mathcal{D}_{V'}, \kappa^n_{(i)\sigma_U}(h_V), \mathcal{R}_{V'})$ where $\mathcal{R}_{V'}$ is produced by adding a caret to the $(i)(\sigma_U \cdot \sigma_V)^{th}$ leaf of $\mathcal{R}_V$. Hence $U'V' = (\mathcal{D}_{U'}, \kappa_i^n(h_U) \cdot \kappa^n_{(i)\sigma_U}(h_V), \mathcal{R}_{V'})$.

Consider now the application of the splitting operation to the i^{th} leaf of UV. As $(h_U \cdot h_V)\rho_n = \sigma_U \cdot \sigma_V$ the splitting operation will produce the augmented binary tree $(\mathcal{D}_{U'}, \kappa_i^n(h_U \cdot h_V), \mathcal{R}_{V'})$. By using the first cloning condition **C1** from Definition 4.4 the cloned product $\kappa_i^n(h_U \cdot h_V)$ becomes

$$\kappa_i^n(h_U \cdot h_V) = \kappa_i^n(h_U) \cdot \kappa^n_{(i)\sigma_U}(h_V). \tag{4.7}$$

Therefore $U'V' = (\mathcal{D}_{U'}, \kappa_i^n(h_U \cdot h_V), \mathcal{R}_{V'})$ and thus $[UV] = [U'V']$. Therefore, as the equivalence relation is the symmetric transitive closure of the splitting operation, we can extend our result to show that the binary operation is well defined for *any* choice of U' and V' such that $[U] = [U']$ and $[V] = [V']$.

Finally we check our choice of representative A and B. Suppose A, A' and B, B' are in $\mathfrak{A}$ such that $[A] = [A']$ and $[B] = [B']$. Suppose $U = (\mathcal{D}_U, h_U, \mathcal{R}_U)$ and $V = (\mathcal{D}_V, h_V, \mathcal{R}_V)$ are in $\mathfrak{A}$ such that $[U] = [A] = [A']$, $[V] = [B] = [B']$ and $\mathcal{R}_U = \mathcal{D}_V$. Then $[A] \bullet [B] = [UV] = [A'] \bullet [B']$ and multiplication is well defined in our choice of representatives. □

Proposition 4.14. *$V_{(G,\theta)} = ([\mathfrak{A}], \bullet)$ is a group.*

Proof.

Associativity: Suppose $A = (\mathcal{D}_A, h_A, \mathcal{R}_A)$, $B = (\mathcal{D}_B, h_B, \mathcal{R}_B)$ and $C = (\mathcal{D}_C, h_C, \mathcal{R}_C)$ are augmented tree pairs representing three elements of $[\mathfrak{A}]$. Let $A' = (\mathcal{D}_{A'}, h_{A'}, \mathcal{R}_{A'})$, $B' = (\mathcal{D}_{B'}, h_{B'}, \mathcal{R}_{B'})$ and $C' = (\mathcal{D}_{C'}, h_{C'}, \mathcal{R}_{C'})$ be equivalent augmented tree pairs for A, B and C respectively, such that $\mathcal{R}_{A'} = \mathcal{D}_{B'}$ and $\mathcal{R}_{B'} = \mathcal{D}_{C'}$. These augmented tree pairs can always be found by using the splitting operation on all three elements until the desired pairs are found. This gives us the following two ways to multiply these three elements;

$$[A] \bullet ([B] \bullet [C]) = (\mathcal{D}_{A'}, h_{A'} \cdot (h_{B'} \cdot h_{C'}), \mathcal{R}_{C'})$$
$$([A] \bullet [B]) \bullet [C] = (\mathcal{D}_{A'}, (h_{A'} \cdot h_{B'}) \cdot h_{C'}, \mathcal{R}_{C'})$$

As H_n is a group it is associative thus $h_{A'} \cdot (h_{B'} \cdot h_{C'}) = (h_{A'} \cdot h_{B'}) \cdot h_{C'}$ and hence $[A] \bullet ([B] \bullet [C]) = ([A] \bullet [B]) \bullet [C]$.

Identity: Any augmented tree pair of the form $(\mathcal{D}, (1_{G_n}, 1_{S_n}), \mathcal{D})$ represents the identity in $V_{(G,\theta)}$.

Inverse: Let $[A] = [\mathcal{D}, h, \mathcal{R}]$ be an element of $[\mathfrak{A}]$. The reader can see that $[\mathcal{D}, h, \mathcal{R}] \bullet [\mathcal{R}, h^{-1}, \mathcal{D}] = [\mathcal{D}, (1_{G_n}, 1_{S_n}] = [\mathcal{R}, h^{-1}, \mathcal{D}] \bullet [\mathcal{D}, h, \mathcal{R}]$. Thus $[A]^{-1} = [\mathcal{R}, h^{-1}, \mathcal{D}]$ is the inverse of $[A]$ in $V_{(G,\theta)}$.

□

One can observe that there exists a copy of V in $V_{(G,\theta)}$, namely as the subgroup of all the elements of the form $[\mathcal{D}, 1_{H_n}, \mathcal{R}]$. Additionally there exist a surjective map $\vartheta : V_{(G,\theta)} \to V$ by $[\mathcal{D}, h, \mathcal{R}] \mapsto [\mathcal{D}, \mathcal{R}]$ that splits via $[\mathcal{D}, \mathcal{R}] \mapsto [\mathcal{D}, 1_{H_n}, \mathcal{R}]$. Thus $V_{(G,\theta)} = ker(\vartheta) \rtimes V$ [Observation 3.1, [33]].

The groups $\mathfrak{F}_{aug}$

As mentioned in the introduction the groups in the class $\mathfrak{V}_{aug}$ are of interest to us because of a result given in [3].

Theorem 4.15 (Theorem 4.3 of [3])**.** *The groups in $\mathfrak{V}_{aug}$ are co$\mathcal{CF}$.*

Thus $\mathfrak{V}_{aug}$ could provide counterexamples to Lehnert's conjecture that V is a universal *co*$\mathcal{CF}$ group. To provide a counterexample one would have to prove that there exists a group $V_{(G,\theta)}$ that does not embed into V. Since the class of *co*$\mathcal{CF}$ groups is closed under taking finitely generated subgroups, if one found a f.g. subgroup of $V_{(G,\theta)}$ that did not embed into V, then this would also provide a counterexample to the conjecture. In what follows for the rest of the chapter we will be investigating one particular class of these subgroups which we call $\mathfrak{F}_{aug}$. Given a finite group G and an endomorphism $\theta : G \to G$, the group $F_{(G,\theta)}$ in $\mathfrak{F}_{aug}$ is a subgroup of $V_{(G,\theta)}$ in which every augmented tree pair representative $(\mathcal{D}, h, \mathcal{R})$ in $F_{(G,\theta)}$ satisfies $(h)\rho_n = 1_{S_n}$. In other words, $F_{(G,\theta)}$ contains all the equivalence classes of augmented tree pairs that have the trivial bijection between the leaves of the domain and range trees.

One can see that the relationship between $F_{(G,\theta)}$ and $V_{(G,\theta)}$ is similar to that between the Thompson groups F and V. Much like in F, when we draw elements of a group $F_{(G,\theta)}$ we will forgo the numerical labelling of the leaves which represents the bijection between the leaves of the two trees. As the bijection is always trivial we remove the explicit reminder, making the tree pair diagrams less cluttered. Also, when writing an element $f \in F_{(G,\theta)}$ in the form $[\mathcal{D}, (\gamma, 1_{S_n}), \mathcal{R}]$ we reduce the notation and simply give it as $f = [\mathcal{D}, \gamma, \mathcal{R}]$.

4.3 An infinite presentation for $F_{(G,\theta)}$

In this section we will construct an infinite presentation of the group $F_{(G,\theta)}$ and produce normal forms for elements of any group in $\mathfrak{F}_{aug}$ using the appropriate infinite sets of generators. The construction of the presentation will follow a similar method that was used in the construction of the presentation of Thompson's group F in [12] and also the creation of normal forms of Thompson's group T in [11]. We will begin by introducing an infinite set $\Lambda_{(G,\theta)}$ of generators for $F_{(G,\theta)}$, which we will refer to as Λ if G and θ are understood, and go on to show that exists a unique normal form for each element in $F_{(G,\theta)}$, which is similar in its construction to the normal form introduced in [11] for Thompson's group T. Once this ground work has been laid, we will introduce a presentation $\mathcal{F}^{inf}_{(G,\theta)}$ and go on to prove that $\mathcal{F}^{inf}_{(G,\theta)} \cong F_{(G,\theta)}$.

4.3.1 Normal Form

To create a normal form for elements of $F_{(G,\theta)}$ we begin with a generating set Λ. We define Λ as the union of three infinite sets, $\mathfrak{X}$, $\mathfrak{L}$ and $\mathfrak{R}$ which are given below.

The first set, $\mathfrak{X} := \{X_n : n \in \mathbb{N}_0\}$, consists of all the elements X_n defined by the augmented tree pair in Figure 4.6 below.

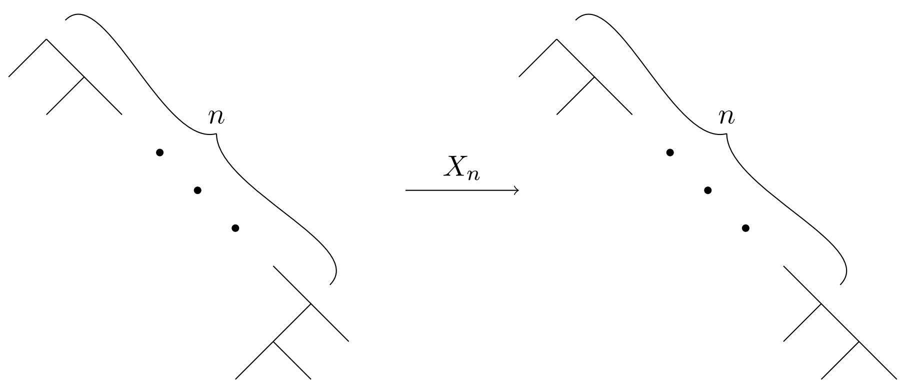


Figure 4.6: The element X_n

Notice that the generators X_n above can be identified with the inverse generators of R. Thompson's group F as given in [12]. However, as *augmented* tree pairs recall that there is also a decoration associated with the tree pair. In what follows, if a leaf of a tree pair is not labelled with an element of G then we assume it is decorated with the idenity element. The reason we work with the inverse generators of those given in [12] is to counter the fact that we write our products from left to right as opposed to those in [12] which are right to left. Using the inverse generators will mean that the relations for F which will appear in our presentation of $F_{(G,\theta)}$ will have the same form as those from [12].

The second set $\mathfrak{L} := \{\Sigma_{g,n} : g \in G, n \in \mathbb{N}\}$ consists of the elements $\Sigma_{g,n} = [T_n, \gamma_\Sigma, T_n]$, where T_n is the $(n+1)$ leaved binary tree consisting solely of the right vine, and $\gamma_\Sigma \in G^n$ is defined as;

$$\gamma_\Sigma(k) = \begin{cases} g & \text{if } k = n \\ 1_G & \text{otherwise.} \end{cases}$$

Figure 4.7 represents the augmented tree pair $(T_n, \gamma_\Sigma, T_n)$.

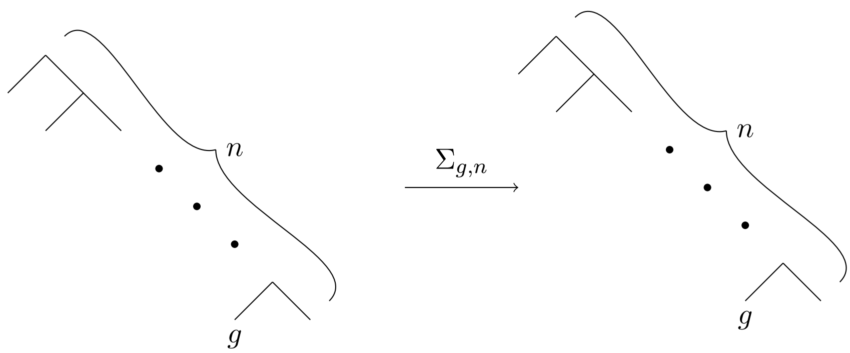


Figure 4.7: An augmented tree pair representative of $\Sigma_{g,n}$.

The final set we define is $\mathfrak{R} := \{\Gamma_{g,n} : g \in G, n \in \mathbb{N}\}$, which contains elements of the form $\Gamma_{g,n} = [T_n, \gamma_\Gamma, T_n]$ for all $n \geq 0$, where

$$\gamma_\Gamma(k) = \begin{cases} g & \text{if } k = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

Figure 4.8 gives the augmented tree pair $(T_n, \gamma_\Gamma, T_n)$.

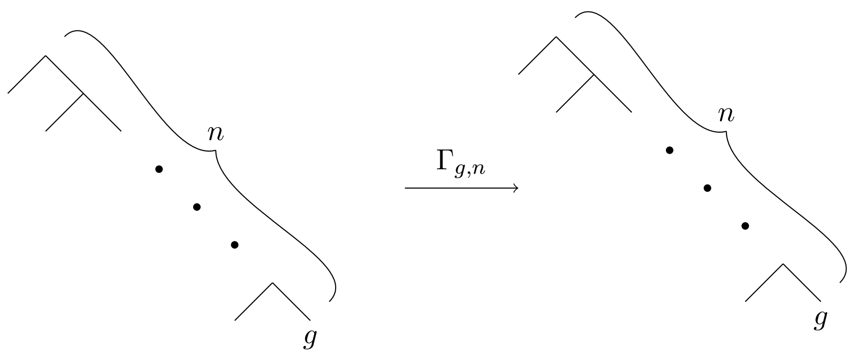


Figure 4.8: An augmented tree pair representative of $\Gamma_{g,n}$.

We will now prove some important results regarding elements from $\mathfrak{L}$ and $\mathfrak{R}$ that will be used extensively in the rest of the chapter. The first is a commutativity result involving elements from $\mathfrak{L}$.

Proposition 4.16. *For all $g, h \in G$ if $i \neq j$ then $\Sigma_{g,i}\Sigma_{h,j} = \Sigma_{h,j}\Sigma_{g,i}$.*

Proof. Suppose i, j are positive integers such that $i < j$. By definition the elements $\Sigma_{g,i}$ and $\Sigma_{h,j}$ are represented by the augmented tree pairs (T_i, γ_1, T_i) and (T_j, γ_2, T_j) where

$$\gamma_1(k) = \begin{cases} g & \text{if } k = i \\ 1_G & \text{otherwise,} \end{cases} \qquad \gamma_2(k) = \begin{cases} h & \text{if } k = j \\ 1_G & \text{otherwise.} \end{cases}$$

To find the product $\Sigma_{g,i}\Sigma_{h,j}$ we first create an equivalent augmented tree pair for $\Sigma_{g,i}$ by splitting the final leaf of (T_i, γ_1, T_i) a total of $j - i$ times. As the decoration on the final leaf of (T_i, γ_1, T_i) is $\gamma_1(i+1) = 1_G$ the process of splitting produces the augmented tree pair (T_j, γ_1', T_j) where the element $\gamma_1' \in G^{j+1}$ is defined similarly to $\gamma_1 \in G^{i+1}$;

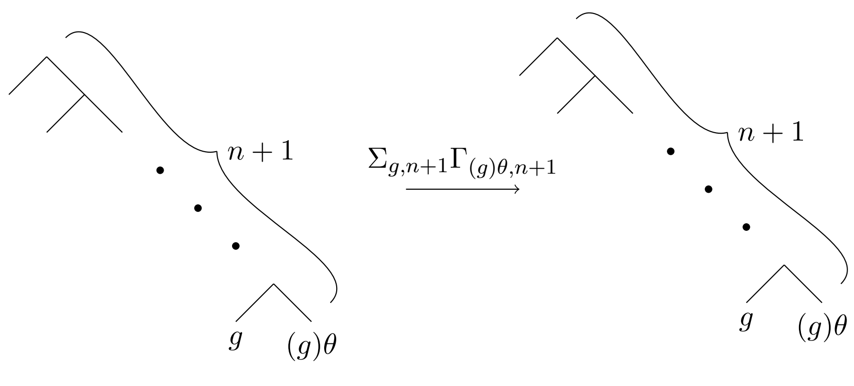
$$\gamma_1'(k) = \begin{cases} g & \text{if } k = i \\ 1_G & \text{otherwise.} \end{cases}$$

The product $\Sigma_{g,i}\Sigma_{h,j}$ is then given by the augmented tree pair $(T_j, \gamma_1', T_j)(T_j, \gamma_2, T_j) = (T_j, \gamma_1' \cdot \gamma_2, T_j)$. As $i \neq j$ we have that $\gamma_1' \cdot \gamma_2 = \gamma_2 \cdot \gamma_1'$, and therefore $\Sigma_{g,i}\Sigma_{h,j} = [T_j, \gamma_1' \cdot \gamma_2, T_j] = [T_j, \gamma_2 \cdot \gamma_1', T_j] = \Sigma_{h,j}\Sigma_{i,h}$. ∎

The second proposition describes the relationship between elements from the set $\mathfrak{L}$ and those in $\mathfrak{R}$.

Proposition 4.17. *The equality $\Gamma_{g,n} = \Sigma_{g,n+1}\Gamma_{(g)\theta,n+1}$ holds for all $n \geq 0$ and $g \in G$.*

Proof. The product $\Sigma_{g,n+1}\Gamma_{(g)\theta,n+1}$ can be represented by the augmented tree pair,



.

The final two leaves of both these trees form an exposed caret and thus one can use the shrinking operation to delete the caret from the augmented tree pair and create a representative of $\Gamma_{g,n}$ as required.

$\square$

The normal form for R. Thompson's group F

The normal form we are going to produce for elements of $F_{(G,\theta)}$ has its foundation in the normal form for Thompson's group F as described in [12]. Thus before we prove the normal form for $F_{(G,\theta)}$ we will first outline the normal form for F. We will not give any proofs, all of which can be found in [12].

One begins with the definition of an **exponent** of a leaf of a binary tree.

Definition 4.18 (Exponent of a leaf)**.** *Suppose T is a finite binary tree, and l is a leaf of T. The* ***exponent*** *of l is the length of the longest path of (upward) left branches in T which begins at l and does not reach the right-hand side of T.*

Example 4.19. Let T be the binary tree in Figure 4.9.

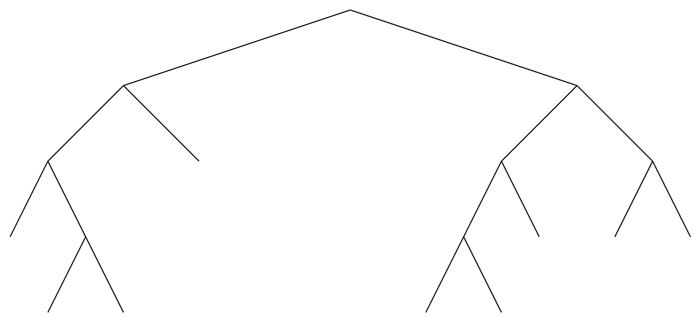


Figure 4.9

The exponents of the leaves for T, from left to right, are $2, 1, 0, 0, 2, 0, 0, 0, 0$.

Suppose $(\mathcal{D}, \mathcal{R})$ is an n-leaved binary tree pair that represents an element $f \in F$. Then by Theorem 2.5 in [12] the element f can be given by the product

$$f = X_0^{a_0} X_1^{a_1} \cdots X_n^{a_n} X_n^{-b_n} X_{n-1}^{-b_{n-1}} \cdots X_0^{-b_0}$$

where $a_0, \ldots, a_n$ are the exponents of $\mathcal{D}$ and $b_0, \ldots, b_n$ are the exponents of $\mathcal{R}$. (Recall that our products are multiplied from left to right and all the generators are inverses of those in [12], hence the difference in the exponents.) Further, the tree pair $(\mathcal{D}, \mathcal{R})$ is irreducible if and only if the following conditions are satisfied;

1. if the last two leaves of $\mathcal{D}$ lie in a caret then the last two leaves of $\mathcal{R}$ do not lie in a caret,

2. for all k in $\mathbb{N}_0$ with $k \leq n$, if $a_k > 0$ and $b_k > 0$ then at least one of either $a_{k+1} > 0$ or $b_{k+1} > 0$ must hold.

Thus if $f \in F$ then f can be expressed uniquely as the product

$$f = X_{i_0}^{a_0} X_{i_1}^{a_1} \cdots X_{i_m}^{a_m} X_{j_n}^{-b_n} X_{j_{n-1}}^{-b_{n-1}} \cdots X_{j_0}^{-b_0},$$

where $a_k, b_k \in \mathbb{N}_0$, $0 \leq i_0 < i_2 < \ldots < i_m$ and $0 \leq j_0 < j_2 < \ldots < j_n$, such that $i_m \neq j_n$ and when both X_k and X_k^{-1} both appear in the expression so does at least one of X_{k+1} or X_{k+1}^{-1}. Any element that has no negative exponents in its normal form we call **positive** and any that has no positive exponents we call **negative**. Thus every element $f \in F$ can be factorised as $f = PQ$ where P is a positive element and Q is a negative element. When written in this way we say that f is given in **pq form**. A subtle point needs to be made about the relationship between tree pairs and elements of F written in *pq* form. For every tree pair $(\mathcal{D}, \mathcal{R})$ there exists an associated element of F written in *pq* form. We call this element the **pq factorisation** of the tree pair $(\mathcal{D}, \mathcal{R})$, using the same language as in [11]. The important point is that every *pq* form is a *pq* factorisation for some (non-unique) tree pair, that is, for every product PQ in *pq* form there exists a tree pair $(\mathcal{D}, \mathcal{R})$ such that its *pq* factorisation is exactly PQ.

The normal form for $F_{(G,\theta)}$

We now return to consider the group $F_{(G,\theta)}$. Following the definitions of positive and negative elements in $F_{(G,\theta)}$, we now introduce an **augmenting element**, defined in three distinct types.

Definition 4.20 (Augmenting elements)**.** *An element from $F_{(G,\theta)}$ is said to be **augmenting** if it can be factorised into one of the following three types;*

Type 1: $\Gamma_{a,n}$,

Type 2: $\Sigma_{g_1,i_1} \Sigma_{g_2,i_2} \cdots \Sigma_{g_m,i_m}$,

Type 3: $\Sigma_{h_1,j_1} \Sigma_{h_2,j_2} \cdots \Sigma_{h_m,j_m} \Gamma_{b,l}$,

where $m, l \in \mathbb{N}_1$, $n \in \mathbb{N}_0$, $0 < i_1 < i_2 < \ldots < i_m$, $0 < j_1 < j_2 < \ldots < j_m \leq l$, and $a, b, g_k, h_k \in G \setminus \{1_G\}$ such that

1. $(g_m)\theta \neq 1_G$,

2. if $j_m = l$ then $(h_m)\theta \neq b$.

*We call each of these types an **augmenting product**.*

We call any augmented tree pair of the form (T_n, γ, T_n), for some non-trivial $\gamma \in G^{n+1}$, a **right sided** augmented tree pair. Recall that T_n is the $(n+1)$-leaved binary tree consisting solely of the right vine. The following proposition establishes a relationship between augmenting products and right sided augmented tree pairs.

Proposition 4.21. *Suppose $f \in F_{(G,\theta)}$ is a non-identity element that be can represented by a right sided augmented tree pair of the form $S = (T_n, \gamma, T_n)$ for some $\gamma \in G^{n+1}$. Then f can be factorised as an augmenting product of Type 1, 2 or 3. We call this the* ***augmentation factorisation*** *for f associated to the tree pair S.*

Furthermore, every augmenting product is an augmentation factorisation of an element of $F_{(G,\theta)}$ associated to some irreducible *right sided augmented tree pair as given below. (The variables below have their definitions given in Definition 4.20.)*

Type 1: *The augmented tree pair (T_n, γ_1, T_n) where*

$$\gamma_1(k) = \begin{cases} a & \textit{if } k = n \\ 1_G & \textit{otherwise.} \end{cases}$$

Type 2: *The augmented tree pair $(T_{i_m}, \gamma_2, T_{i_m})$ where*

$$\gamma_2(k) = \begin{cases} g_t & \textit{if } k = i_t \textit{ for some } 1 \leq t \leq m \\ 1_G & \textit{otherwise.} \end{cases}$$

Type 3: *The augmented tree pair (T_l, γ_3, T_l) where*

$$\gamma_3(k) = \begin{cases} b & \textit{if } k{=}l \\ h_t & \textit{if } k = j_t \textit{ for some } 1 \leq t \leq m \\ 1_G & \textit{otherwise.} \end{cases}$$

Proof. Let f be a non-trivial element of $F_{(G,\theta)}$ such that $f = [T_n, \gamma, T_n]$ for some $\gamma \in G^{n+1}$, this is fixed for the rest of the proof. Suppose that $n = 0$. Then $[T_0, \{g\}, T_0] = \Gamma_{g,0}$ is an augmenting product of Type 1 and we are done.

Suppose instead that $n > 0$ and $\gamma(n+1) = 1_G$. Let $0 < i_1 < i_2 < \ldots < i_m < n+1$ be m integers and $g_1, g_2, \ldots, g_m$ be m non-identity group elements from G such that γ satisfies

$$\gamma(k) = \begin{cases} g_l & \text{if } k = i_l \text{ for some } l \leq m \\ 1_G & \text{otherwise.} \end{cases}$$

For all $j \leq m$, the indexes i_j identify each leaf in the augmented binary tree that has a non-identity element associated to it. By the fact that f is non-trivial γ must have at least one non-trivial entry from G thus $m \geq 1$. As each integer i_l is distinct, (T_n, γ, T_n) can be factorised into the product

$$(T_n, \gamma_{g_1}, T_n)(T_n, \gamma_{g_2}, T_n) \cdots (T_n, \gamma_{g_m}, T_n),$$

where for each $1 \leq j \leq m$ the decoration γ_{g_j} is defined as

$$\gamma_{g_j}(k) = \begin{cases} g_j & \text{if } k = i_j \\ 1_G & \text{otherwise.} \end{cases}$$

The augmented tree pair (T_n, γ_{g_j}, T_n) is a representative of Σ_{g_j, i_j} and thus f can be factorised into the product

$$\Sigma_{g_1,i_1} \Sigma_{g_2,i_2} \cdots \Sigma_{g_m,i_m}.$$

If $(g_m)\theta \neq 1_G$ then by Definition 4.20 this is an augmenting product of Type 2. If instead $(g_m)\theta = 1_G$ then by Proposition 4.17 we can replace Σ_{g_m,i_m} with Γ_{g_m,i_m-1} and produce another factorisation of f, namely

$$\Sigma_{g_1,i_1} \Sigma_{g_2,i_2} \cdots \Sigma_{g_{m-1},i_{m-1}} \Gamma_{g_m,i_m-1}.$$

If $m = 1$ then this is an augmenting product of Type 1. Otherwise, if $i_{m-1} \neq i_m - 1$ or if $\theta_{g_{m-1}} \neq g_m$ then by Definition 4.20 the product is an augmenting product of Type 3. If both these conditions fail we once again apply Proposition 4.17 to the final two elements $\Sigma_{g_{m-1},i_{m-1}} \Gamma_{g_m,i_m-1}$ of the product and reduce the length of the product by one. We then repeat the process above, checking to see if the product is an augmenting product after each application of Proposition 4.17. As the product gets shorter with each application of Proposition 4.17 we will either eventually reach an augmenting product of Type 3 or be left with a single element of the form $\Gamma_{g_m,0}$ which is an augmenting product of Type 1.

Finally, suppose that $n > 0$ and $\gamma(n+1) = h \neq 1_G$. Again we let $0 < i_1 < i_2 < i_3 < \ldots < i_m < n+1$ be m integers and $g_1, g_2, \ldots, g_m$ be m non-identity group elements from G such that γ satisfies

$$\gamma(k) = \begin{cases} h & \text{if } k = n+1 \\ g_l & \text{if } k = i_l \text{ for some } l \leq m \\ 1_G & \text{otherwise.} \end{cases}$$

The same argument from the previous case proceeds. However, now the factorisation of

(T_n, γ, T_n) includes the factor (T_n, γ_h, T_n) where γ_h is defined as

$$\gamma_h(k) = \begin{cases} h & \text{if } k = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

This tree pair represents an element from the set $\mathfrak{R}$ namely $\Gamma_{h,n}$. Thus f can be factorised as

$$\Sigma_{g_1,i_1}\Sigma_{g_2,i_2}\cdots\Sigma_{g_m,i_m}\Gamma_{h,n}.$$

As $\gamma(n+1) \neq 1_G$ it is possible for $m = 0$ which gives the case $(T_n, \gamma, T_n) = (T_n, \gamma_h, T_n)$, thus the factorisation of f is an augmenting product of Type 1. If $m > 0$ then the product is an augmenting product of Type 3 if either $i_m \neq n$ or $(g_m)\theta \neq h$. If both of these conditions fail then we find ourselves in the same situation as we were when dealing with the previous case (when $\gamma(n+1) = 1_G$) and thus we can repeatedly use Proposition 4.17 to shorten the product until we reach an augmenting product of Type 1 or 3.

We now prove the second half of the proposition. Let A be an augmenting product. If A is of Type 1 then it has the form $\Gamma_{a,n}$ for some $a \in G \setminus \{1_G\}$ and $n \in \mathbb{N}_0$. This is the augmenting factorisation of the irreducible tree pair (T_n, γ_g, T_n) where γ_g is defined as

$$\gamma_g(k) = \begin{cases} g & \text{if } k = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

If A is of Type 2 then it has the form

$$\Sigma_{g_1,i_1}\Sigma_{g_2,i_2}\cdots\Sigma_{g_m,i_m}$$

where $g_j \in G \setminus \{1_G\}$ and $0 < i_1 < i_2 < \ldots < i_m$. One can check that this is the augmenting factorisation of the tree pair $(T_{i_m}, \gamma, T_{i_m})$, where γ is defined as;

$$\gamma_2(k) = \begin{cases} g_t & \text{if } k = i_t \text{ for some } 1 \leq t \leq m \\ 1_G & \text{otherwise.} \end{cases}$$

The only potential exposed caret in $(T_{i_m}, \gamma, T_{i_m})$ is the caret consisting of its final two leaves. The decoration on the left hand leaf of this caret is g_m, which by condition 1 of Definition 4.20 satisfies $(g_m)\theta \neq 1$. Thus the tree pair $(T_{i_m}, \gamma, T_{i_m})$ must be irreducible.

Finally, if A is of Type 3 then it has the form

$$\Sigma_{h_1 j_1}\Sigma_{h_2,j_2}\cdots\Sigma_{h_m,j_m}\Gamma_{b,l}$$

where $h_k, b \in G \setminus \{1_G\}$, $0 < i_1 < i_2 < \ldots < i_m \leq l$ and $m \in \mathbb{N}_1$. One can check that this is the augmenting factorisation of the tree pair (T_l, γ, T_l), where γ is defined as;

$$\gamma_3(k) = \begin{cases} b & \text{if k=l} \\ h_t & \text{if } k = j_t \text{ for some } 1 \leq t \leq m \\ 1_G & \text{otherwise.} \end{cases}$$

As above, the only potential exposed caret in $(T_{i_m}, \gamma, T_{i_m})$ is the caret consisting of its final two leaves. If $j_m = l$ then the decoration on the left hand leaf of this caret is h_m and on the right hand leaf is b, which by condition 2 of Definition 4.20 satisfies $(h_m)\theta \neq b$. If $j_m \neq l$ then the decoration of the left hand leaf is the identity. Thus, in either case, the tree pair (T_l, γ, T_l) must be irreducible.

□

We will use augmenting elements to create a natural factorisation for elements of $F_{(G,\theta)}$, similar to the *pq* factorisation for elements of F. The method is very similar to that used by Burillo, Cleary, Stein and Taback in [11] to create factorisations and normal forms of elements of Thompson's group T. In their paper the authors introduce a set of torsion generators that one can insert between positive and negative elements of F to produce factorisations of elements of T. It is then from these factorisations, which they call *pcq* factorisations, that their normal form arises. We will follow a similar line using the augmenting elements described above. Note that augmenting elements are also torsion. In fact there is a stronger observation regarding torsion elements of $F_{(G,\theta)}$.

Observation 4.22. *An element f in $F_{(G,\theta)}$ is torsion if and only if $f = [\mathcal{T}, \gamma, \mathcal{T}]$ for some arbitrary n-leaved binary tree $\mathcal{T}$ and some $\gamma \in G^n$.*

The first step is to use augmenting tree pairs to construct factorisations for elements of $F_{(G,\theta)}$. These factorisations will not be unique for the elements of $F_{(G,\theta)}$ but will provide the ground for our own normal form.

Lemma 4.23. *Suppose $S = (\mathcal{D}, \gamma, \mathcal{R})$ is an augmented tree pair such that $\mathcal{D}$ and $\mathcal{R}$ are two binary trees with $n+1$ leaves and $\gamma \in G^{n+1}$. Let $a_0, \ldots, a_n$ and $b_0, \ldots, b_n$ be the exponents of $\mathcal{D}$ and $\mathcal{R}$ respectively. Then the element in $F_{(G,\theta)}$ represented by S is given by the product;*

$$(X_0^{a_0} X_1^{a_1} \ldots X_n^{a_n}) \cdot A \cdot (X_n^{-b_n} X_{n-1}^{-b_{n-1}} \ldots X_0^{-b_0}), \tag{4.8}$$

where A is either the identity or an augmenting product of Type 1, 2 or 3. We call this the ***paq factorisation*** *associated to S.*

Furthermore, S is irreducible if and only if the following conditions hold;

1. *If the final two leaves of both $\mathcal{D}$ and $\mathcal{R}$ lie in a single caret then $\gamma(n+1) \neq (\gamma(n))\theta$.*
2. *If $a_k > 0$ and $b_k > 0$ then either*
   (a) *$a_{k+1} > 0$,*
   (b) *$b_{k+1} > 0$, or*
   (c) *The augmenting product A satisfies one of the following conditions;*
      i. *A contains $\Sigma_{g,k+1}$ for some g such that $(g)\theta = h_1 \neq 1_G$, and A does **not** contain either $\Sigma_{h_1,k+2}$ or $\Gamma_{h_1,k+1}$,*
      ii. *A contains $\Sigma_{g,k+1}$, for some g such that $(g)\theta = 1_G$, and A **also** contains either $\Sigma_{h_2,k+2}$ or $\Gamma_{h_2,k+1}$ for some non-trivial $h_2 \in G$,*
      iii. *A contains either $\Sigma_{g,k+2}$ or $\Gamma_{g,k+2}$ for some non-trivial $g \in G$, and A does not contain $\Sigma_{h_3,k+1}$ for any non-trivial $h_3 \in G$.*

Proof. Let $S = (\mathcal{D}, \gamma, \mathcal{R})$ be an augmented tree pair such that $\mathcal{D}$ and $\mathcal{R}$ are $(n+1)$-leaved binary trees and $\gamma \in G^{n+1}$.

We split the proof into two parts.

Part 1: paq factorisation of S

We can factorise S to be the product of tree pairs $(\mathcal{D}, 1_G^{n+1}, T_n)(T_n, \gamma, T_n)(T_n, 1_G^{n+1}, \mathcal{R})$. By Theorem 2.5 in [12], the function in $F_{(G,\theta)}$ with tree pair $(\mathcal{D}, 1_G^{n+1}, T_n)$ is $X_0^{a_0} X_1^{a_1} \ldots X_n^{a_n}$ where a_k is the exponent of the $(k+1)^{th}$ leaf of $\mathcal{D}$. Likewise, the function in $F_{(G,\theta)}$ with tree pair $(T_n, 1_G^{n+1}, \mathcal{R})$ is $X_n^{-b_n} X_{n-1}^{-b_{n-1}} \ldots X_0^{-b_0}$ where b_k is the exponent of the $(k+1)^{th}$ leaf of $\mathcal{R}$. Finally, let A be the element of $F_{(G,\theta)}$ that has the tree pair (T_n, γ, T_n). If $\gamma = 1_G^{n+1}$ then A is the identity in $F_{(G,\theta)}$. If γ is a non-identity element of G^{n+1} then by Proposition 4.21 A must be an augmenting element of Type 1, 2 or 3.

Therefore the element in $F_{(G,\theta)}$ that has tree pair

$$S = (\mathcal{D}, 1_G^{n+1}, \mathcal{R}) = (\mathcal{D}, 1_G^{n+1}, T_n)(T_n, \gamma, T_n)(T_n, 1_G^{n+1}, \mathcal{R})$$

is given by the product

$$(X_0^{a_0} X_1^{a_1} \ldots X_n^{a_n}) \cdot A \cdot (X_n^{-b_n} X_{n-1}^{-b_{n-1}} \ldots X_0^{-b_0}),$$

where A is either the identity or an augmenting element of Type 1, 2 or 3.

Part 2: Irreducibility of S

Suppose $S = (\mathcal{D}, \gamma, \mathcal{R})$ is an irreducible augmented tree pair, we will now show that conditions 1 and 2 of the lemma above must be true. Suppose the final two leaves of both $\mathcal{D}$ and $\mathcal{R}$ lie in a single caret. As S is irreducible this shared caret cannot be exposed and thus the decoration γ must satisfy $\gamma(n+1) \neq (\gamma(n))\theta$. This satisfies condition 1 of the lemma.

Suppose the $(k+1)$th leaf of S was such that its exponents in $\mathcal{D}$ and $\mathcal{R}$ satisfied $a_k > 0$ and $b_k > 0$ respectively. Any leaf that has a non-zero exponent must be a left leaf as all right leafs have exponent zero. Thus the $(k+1)$th leaf of S must be a left leaf of both $\mathcal{D}$ and $\mathcal{R}$. As S is irreducible we know that this leaf cannot lie in an exposed caret else the caret could be reduced. This gives us at most three possible situations for S (more than one of which may be true).

Situation 1: The $(k+2)$th leaf of $\mathcal{D}$ is a left leaf,

Situation 2: The $(k+2)$th leaf of $\mathcal{R}$ is a left leaf,

Situation 3: The decoration γ satisfies $\gamma(k+2) \neq (\gamma(k+1))\theta$.

As an exposed caret must consist of one left leaf and one right left, if either of the first two options are true then this guarantees that the $(k+1)$th leaf is not lying in an exposed caret. If the final condition is true then if the $(k+2)$th leaf of S is a right leaf then the decorations on those leaves will guarantee that the caret is not exposed. If none of these conditions are met then the $(k+2)$th leaf must be a right leaf and the decorations on the leaves will satisfy the conditions of an exposed caret. Hence, if S is irreducible it must satisfy at least one of the three situations above. We will consider each situation in turn and show that if it is true then either $2(a)$, $2(b)$ or $2(c)$ from Lemma 4.23 must also be true.

Situation 1: The $(k+2)$th leaf of $\mathcal{D}$ is a left leaf.

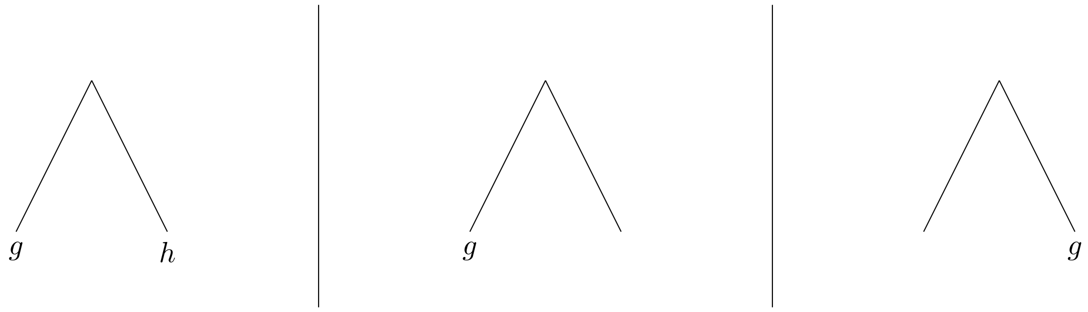
If the first situation is true then the exponent a_{k+1} of the $(k+2)$th leaf of $\mathcal{D}$ must be non-zero and thus condition 2(a) of Lemma 4.23 would be satisfied.

Situation 2: The $(k+2)$th leaf of $\mathcal{R}$ is a left leaf.

If the second situation is true then the exponent b_{k+1} of the $(k+2)$th leaf of $\mathcal{D}$ must be non-zero and thus condition 2(b) of Lemma 4.23 would be satisfied.

Situation 3: The decoration γ satisfies $\gamma(k+2) \neq (\gamma(k+1))\theta$.

Suppose Situation 3 is true and Situation 1 and Situation 2 are false. Then S will contain an exposed caret in one of three types drawn below, for g and h non-trivial elements of G. (By which we mean the $(k+1)^{th}$ and the $(k+2)^{th}$ leaves of both the domain tree and the range tree of S have the following form).



Note that at least one of decorations must be non-trivial to satisfy the property $\gamma(k+2) \neq (\gamma(k+1))\theta$ given on the decoration by Situation 3.

Suppose the shared caret of Situation 3 is in the **leftmost** form drawn above for some non-trivial g and h. If $(g)\theta \neq 1$ then $h \neq (g)\theta$ by the property $\gamma(k+2) \neq (\gamma(k+1))\theta$, and therefore condition $2(c)i$ of Lemma 4.23 must be true. If $(g)\theta = 1$ then condition $2(c)ii$ must be true.

Suppose the shared caret of Situation 3 is in the **middle** form drawn above for some non-trivial g. If $(g)\theta = 1_G$ then the caret would be reducible and thus a contradiction to the irreducibility of S. Thus $(g)\theta \neq 1_G$ and condition $2(c)i$ of Lemma 4.23 is satisfied.

Finally, suppose the shared caret of Situation 3 is in the **rightmost** form drawn above for some non-trivial g. Then condition $2(c)iii$ of Lemma 4.23 is satisfied.

Therefore we have shown that if $a_k > 0$ and $b_k > 0$ then at least one of the options given in condition 2 of Lemma 4.23 must be true. Hence this direction of the proof is done.

Now suppose $S = (\mathcal{D}, \gamma, \mathcal{R})$ is an augmented tree pair such that its *paq* factorisation satisfies the conditions given in Lemma 4.23. By condition 1 the last two leaves of S cannot form an exposed caret. For a shared caret in S that does not consist of the final two leaves to be exposed the exponent of its left leaf must be non-zero in both $\mathcal{D}$ and $\mathcal{R}$.

Consider the kth leaf of S where $k < n$ and suppose that its exponents a_{k-1} and b_{k-1} are both non-zero in $\mathcal{D}$ and $\mathcal{R}$ respectively. Condition 2 gives three possible options that could be true for S, we will show that the kth leaf cannot be part of an exposed caret should any of these options be true. The first option is condition 2(a), namely that the exponent a_n of the $(k+1)$th leaf of $\mathcal{D}$ is non zero. As mentioned before, if a leaf has a non-zero exponent it must be a left leaf as all right leaves have zero exponent by definition. Thus the kth and $(k+1)$th leaves of $\mathcal{D}$ are both left leaves and hence the kth leaf of S cannot be in an exposed caret. Condition 2(b) gives the same situation but this time we are considering leaves in the range tree $\mathcal{R}$. As before the conclusion is that the kth leaf of S cannot be in an exposed caret. This leaves us with the final condition 2(c), which gives three possible conditions on the augmented product A. The product A is determined by the tree S thus we will consider each subproduct in turn and determine what decorations on the kth and $(k+1)$th leaves of S could create such a subproduct in A. In each case we will show that the decorations imply an irreducible caret.

2(c)i: A contains $\Sigma_{g,k+1}$ for some g such that $(g)\theta = h_1 \neq 1_G$, and A does **not** contain either $\Sigma_{h_1,k+2}$ or $\Gamma_{h_1,k+1}$,

The decoration on the $(k+1)$th leaf must therefore be g. As the decoration on the $(k+2)$th is not $(g)\theta$ the caret cannot be irreducible.

2(c)ii: A contains $\Sigma_{g,k+1}$, for some g such that $(g)\theta = 1_G$, and A **also** contains either $\Sigma_{h_2,k+2}$ or $\Gamma_{h_2,k+1}$ for some non-trivial $h_2 \in G$,

The decorations on the $(k+1)$th and the $(k+2)$th leaves must be g and h respectively. As $(g)\theta = 1_G \neq h$ the caret must be irreducible.

2(c)iii: A contains either $\Sigma_{g,k+2}$ or $\Gamma_{g,k+2}$ for some non-trivial $g \in G$, and A does not contain $\Sigma_{h_3,k+1}$ for any non-trivial $h_3 \in G$.

The decoration on the right $(k+2)$th leaf must therefore be g and the decoration on the left $(k+1)$th leaf must be the identity. Therefore the caret must be irreducible.

Therefore, if $a_k > 0$ and $b_k > 0$ then the $(k+1)$th leaf cannot be part of an exposed caret. Thus S contains no exposed carets and hence S must be irreducible. □

Corollary 4.24. *The set $X = \mathfrak{X} \cup \mathfrak{L} \cup \mathfrak{R}$ is a generating set for $F_{(G,\theta)}$.*

In Lemma 4.23 above we factorised an element of $F_{(G,\theta)}$ into a product of three parts, a positive product P, a negative product Q and an augmenting element A. Any product in $F_{(G,\theta)}$ that has this particular form we will call a **paq product** or a product in *paq* form. Note that a product in *paq* form does **not** have to satisfy the conditions for irreducibility in Lemma 4.23. The next result we need is to show that every product in *paq* form is in fact a *paq* factorisation for some augmented tree pair.

Lemma 4.25. *Consider the product PAQ where P is a positive product, Q is a negative product and A is an augmenting product of Type 1, 2 or 3. Then there exists an augmented tree pair S such that the paq factorisation of S is the product PAQ.*

Proof. Let PAQ be a product in the generators of Λ such that P is a positive product, Q is a negative product and A is an augmenting product of Type 1, 2 or 3. Suppose $P_T = (\mathcal{D}, T_n)$ is the unique irreducible representative of P, $Q_T = (T_m, \mathcal{R})$ is the unique irreducible representative of Q and $A_T = (T_k, \gamma, T_k)$ is the unique irreducible representative of A. Consider the augmented tree pair $S = P_T A_T Q_T$ created by multiplying these three tree pairs together. Our claim is that the *paq* factorisation of S is exactly the product PAQ. To create S one must first take the product of tree pairs

$$(\mathcal{D}, T_n)(T_k, \gamma, T_k)(T_m, \mathcal{R})$$

and apply the splitting relation to each tree pair where necessary to produce the product

$$(\mathcal{D}', T_M)(T_M, \gamma', T_M)(T_M, \mathcal{R}'),$$

for some $M \geq \max\{n, m, k\}$. As any such value of M will suffice we will choose to set M to be the smallest possible i.e. $M = \max\{n, m, k\}$. The important point to note is that the splitting operation is only ever applied to the final leaf of any of the tree pairs. If $n = m = k$ then no splitting operations are required and we immediately get our result that the *paq* factorisation of S is PAQ. Suppose $n < M$, then $(\mathcal{D}', T_M)$ is produced by splitting the final leaf of P_T $(M - n)$-times. As we are only ever adding carets to the final leaf, the exponents of the new leaves that are created are always zero. Thus the positive product that is associated to $(\mathcal{D}', T_M)$ is the same as that for P_T, which is P. The same argument also holds for the case where $m < M$ where the negative product associated to $(T_M, \mathcal{R}')$ must be Q. Finally, suppose $k < M$ and we are required to split the final leaf of A_T. Since A_T is the reduced version of (T_M, γ', T_M) by definition the augmentation factorisation associated to (T_M, γ', T_M) must be the same as that for A_T, which is A. Thus regardless of which tree pairs need to be expanded the augmented tree pair S has *paq* factorisation PAQ.

In this case when $M = \max\{n, m, k\}$ we call S the **canonical** augmented tree pair associated to PAQ.

□

Observation 4.26. *Suppose $S = (\mathcal{D}, \gamma, \mathcal{R})$ is the canonical augmented tree pair associated to a paq product PAQ. If the final two leaves of S lie in a shared caret then the decoration on the final two leaves must satisfy $\gamma(n+1) \neq (\gamma(n))\theta$.*

Proof. Let $S = (\mathcal{D}, \gamma, \mathcal{R})$ be the canonical augmented tree pair associated to a *paq* product PAQ and suppose the final two leaves of S lie in a shared caret. As the exponents of all the leaves in this shared caret are zero the caret cannot exist in either the irreducible representative of P or the irreducible representative of Q. Therefore, it must have come from the irreducible representative of augmenting product A. The final two leaves of this irreducible representative lie in a shared caret, therefore the decoration on the leaves must satisfy $\gamma(n+1) \neq (\gamma(n))\theta$ else the caret would be exposed. □

Theorem 4.27. *For every non-identity element in $F_{(G,\theta)}$, there exists a unique normal* *form, written in the generators* Λ *as*

$$X_0^{a_0} X_1^{a_1} \dots X_n^{a_n} \cdot A \cdot X_n^{-b_n} X_{n-1}^{-b_{n-1}} \dots X_0^{-b_0}, \tag{4.9}$$

where $n, a_0, \dots, a_n, b_1, \dots, b_n$ *are non-negative integers, and* A *is either empty or an augmenting product of Type 1, 2 or 3, such that if* $a_k > 0$ *and* $b_k > 0$ *for any natural number* $k \leq n$ *then either*

1. $a_{k+1} > 0$,
2. $b_{k+1} > 0$, *or*
3. *The augmenting product* A *satisfies one of the following conditions;*
   (a) A *contains* $\Sigma_{g,k+1}$ *for some* g *such that* $(g)\theta = h_1 \neq 1_G$, *and* A *does* ***not*** *contain either* $\Sigma_{h_1,k+2}$ *or* $\Gamma_{h_1,k+1}$,
   (b) A *contains* $\Sigma_{g,k+1}$, *for some* g *such that* $(g)\theta = 1_G$, *and* A ***also*** *contains either* $\Sigma_{h_2,k+2}$ *or* $\Gamma_{h_2,k+1}$ *for some non-trivial* $h_2 \in G$,
   (c) A *contains either* $\Sigma_{g,k+2}$ *or* $\Gamma_{g,k+2}$ *for some non-trivial* $g \in G$, *and* A *does not contain* $\Sigma_{h_3,k+1}$ *for any non-trivial* $h_3 \in G$.

Proof. By Lemma 4.23 the *paq* factorisation of an irreducible augmented tree pair S satisfies the conditions of the normal form. Therefore, as every element of $F_{(G,\theta)}$ can be represented by an irreducible augmented tree pair, every element can also be written in the normal form.

We now prove uniqueness. Let PAQ be a *paq* product that satisfies the conditions of the theorem. By Lemma 4.25, PAQ is the *paq*-factorisation associated to some augmented tree pair $S = (\mathcal{D}, \gamma, \mathcal{R})$. We now show that S must be irreducible, that is, we show that it satisfies the conditions of an irreducible augmented tree pair given in Lemma 4.23. Recall that the first condition of Lemma 4.23 requires that if the final two leaves of $\mathcal{D}$ and $\mathcal{R}$ lie in a single caret then $\gamma(n+1) \neq (\gamma(n))\theta$. By Observation 4.26, S satisfies this first condition. Notice that the second condition of Lemma 4.23 is exactly the same as the condition already placed upon S by the properties of PAQ imposed by the theorem. Thus S must be irreducible. Suppose $P_1A_1Q_1$ and $P_2A_2Q_2$ are two distinct *paq* products that satisfy the conditions of the theorem and represent the same element in $F_{(G,\theta)}$. By Lemma 4.25 the product $P_1A_1Q_1$ is the *paq*-factorisation associated to some augmented tree pair S_1 and the product $P_2A_2Q_2$ is the *paq*-factorisation associated to some augmented tree pair S_2. From the discussion above both S_1 and S_2 must be irreducible. Since each element of $F_{(G,\theta)}$ has a unique irreducible augmented tree pair representative by Proposition 4.11, we must have $S_1 = S_2$. Therefore $P_1A_1Q_1 = P_2A_2Q_2$, which is a contradiction to the original assumption that they were distinct.

Therefore, each element in $F_{(G,\theta)}$ must have a unique normal form representative as defined by the theorem. $\square$

4.3.2 Presentation

Let $G := \{g_i\}_{i \in \boldsymbol{N}}$ be a finite group of order N, with $g_1 = 1_G$, and let $\theta : G \to G$ be an endomorphism of G. Recall that the function $\delta : \boldsymbol{N} \to \boldsymbol{N}$ is defined by the rule $(i)\delta = j$ if and only if $(g_i)\theta = g_j$. Also recall that the function $\Delta : \boldsymbol{N} \times \boldsymbol{N} \to \boldsymbol{N}$ is defined by $(i,j)\Delta = k$ if and only if $g_i g_j = g_k$. Let the group $\mathcal{F}^{inf}_{(G,\theta)} = \langle X | R \rangle$ be following infinite presentation with generators X and relations R defined below.

Set of generators: $X := \boldsymbol{X1} \sqcup \boldsymbol{X2} \sqcup \boldsymbol{X3}$:

$\mathbf{X1} = \{Y_n | n \in \mathbb{N}_0\}$,
$\mathbf{X2} = \{S_{i,n} | i \in \boldsymbol{N}, n \in \mathbb{N}_1\}$,
$\mathbf{X3} = \{T_{i,n} | i \in \boldsymbol{N}, n \in \mathbb{N}_0\}$.

Set of relations: $R := \bigsqcup_{i=1}^{12} \boldsymbol{Ri}$
(Note that we no longer use set notation as we did for the generators above to make the sets of relations easier to read.)

- **R1** $Y_k^{-1} Y_n Y_k = Y_{n+1},$ for all k and n in $\mathbb{N}_0$ such that $k < n$,
- **R2** $Y_k^{-1} S_{i,n} Y_k = S_{i,n+1},$ for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n-1$,
- **R3** $Y_k^{-1} S_{i,n} Y_k = S_{i,n},$ for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k \geq n$,
- **R4** $Y_{n-1}^{-1} S_{i,n} Y_{n-1} = S_{i,n} S_{(i)\delta,n+1},$ for all i in $\boldsymbol{N}$ and n in $\mathbb{N}_1$,
- **R5** $Y_k^{-1} T_{i,n} Y_k = T_{i,n+1},$ for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n$,
- **R6** $S_{i,n} S_{j,m} = S_{j,m} S_{i,n},$ for all i and j in $\boldsymbol{N}$ and n and m in $\mathbb{N}_1$ such that $n > m$,
- **R7** $S_{i,n} S_{j,n} = S_{(i,j)\Delta,n},$ for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_1$,
- **R8** $T_{i,n} T_{j,n} = T_{(i,j)\Delta,n},$ for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_0$,
- **R9** $T_{i,n} = S_{i,n+1} T_{(i)\delta,n+1},$ for all i in $\boldsymbol{N}$ and n in $\mathbb{N}_0$,
- **R10** $S_{i,k} T_{j,n} = T_{j,n} S_{i,k},$ for all i and j in $\boldsymbol{N}$ and k and n in $\mathbb{N}_1$, such that $k \leq n$,
- **R11** $S_{1,n} = 1,$ for all n in $\mathbb{N}_1$,
- **R12** $T_{1,n} = 1,$ for all n in $\mathbb{N}_0$.

We note that the relations **R11** and **R12** can remove the extraneous generators $S_{1,n}$ and $T_{1,n}$ but we retain them for the ease of writing the other relations in R.

Before we move on, we define some notation that will appear later on in the paper. We will refer to generators of the form Y_n as **Y-generators**, $S_{i,n}$ as **S-generators** and $T_{i,n}$ as **T-generators**. In addition, for the elements $S_{i,n}$ and $T_{i,n}$ we refer the subscript i as the **augmenting index** and the subscript n as the **depth index**. The definition of the depth index n also hold for the generators X_n.

The proof of the following theorem will be proof for Theorem 4.2 from the introduction.

Theorem 4.28. *Let G be a finite group of order N and $\theta : G \to G$ be a group homomorphism from G to itself. Then there exists an isomorphism $\phi : \mathcal{F}^{inf}_{(G,\theta)} \to F_{(G,\theta)}$ such that $\phi(Y_i) = X_i$, $\phi(S_{i,j}) = \Sigma_{g_i,j}$ and $\phi(T_{i,j}) = \Gamma_{g_i,j}$.*

Proof. Define a map ϕ from the free group F_X generated by the set of formal symbols $X := \{Y_n | n \in \mathbb{N}_0\} \sqcup \{S_{i,j} | i \in \boldsymbol{N}, j \in \mathbb{N}_1\} \sqcup \{T_{i,j} | i \in \boldsymbol{N}, j \in \mathbb{N}_1\}$ to the group $F_{(G,\theta)}$ by $\phi(Y_i) = X_i$, $\phi(S_{i,j}) = \Sigma_{g_i,j}$, and $\phi(T_{i,j}) = \Gamma_{g_i,j}$. From Corollary 4.24 we know that the set $\phi(X)$ generates $F_{(G,\theta)}$, thus to prove that ϕ is a surjective homomorphism from $\mathcal{F}^{inf}_{(G,\theta)}$ to $F_{(G,\theta)}$ we must show that the twelve sets of relations in $\mathcal{F}^{inf}_{(G,\theta)}$ hold in $F_{(G,\theta)}$. We consider each family of relations in turn.

R1. $Y_k^{-1} Y_n Y_k = Y_{n+1}$, for all k and n in $\mathbb{N}_0$ such that $k < n$.

Let n and k be in $\mathbb{N}_0$ such that $k < n$. Under the homorphism ϕ the product $Y_k^{-1} Y_n Y_k$ is taken to the product $X_k^{-1} X_n X_k$ in $F_{(G,\theta)}$, which is equal to X_{n+1} by Theorem 3.4 in [12].

R2. $Y_k^{-1} S_{i,n} Y_k = S_{i,n+1}$, for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n - 1$.

Let k be in $\mathbb{N}_0$ and n be in $\mathbb{N}_1$ such that $k < n - 1$. Under the homomorphism ϕ the product $Y_k^{-1} S_{i,n} Y_k$ is taken to the product $X_k^{-1} \Sigma_{g_i,n} X_k$ in $F_{(G,\theta)}$. We will evaluate this product by using multiplication of tree pairs. We begin with the tree pair representatives of X_k, X_k^{-1} and $\Sigma_{g_i,n}$ as drawn explicitly in Figures 4.6 and 4.7 earlier in the chapter. We can write these concisely as tuples;

$$\begin{aligned} X_k : &\quad (\mathcal{D}, 1_G^{k+3}, T_{k+2}), \\ X_k^{-1} : &\quad (T_{k+2}, 1_G^{k+3}, \mathcal{D}), \\ \Sigma_{g_i,n} : &\quad (T_n, \gamma_{i,n}, T_n), \end{aligned}$$

where $\mathcal{D}$ is the domain tree given in Figure 4.6, T_{k+2} is the right sided tree with $k+3$ leaves, T_n is the right sided tree with $n+1$ leaves, 1_G^{k+3} is the identity of G^{k+3} and $\gamma_{i,n}$ is an element of G^{n+1} defined by

$$\gamma_{i,n}(j) = \begin{cases} g_i & \text{if } j = n \\ 1_G & \text{otherwise.} \end{cases}$$

To find a representative tree pair for the product $X_k^{-1} \Sigma_{g_i,n} X_k$ one follows the rules for augmented tree pair multiplication given earlier in the chapter. We begin with the three tree pairs being multiplied in the following order

$$(T_{k+2}, 1_G^{k+3}, \mathcal{D})(T_n, \gamma_{i,n}, T_n)(\mathcal{D}, 1_G^{k+3}, T_{k+2}). \tag{4.10}$$

The first step is to expand the tree pairs using the splitting operation such that under the expansion the trees $\mathcal{D}$ and T_n become the same tree which we will call $\mathcal{D}'$. Applying the following operations to the tree pairs will give the desired result.

$$\begin{aligned}
(T_{k+2}, 1_G^{k+3}, \mathcal{D}) &\Rightarrow (T_{n+1}, 1_G^{n+2}, \mathcal{D}') && \text{by splitting the final leaf } (n-k-1)\text{-times,}\\
(T_n, \gamma_{i,n}, T_n) &\Rightarrow (\mathcal{D}', \gamma'_{i,n}, \mathcal{D}') && \text{by splitting the } (k+1)\text{th leaf once,}\\
(\mathcal{D}, 1_G^{k+3}, T_{k+2}) &\Rightarrow (\mathcal{D}', 1_G^{n+2}, T_{n+1}) && \text{by splitting the final leaf } (n-k-1)\text{-times,}
\end{aligned}$$

where the decoration $\gamma'_{i,n}$ is an element of G^{n+2}. The decoration $\gamma'_{i,n}$ will be determined by the group element that was associated to the $(k+1)$th leaf of $(T_n, \gamma_{i,n}, T_n)$ when it was split. As the only non-trivial decoration was on the nth leaf of $(T_n, \gamma_{i,n}, T_n)$, the decoration on the $(k+1)$th leaf must be trivial as $k < n-1$. Therefore $\gamma'_{i,n}$ is given by;

$$\gamma'_{i,n}(j) = \begin{cases} g_i & \text{if } j = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

Thus the product (4.10) now becomes;

$$(T_{n+1}, 1_G^{n+2}, \mathcal{D}')(\mathcal{D}', \gamma'_{i,n}, \mathcal{D}')(\mathcal{D}', 1_G^{n+2}, T_{n+1}), \tag{4.11}$$

which gives the single augmented tree pair

$$(T_{n+1}, \gamma'_{i,n}, T_{n+1}).$$

This tree pair is, by definition, an augmented tree pair representative of $\Sigma_{g_i,n+1}$, hence we have proved this step.

As an example of the above multiplication consider the case when $n = 3$ and $k = 0$. We draw the tree pairs below and use dotted lines to indicate the expansion of the tree pairs under the operations described above. Notice how the expansions guarantee that the four interior trees are all identical.

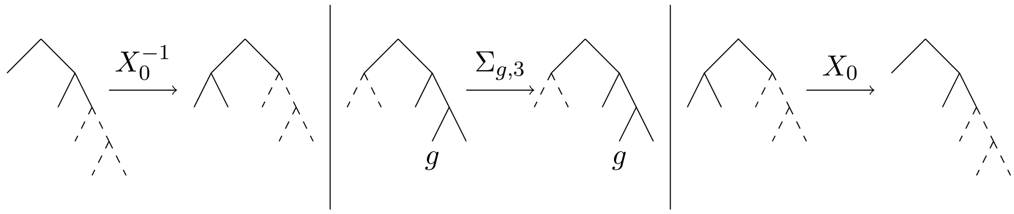


Figure 4.10: The multiplication of $X_0^{-1}\Sigma_{g,3}X_0$

The resulting product gives the following tree pair which is indeed a representative of $\Sigma_{g,4}$ as expected.

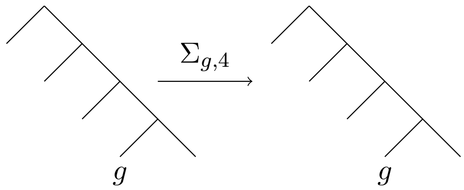


Figure 4.11: The resulting tree pair representing the product $X_0^{-1}\Sigma_{g,4}X_0$

R3. $Y_k^{-1}S_{i,n}Y_k = S_{i,n}$, for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k \geq n$.

Let k be in $\mathbb{N}_0$ and n be in $\mathbb{N}_1$ such that $k \geq n$. Under the map ϕ the product $Y_k^{-1}S_{i,n}Y_k$ is taken to the product $X_k^{-1}\Sigma_{g_i,n}X_k$ in $F_{(G,\theta)}$. We will again use tree pair multiplication to understand the product. We begin with the same three types of tree that we began with when considering the case for **R2** previously;

$$\begin{aligned} X_k : &\quad (\mathcal{D}, 1_G^{k+3}, T_{k+2}), \\ X_k^{-1} : &\quad (T_{k+2}, 1_G^{k+3}, \mathcal{D}), \\ \Sigma_{g_i,n} : &\quad (T_n, \gamma_{i,n}, T_n). \end{aligned}$$

However, in this case $k \geq n$ and thus the splitting operations we apply to the three trees will be different. Notice that for any values of k or n that satisfy the inequality the only tree that needs to be expanded is $(T_n, \gamma_{i,n}, T_n)$. This is because T_n is already a subtree of $\mathcal{D}$. When expanding $(T_n, \gamma_{i,n}, T_n)$ one begins by expanding the final leaf $(k-n+1)$-times. Then to produce $\mathcal{D}$ one must finally expand the penultimate leaf once. This results in the following product of tree pairs;

$$(T_{k+2}, 1_G^{k+3}, \mathcal{D})(\mathcal{D}, \gamma'_{i,n}, \mathcal{D})(\mathcal{D}, 1_G^{k+3}, T_{k+2}). \tag{4.12}$$

Notice that each leaf split in the operation above has been decorated with the identity from G. Thus $\gamma'_{i,n}$, though now an element of G^{k+3}, is given by;

$$\gamma'_{i,n}(j) = \begin{cases} g_i & \text{if } j = n \\ 1_G & \text{otherwise.} \end{cases}$$

Thus the tree pair

$$(T_{k+2}, \gamma'_{i,n}, T_{k+2})$$

can be reduced back down to the tree pair

$$(T_n, \gamma_{i,n}, T_n),$$

which represents the element $\Sigma_{g_i,n}$ as required.

As an example consider the case $k = n = 1$.

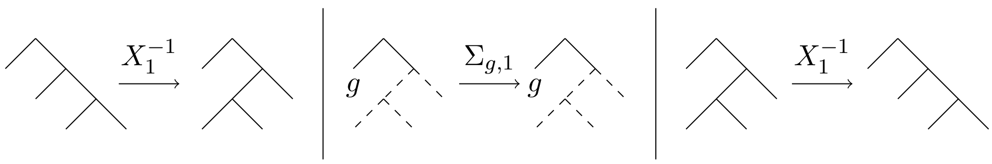


Figure 4.12: The multiplication of $X_1^{-1}\Sigma_{g,1}X_1$

Observe the carets added to the tree pair representation of $\Sigma_{g,1}$ will be exposed in the product $X_1^{-1}\Sigma_{g,1}X_1$ and therefore the resulting element is again $\Sigma_{g,1}$.

R4. $Y_{n-1}^{-1}S_{i,n}Y_{n-1} = S_{i,n}S_{(i)\delta,n+1}$, for all i in $\boldsymbol{N}$ and n in $\mathbb{N}_1$.

Let n be in $\mathbb{N}_1$. Under the map ϕ the product $Y_{n-1}^{-1}S_{i,n}Y_{n-1}$ is taken to the product $X_{n-1}^{-1}\Sigma_{g_i,n}X_{n-1}$ in $F_{(G,\theta)}$. Consider the tree pair representatives for these elements as before;

$$\begin{aligned} X_{n-1}: &\quad (\mathcal{D}, 1_G^{n+2}, T_{n+1}), \\ X_{n-1}^{-1}: &\quad (T_{n+1}, 1_G^{n+2}, \mathcal{D}), \\ \Sigma_{g_i,n}: &\quad (T_n, \gamma_{n+2}, T_n). \end{aligned}$$

As in the case we previously considered to product the tree pair representing the product of these three elements we need only expand the tree pair $(T_n, \gamma_{i,n}, T_n)$ as T_n is a subtree of $\mathcal{D}$. In fact, by splitting only the nth leaf of this tree pair the tree T_n becomes the tree $\mathcal{D}$. Thus the tree pair multiplication becomes;

$$(T_{n+1}, 1_G^{n+2}, \mathcal{D})(\mathcal{D}, \gamma'_{i,n}, \mathcal{D})(\mathcal{D}, 1_G^{n+2}, T_{n+1}). \tag{4.13}$$

The decoration $\gamma'_{i,n}$ takes some careful consideration. The leaf that we split in the original tree pair $(T_n, \gamma_{i,n}, T_n)$ was the nth leaf and thus by the definition of $\gamma_{i,n}$ had the non-trivial decoration g associated to it. Thus by splitting leaf and producing the tree pair $(\mathcal{D}, \gamma'_{i,n}, \mathcal{D})$ the decoration $\gamma'_{i,n}$ is defined by;

$$\gamma'_{i,n}(j) = \begin{cases} g_i & \text{if } j = n \\ (g_i)\theta & \text{if } j = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

Thus the tree pair that we produce from our product

$$(T_{n+1}, \gamma'_{i,n}, T_{n+1})$$

must represent the product $\Sigma_{g_i,n}\Sigma_{(g_i)\theta,n+1}$, as required.

As an example consider the case when $k = 0$ and $n = 1$.

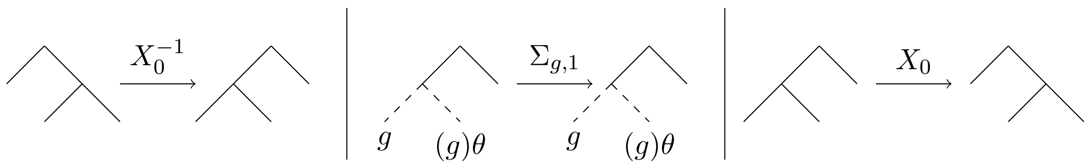
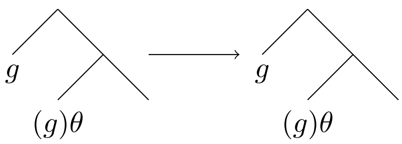


Figure 4.13: The multiplication of $X_0^{-1}\Sigma_{g,1}X_0$

In the figures above we have split the first leaf of the reduced tree pair representative of S_1 and in doing so have split the decoration on the tree. When this decoration is passed to the trees representing X_0^{-1} and X_0 upon multiplication, the resulting tree pair is as below.



The augmented tree pair above is a representative of the element $\Sigma_{g,1}\Sigma_{(g)\theta,2}$, as expected from the relation.

R5. $Y_k^{-1}T_{i,n}Y_k = T_{i,n+1}$, for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n$.

Let k be in $\mathbb{N}_0$ and n be in $\mathbb{N}_1$ such that $k < n$. Under the map ϕ the product $Y_k^{-1}T_{i,n}Y_k$ is taken to the product $X_k^{-1}\Gamma_{i,n}X_k$ in $F_{(G,\theta)}$. This time we require a tree pair representative for $\Gamma_{i,n}$, which is provided explicitly by Figure 4.8 earlier in the chapter. Written as a tuple, this tree pair is given by $(T_n, \gamma_\Gamma, T_n)$ where γ_Γ is in G^{n+1} and defined by

$$\gamma_\Gamma(j) = \begin{cases} g_i & \text{if } j = n+1 \\ 1_G & \text{otherwise.} \end{cases}$$

When performing the tree pair multiplication we use the same tree pairs for X_k and X_k^{-1} as before. Thus we begin the product of tree pairs;

$$(T_{k+2}, 1_G^{k+3}, \mathcal{D})(T_n, \gamma_\Gamma, T_n)(\mathcal{D}, 1_G^{k+3}, T_{k+2}). \tag{4.14}$$

Taking the product of these tree pairs requires us to expand the three trees so that the trees $\mathcal{D}$ and T_n both become the same tree which we call $\mathcal{D}'$. The operations required to achieve this are the same as the **R2** case;

$$\begin{aligned} (\mathcal{D}, 1_G^{k+3}, T_{k+2}) &\Rightarrow (\mathcal{D}', 1_G^{n+2}, T_{n+1}) && \text{by splitting the final leaf } (n-k-1)\text{-times,} \\ (T_n, \gamma_\Gamma, T_n) &\Rightarrow (\mathcal{D}', \gamma'_\Gamma, \mathcal{D}') && \text{by splitting the } (k+1)\text{th leaf once,} \\ (T_{k+2}, 1_G^{k+3}, \mathcal{D}) &\Rightarrow (T_{n+1}, 1_G^{n+1}, \mathcal{D}') && \text{by splitting the final leaf } (n-k-1)\text{-times,} \end{aligned}$$

The tree $\mathcal{D}'$ created by these operations can be again thought of as the tree T_n with an extra caret attached to the $(k+1)$th leaf. Notice that when $k = n-1$ we have $\mathcal{D}' = \mathcal{D}$. This is because in this case the tree T_n is a subtree of $\mathcal{D}$.

The decoration γ'_Γ is a group element in G^{n+2} and was created from the splitting of the $(k+1)$th leaf of $(T_n, \gamma_\Gamma, T_n)$. The only non-trivial decoration on $(T_n, \gamma_\Gamma, T_n)$ is on the $(n+1)$th leaf, thus for any value of k the decoration on the $(k+1)$th leaf must be trivial. Thus the new decoration γ'_Γ is defined by;

$$\gamma'_\Gamma(j) = \begin{cases} g_i & \text{if } j = n+2 \\ 1_G & \text{otherwise.} \end{cases}$$

Therefore the product of tree pairs produces

$$(T_{n+1}, \gamma'_\Gamma, T_{n+1})$$

which is a representative of the element $\Gamma_{g_i,n+1}$ as required.

R6. $S_{i,n}S_{j,m} = S_{j,m}S_{i,n}$, for all i and j in $\boldsymbol{N}$ and n and m in $\mathbb{N}_1$ such that $n > m$.

Let i and j be in $\boldsymbol{N}$, and n and m be in $\mathbb{N}_1$ such that $n > m$. Under the function ϕ the product $S_{i,n}S_{j,m}$ becomes $\Sigma_{g_i,n}\Sigma_{g_j,m}$, which by Lemma 4.16 commutes.

R7. $S_{i,n}S_{j,n} = S_{(i,j)\Delta}$, for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_1$.

Let i and j be in $\boldsymbol{N}$ and n be in $\mathbb{N}_1$ such that $(i,j)\Delta \neq 1$. The product $\phi(S_{i,n}S_{j,n}) = \Sigma_{g_i,n}\Sigma_{g_j,n} = \Sigma_{g_ig_j,n}$. If $g_ig_j = g_k$, then by definition of the function Δ, $\phi(S_{(i,j)\Delta}) = \Sigma_{g_k,n}$ and **R7** holds in $F_{(G,\theta)}$.

R8. Follows by the same argument as **R7.**

R9. $T_{i,n} = S_{i,n+1}T_{(i)\delta,n+1}$, for all i in $\boldsymbol{N}$ and n in $\mathbb{N}_1$.

This particular set of relations is important in the context of $F_{(G,\theta)}$ as it represents the splitting of a leaf. Let n be in $\mathbb{N}_0$ and i be in $\boldsymbol{N}$. Under the map ϕ, $T_{i,n}$ gets mapped to $\Gamma_{g_i,n}$ which can be represented by the leaf map $(1^n, 1_G) \to (1^n, g_i)$. Splitting this leaf map gives us an equivalent element in $F_{(G,\theta)}$ consisting of the leaf maps $(1^n0, 1_G) \to (1^n0, g_i)$ and $(1^n1, 1_G) \to (1^n1, (g_i)\theta)$. Thus by the splitting operation $\Gamma_{g_i,n} = \Sigma_{g_i,n+1}\Gamma_{(g_i)\theta,n+1}$ on augmented tree pairs, the set of relations **R9** hold in $F_{(G,\theta)}$.

R10. $S_{i,k}T_{j,n} = T_{j,n}S_{i_k}$ for all i and j in $\boldsymbol{N}$ and k and n in $\mathbb{N}_1$, such that $k \leq n$.

Let i and j be in $\boldsymbol{N}$, and k and n be in $\mathbb{N}_1$ such that $k \leq n$. The non-trivial leaf map of $\phi(S_{i,k}) = \Sigma_{g_i,k}$ is $(1^{k-1}0, 1_G) \to (1^{k-1}0, g_i)$ and the non-trivial leaf map of $\phi(T_{j,n}) = \Gamma_{g_j,n}$ is $(1^{k-1}1, 1_G) \to (1^{k-1}1, g_j)$. The prefixes $1^{k-1}0$ and $1^{n-1}1$ are incomparable if $k \leq n$ and thus the two elements $\Sigma_{g_i,k}$ and $\Gamma_{g_j,n}$ commute. Hence **R10** holds in $F_{(G,\theta)}$.

R11. $S_{1,n} = 1$ for all n in $\mathbb{N}_1$ and
R12. $T_{1,n} = 1$ for all n in $\mathbb{N}_0$.

The augmented tree pair representatives of $\phi(S_{1,n})$ and $\phi(T_{1,n})$ are the same and consists of the augmented tree pair $(T_n, 1_G^{n+1}, T_n)$ with the identity decoration. This augmented tree pair is a representative of the identity in $F_{(G,\theta)}$ and thus the relations **R11** and **R12** are hold true in $F_{(G,\theta)}$.

Therefore, all the relations of $\mathcal{F}^{inf}_{(G,\theta)}$ are found in $F_{(G,\theta)}$ under the map ϕ. Thus, by von Dyck's theorem, the map ϕ is a surjective group map from $\mathcal{F}^{inf}_{(G,\theta)}$ to $F_{(G,\theta)}$. To prove Theorem 4.28 it suffices to prove that ϕ is also injective.

Injectivity of ϕ

To prove that ϕ is injective we will show that for every non-trivial element f in $\mathcal{F}^{inf}_{(G,\theta)}$ there exists a word f_w written in the generators of $\mathcal{F}^{inf}_{(G,\theta)}$ which is equivalent to f and such that $\phi(f_w)$ is a non-trivial product in the normal form of Theorem 4.27. Thus $\phi(f_w)$ cannot be the identity element and hence ϕ will have been proven to be injective.

Let $\mathcal{S}$ be a finite string constructed from the generators of $\mathcal{F}^{inf}_{(G,\theta)}$. Via a series of stages we will create an equivalent element in $\mathcal{F}^{inf}_{(G,\theta)}$ such that under ϕ the resulting product is in the normal form described in Lemma 4.27. In the first 10 stages we will use the group relations to rewrite $\mathcal{S}$ to be in either *pq* or *paq* form. In the final stage we will reduce the string that we produced in Stage 10 and show that its image under ϕ must satisfy the irreducibility conditions needed for uniqueness from Theorem 4.27. Note that we will abuse notation slightly and apply the descriptions "positive", "negative" and "augmenting" to any string of generators in $\mathcal{F}^{inf}_{(G,\theta)}$ that have such properties in $F_{(G,\theta)}$ under the map ϕ.

Stage 1:

Removing inverse generators of the form $S_{i,n}^{-1}$ and $T_{i,n}^{-1}$

Consider the symbol $S_{i,n}$ for some i in $\boldsymbol{N}$ and some n in $\mathbb{N}_1$. Define j to be the index in $\boldsymbol{N}$ such that $g_j = g_i^{-1}$. Then there exists a relation in **R7** of the form $S_{i,n}S_{j,n} = S_{1,n}$ which combines with the relation $S_{1,n} = 1$ from **R11** to produce $S_{i,n}^{-1} = S_{j,n}$. This technique for replacing inverse generators with regular generators works for any inverse generators of the form $S_{i,n}^{-1}$ and $T_{i,n}^{-1}$. Thus we replace every inverse generator of the form $S_{i,n}^{-1}$ or $T_{i,n}^{-1}$ in the string $\mathcal{S}$ with its equivalent non-inverted generator. We call the new string $\mathcal{S}_1$.

Stage 2:

Moving all the positive elements to the left of the string.

The next stage involves rewriting the string $\mathcal{S}_1$ to a string of the form $\mathcal{S}_2 := \bar{\mathcal{P}}\mathcal{X}$ where $\bar{\mathcal{P}}$ is a string of positive elements and $\mathcal{X}$ is a string composed from negative and augmenting elements.

The set of relations **R1** imply that for all k, n in $\mathbb{N}_0$ such that $k < n$,

$$Y_n^{-1}Y_k = Y_kY_{n+1}^{-1}, \qquad Y_k^{-1}Y_n = Y_{n+1}Y_k^{-1}.$$

These relations provide us the means by which to move positive generators to the left of negative generators.

Likewise relations **R2**, **R3**, **R4** and **R5** provide the following rules by which we can move positive elements to the left of augmenting elements,

$$\begin{aligned} S_{i,n}Y_k &= Y_kS_{i,n+1}, && (k < n-1) \\ Y^{-1}S_{i,n} &= S_{i,n}Y_k^{-1}, && (k \geq n) \\ S_{i,n}Y_{n-1} &= Y_{n-1}S_{i,n}S_{(i)\delta,n+1}, && \\ T_{i,n}Y_k &= Y_kT_{i,n+1}, && (k < n). \end{aligned}$$

From all the rules above we can move any positive element to the left of nearly every generator, negative or augmenting, in $\mathcal{F}^{inf}_{(G,\theta)}$. The only situation for which we don't have a rule yet is when we encounter the substring $T_{i,k}Y_n$ when $k \leq n$. In this situation we combine multiple relations from **R9** together to produce the relation

$$T_{i,k} = S_{i,k+1} \dots S_{\delta^{n-k}(i),n} T_{\delta^{n-k+1}(i),n+1},$$

and replace the substring $T_{i,k}Y_n$ with $(S_{i,k+1} \dots S_{\delta^{n-k}(i),n} T_{\delta^{n-k+1}(i),n+1})Y_n$.

We can then use the previously defined rules to move Y_n to the left of this product. This process of moving all positive elements of the form Y_n to the left of all the other elements in the string is a finite process and must terminate. Thus after Stage 2 we have produced the string $\mathcal{S}_2$.

Stage 3:

Rearranging the positive elements

Following the definition of a positive function in $F_{(G,\theta)}$, a positive string is one of the form $Y_0^{a_0} Y_1^{a_1} \dots Y_m^{a_m}$ for some powers a_i in $\mathbb{N}_0$ (note that a positive string is **not** simply a string of positive elements but a specially ordered one). Given a string of positive elements, the relations of $\mathcal{F}_{(G,\theta)}^{inf}$ provide the rules necessary to rearrange the elements and create a positive string. Specifically, the set of relations in **R1** provide the rules

$$Y_n Y_k = Y_k Y_{n+1},\ \forall k, n \in \mathbb{N}_0 \text{ such that } k < n,$$

which moves positive generators with lower depth numbers to the left of positive generators with higher depth numbers. We thus apply this rule to the substring $\bar{\mathcal{P}}$ in $\mathcal{S}_2$ to create the positive string $\mathcal{P}$. This process ends when there exists no substring $Y_k Y_n$ such that $k < n$, which must be a finite process given we have a finite number of positive elements. We call the new string that has been created $\mathcal{S}_3 := \mathcal{P}\mathcal{X}$.

Stage 4:

Moving all the negative elements to the right of the string

After the first three stages the original string $\mathcal{S}$ has been replaced by the string $\mathcal{S}_3 = \mathcal{P}\mathcal{X}$ where $\mathcal{P}$ is a positive string and $\mathcal{X}$ is a string of negative and augmenting elements. Stage 4 rewrites the substring $\mathcal{X}$ to create a new substring $\mathcal{A}_1\bar{\mathcal{Q}}$ where $\mathcal{A}_1$ is a string of augmenting elements and $\bar{\mathcal{Q}}$ is a string of negative elements. To create the new substring we require that all the augmenting elements in $\mathcal{X}$ are moved to the left of the all the negative elements. The relations **R2**, **R3**, **R4**, **R5** and **R9** provide the means for this to happen. The first four of these relations imply the following:

$$\begin{aligned} Y_k^{-1} S_{i,n} &= S_{i,n+1} Y_k^{-1}, && (k < n-1) \\ Y_k^{-1} S_{i,n} &= S_{i,n} Y_k^{-1}, && (k \geq n) \\ Y_{n-1}^{-1} S_{i,n} &= S_{i,n} S_{(i)\delta,n+1} Y_{n-1}^{-1}, \end{aligned}$$

$$Y_k^{-1}T_{i,n} = T_{i,n+1}Y_k^{-1}, \qquad (k < n).$$

The final substring to consider is $Y_n^{-1}T_{i,k}$, when $k \leq n$. By using the method similar to that found in Stage 2, we use the relation **R9** to replace the generator $T_{i,k}$ with

$$T_{i,k} = S_{i,k+1} \dots S_{\delta^{n-k}(i),n}T_{\delta^{n-k+1}(i),n+1}.$$

The string $Y_n^{-1}T_{i,k}$ thus becomes $Y_n^{-1}S_{i,k+1} \dots S_{\delta^{n-k}(i),n}T_{\delta^{n-k+1}(i),n+1}$ and the negative element Y_n^{-1} can now be moved to the right by making use of the four rewrite rules previously defined.

We now therefore create a new string $\mathcal{S}_4 := \mathcal{P}\mathcal{A}_1\bar{\mathcal{Q}}$. Note that no positive elements are produced in any of the rewrite rules above, thus we can now continue the process without having to return to stages 2 or 3.

Stage 5:

Rearranging the negative elements

A negative string has the form $Y_m^{-a_m} \dots Y_1^{-a_1}Y_0^{-a_0}$ for powers $a_i \in \mathbb{N}_0$. In Stage 5 we will rewrite the substring $\bar{\mathcal{Q}}$ to produce a negative string $\mathcal{Q}$. The set of relations in **R1** again provide the necessary rewrite rules;

$$Y_k^{-1}Y_n^{-1} = Y_{n+1}^{-1}Y_k^{-1}, \text{ for all } k, n \text{ in } \mathbb{N}_0 \text{ such that } k < n.$$

Using these rules, one can rewrite $\bar{\mathcal{Q}}$ to produce the string $\mathcal{Q}$ where the indexing of the negative elements in the string decreases from left to right, as is required of a negative string. Thus after Stage 5 we have produced the string $\mathcal{S}_5 := \mathcal{P}\mathcal{A}_1\mathcal{Q}$.

Stage 6:

Expanding the T-generators

Stages 6-10 concentrate on rewriting $\mathcal{A}_1$ as an **augmenting** string. If $\mathcal{A}_1$ is empty then there is no work to be done and we go straight to Stage 11.

We define an augmenting string to be a string of generators which is taken to an augmenting product under the map ϕ. As there are three types of augmenting product there are also three types of augmenting strings that correspond to each.

Type 1: $T_{i,n}$,

Type 2: $S_{i_1,k_1}S_{i_2,k_2} \cdots S_{i_m,k_m}$,

Type 3: $S_{j_1,k_1}S_{j_2,k_2} \cdots S_{j_m,k_m}T_{l,n}$, where $k_1 < k_2 < \ldots < k_m \leq n$, $(i_m)\delta \neq 1$ and if $k_m = n$ then $(j_m)\delta \neq l$. Under the map ϕ each type given above gets mapped to the corresponding type of augmenting product in $F_{(G,\theta)}$. After Stages 6-10 the string $\mathcal{A}_1$ will have been rewritten as one of the three types listed above.

In Stage 6 we deal exclusively with the T-generators that are found in $\mathcal{A}_1$. If there are no T-generators in the string then we go straight to Stage 7.

Suppose then that the substring $\mathcal{A}_1$ in $\mathcal{S}_5$ contains one or more generators of the form $T_{i,k}$ for a range of depth indexes k. In Stage 6 we do not concern ourselves with the number of T-generators and instead concentrate on their depth indexes with the intention of making them all equal. This will ultimately allow us to combine them all into one generator, but that step in the process is not reached until Stage 8.

The relation **R9** gives us the rule to rewrite any T-generator of the from $T_{i,n}$ with $S_{i,n+1}T_{(i)\delta,n+1}$. Therefore by using this relation multiple times on each T-generator in $\mathcal{A}_1$ we can increase the depth index on all these generators to one common number m. We choose m to be the smallest natural number that is greater than or equal to the depth indexes of all S-generators in $\mathcal{A}_1$, should they exist. This is required for all augmenting strings of Type 3.

Thus we now apply the rewrite rules supplied by **R9** to create a string in which every T-generator has depth index m. Note that although our use of the relation **R9** also adds extra S-generators to the string none of these have a depth number greater than m. Therefore, the three conditions by which we chose the number m will hold throughout the entirety of this stage. Once every T-generator in the newly written string has depth index m we end the rewriting process and call the resulting substring $\mathcal{A}_2$.

Thus, at the end of Stage 6 we have produced the string $\mathcal{S}_6 = \mathcal{P}\mathcal{A}_2\mathcal{Q}$. Note that if $\mathcal{A}_1$ contained no T-generators then $\mathcal{A}_2 = \mathcal{A}_1$.

Stage 7:

Reordering the generators in the substring $\mathcal{A}_2$

In this stage we will be ordering the S and T generators in $\mathcal{A}_2$ relative to each other. If $\mathcal{A}_2$ has no S-generators then we can go straight to Stage 8.

If $\mathcal{A}_2$ contains both S-generators and T-generators then we require every S-generator to be to the left of each T-generator in anticipation of obtaining an augmenting string of Type 3. This is achieved by using the relations in **R10**, which we recall as those of the form $S_{i,k}T_{j,n} = T_{j,n}S_{i_k}$ for all i, j in $\boldsymbol{N}$ and k, n in $\mathbb{N}_1$ such that $k \leq n$. Due to the work done in Stage 6, the depth index of all the T-generators is greater than or equal to every depth index of the S-generators and thus we satisfy the conditions required to use **R10** to reorder $\mathcal{A}_2$ and move all the T-generators to the right.

For augmenting strings of Type 2 and Type 3 we also require that the S-generators are ordered from left to right according the their depth indexes, with the largest being at the right. We can achieve this ordering in $\mathcal{A}_2$ by making use of **R6** which states that all S-generators of different depth indexes commute.

We apply these two reordering processes to the string A_2 where necessary and call the result $\mathcal{A}_3$. Thus at the end of Stage 7 we have produced the string $\mathcal{S}_7 = \mathcal{P}\mathcal{A}_3\mathcal{Q}$.

Stage 8:

Combining generators in $\mathcal{A}_3$

The substring $\mathcal{A}_3$ may contain substrings of the form $T_{i,n}T_{j,n}$ or $S_{q,k}S_{r,k}$, where the generators have identical depth indexes. We can use the relations **R7** and **R8** to rewrite these substrings as single elements. Eventually this process will terminate and for each depth index there will eventually exist at most one S-generator or T-generator. We denote the substring that remains after this process $\mathcal{A}_4$ and the overall string $\mathcal{S}_8 := \mathcal{P}\mathcal{A}_4\mathcal{Q}$.

Stage 9:

Removing identity S and T generators

The generators $S_{1,n}$ and $T_{1,n}$ are taken to the identity under the map ϕ, thus need to be removed from the string. The relations **R11** and **R12** give the means with which to do this. After removing these extraneous generators the substring $\mathcal{A}_4$ has now become $\mathcal{A}_5$. If $\mathcal{A}_5$ is empty then we go straight to Stage 11, else we carry on to Stage 10. We call the resulting string from this stage $\mathcal{S}_9 := \mathcal{P}\mathcal{A}_4\mathcal{Q}$.

Stage 10:

Satisfying the conditions necessary for augmenting strings

The substring $\mathcal{A}_5$ in now in one of three possible forms

Form 1: $T_{i,n}$,

Form 2: $S_{i_1,k_1}S_{i_2,k_2}\cdots S_{i_m,k_m}$,

Form 3: $S_{j_1,k_1}S_{j_2,k_2}\cdots S_{j_m,k_m}T_{l,n}$,

where $k_1 < k_2 < \ldots < k_m \leq n$. However, it still may fail to be an augmenting string if it fails one of the two final conditions, namely if either

1. $\mathcal{A}_4$ is in Form 2 and $(i_m)\delta = 1$ or,

2. $\mathcal{A}_4$ is in Form 3, $k_m = n$ and $(j_m)\delta = l$.

We will consider each possibility in turn.

Case 1: $\mathcal{A}_5$ is in Form 2 and $(i_m)\delta = 1$

Suppose that $\mathcal{A}_5 = S_{i_1,k_1}S_{i_2,k_2}\cdots S_{i_m,k_m}$ and satisfies the conditions in Case 1. Then by **R12** we can add the generator T_{1,k_m} to the end of the $\mathcal{A}_5$ and produce the substring $S_{i_m,k_m}T_{(i_m)\delta,k_m}$. By **R9** we can then reduce this substring to T_{i_m,k_m-1} and the offending generator S_{i_m,k_m} is removed. In $F_{(G,\theta)}$ this is the equivalent to shrinking an exposed caret. The new string will now need to be checked to see if it now fails the second condition above.

Case 2: $\mathcal{A}_5$ is in Form 3, $k_m = n$ and $(j_m)\delta = l$

Suppose that $\mathcal{A}_5 = S_{j_1,k_1}S_{j_2,k_2}\cdots S_{j_m,k_m}T_{l,n}$ and satisfies the conditions in Case 2. Then the substring $S_{j_m,k_m}T_{l,n}$ is the same as $S_{j_m,k_m}T_{(j_m)\delta,k_m}$ which can be rewritten as the single generator T_{j_m,k_m-1} by **R9**. As in the previous case this is the equivalent to shrinking an exposed caret in $F_{(G,\theta)}$. Once the offending substring $S_{j_m,k_m}T_{l,n}$ has been replaced the new string must be checked again to see if the new generator T_{j_m,k_m-1} creates the same issue. If it does then the process is repeated again until a string is produced that either satisfies the condition or becomes an augmenting string of Type 1.

At the end of Stage 10 the substring $\mathcal{A}_5$ has now been rewritten as an augmenting string $\mathcal{A}$. Overall we leave the stage with the string $\mathcal{S}_{10} = \mathcal{PAQ}$.

Stage 11:

Creating irreducibility

The current string $\mathcal{S}_{10}$ will get taken to either a pq or a paq product under the map ϕ, however these products may not be in the normal form defined by Theorem 4.27. We will now consider all the forms that $\mathcal{S}_{10}$ could take such that $\phi(\mathcal{S}_{10})$ will not satisfy any of the three conditions given in Theorem 4.27.

Conditions 1 and 2 will fail if and only if $\mathcal{S}_{10}$ contains Y_k and Y_k^{-1} for some k but does not contain Y_{k+1} and Y_{k+1}^{-1}. Additionally, there are three different cases in which $\mathcal{S}_{10}$ will fail condition 3. We will consider each possible failure in turn and use the relations from the group to correct them when they occur.

Failure 1: $\mathcal{A}$ contains S generators or T generators with depth index $k+1$ and $k+2$.

To reiterate, in each of these three failures $\mathcal{S}_{10}$ contains Y_k and Y_k^{-1} for some k but does not contain Y_{k+1} and Y_{k+1}^{-1}, and thus $\mathcal{S}_{10}$ already fails conditions 1 and 2. Failure 1 occurs when $\mathcal{A}$ is empty or simply does not contain generators at the correct depth indexes. The method of dealing with this case remains the same in both scenarios. We begin by inserting the subword $(Y_k^{-1}Y_k)$ between each intervening generator in $\mathcal{S}_{10}$ that lies between Y_k and Y_k^{-1}. This creates a series of conjugations by the inverse element Y_k^{-1}. These conjugations can then be reduced by using the group relations, a consequence of which is that the original generators Y_k and Y_k^{-1} are removed from the string. In the table below we give all the possible conjugations that could be created and their replacements under the group relations.

Conjugations	Group relation	Reduction
$Y_kY_nY_k^{-1}$ for some $n > k$	**R1**	Y_{n-1}
$Y_kS_{i,n}Y_k^{-1}$ for some $n < k$	**R3**	$S_{i,n}$
$Y_kS_{i,n}Y_k^{-1}$ for some $n > k+1$	**R2**	$S_{i,n-1}$
$Y_kT_{i,n}Y_k^{-1}$ for some $n \geq k$	**R5**	$T_{i,n-1}$

Table 4.1

After reducing all the conjugations we are left with a string where the number of generators Y_k and Y_k^{-1} it contains have both been reduced by one and the length of the string has been reduced by two. If any generators of the form Y_k and Y_k^{-1} remain we repeat the process again until all such generators have been removed.

Failure 2: $\mathcal{A}$ contains $S_{i,k+1}S_{(i)\delta,k+2}$.

In dealing with the next two types of failure we employ the same method as we did above, namely insert copies of $(Y_k^{-1}Y_k)$ and then reduce the conjugations that are produced by using the group relations. The one exception in this case is the subword $S_{i,k+1}S_{(i)\delta,k+2}$. Here we do **not** insert the product $(Y_k^{-1}Y_k)$ between the two generators as the subword $Y_k^{-1}S_{i,k+1}S_{(i)\delta,k+2}Y_k$ is reduced to $S_{i,k+1}$ by **R4**. In all other respects the process is the same as the previous case and the details are given in Table 4.1. As a result the generators Y_k and Y_k^{-1} are removed and the substring $S_{i,k+1}S_{(i)\delta,k+2}$ has been replaced by $S_{i,k+1}$. We must now check our new string for any of the three failures and repeat the necessary steps if another occurs.

Failure 3: $\mathcal{S}_{10}$ contains $S_{i,k+1}W$ for some generator $W \neq S_{j,k+2}$ or $T_{j,k+1}$ for any j, and $(i)\delta = 1$.

By employing the same method as the two above we reduce the string by removing the generators Y_k and Y_k^{-1} and adjusting the intervening generators by the rules in Table 4.1. Again we must now check our new string for any more failures and repeat the necessary steps if another occurs.

In each of the three cases above we reduce the length of the string by at least two. Thus these processes must end at some point. This will occur either when the string no longer fails in any of the areas given above, or the string has been reduced to the empty string. In the case that we come to the end of Stage 11 and the string is non-empty then its image under ϕ must be in the normal form of Theorem 4.27. As any element in the normal form is necessarily non-trivial ϕ must be injective and hence our proof is complete. $\square$

4.4 A finite presentation for $F_{(G,\theta)}$

In this section we will prove Theorem 4.3 from the introduction, and produce a finite presentation for $F_{(G,\theta)}$. To do this we will make use of the well known **Tietze transformations** introduced in 1908 by Heinrich Franz Friedrich Tietze. Group presentations that differ only by a Tietze transformation define isomorphic groups. Throughout the rest of the paper we will reference the four transformations by using the notation **T1, T2, T3** and **T4**.

To understand the Tietze transformations we first introduce some terminology. Let G be a group given by the presentation $\langle X|R\rangle$, where R is a set of relations (rather than relators). That is, R is a set of equalities between finite strings in $(X^{\pm})^*$. We say that a set S is a **consequence** of the relations in R if every $s \in S$ can be derived from the relations in R.

T1. Adding a set of relations. Let $\langle X|R\rangle$ be a group presentation. Let S be another, possibly infinite, system of relations such that every $s \in S$ is a consequence of the relations in R. Then under **T1** we can create the presentation $\langle X|R \cup S\rangle$.

T2. Deleting a set of relations. Let $\langle X|R\rangle$ be a group presentation. Suppose a subset $S \subset R$ is a consequence of the relations in $R \setminus S$. Then under **T2** we can create the presentation $\langle X|R \setminus S\rangle$. This is the inverse process of **T1**.

T3. Adding a set of generators. Let $\langle X|R\rangle$ be a group presentation. Let Y be a set of symbols disjoint from X and $\{w_y|y \in Y\}$ a set of words over X. Define the set of relations $R_y := \{y(w_y)^{-1} = 1|y \in Y\}$. Then under **T3** we can create the presentation $\langle X \cup Y|R \cup R_y\rangle$.

T4. Deleting a set of generators. Let $\langle X|R\rangle$ be a group presentation isomorphic to a group G. Suppose Y is a subset of X and $R_y := \{y(w_y)^{-1} = 1|y \in Y\}$ is a subset of R where every w_y is a word over $X \setminus Y$. Define a new set R' which consists of all the relations in $R \setminus R_y$ translated such that for all $r \in R \setminus R_y$ and $y, y^{-1} \in Y$ every occurrence of y in r is replaced by w_y and every occurrence of y^{-1} in r is replaced by $(w_y)^{-1}$. Then under the transformation **T4** we can create the presentation $\langle X \setminus Y | R' \rangle$.

Using Tietze transformations, we will take the infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$ and produce the finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$ as given in Theorem 4.3. We will break the process into two parts.

In the first part, we transform $\mathcal{F}^{inf}_{(G,\theta)}$ into the finite presentation $F_{fin} = \langle X_f | R_f \rangle$ given below.

Set of generators X_f: $\{Y_0, Y_1\} \cup \{S_{i,1}, S_{i,2} |$ for all i in $\boldsymbol{N}\} \cup \{T_{i,0}, T_{i,1} |$ for all i in $\boldsymbol{N}\}$.

Set of relations R_f:

R1A $[Y_0 Y_1^{-1}, Y_0^{-1} Y_1 Y_0] = 1,$

R1B $[Y_0 Y_1^{-1}, Y_0^{-2} Y_1 Y_0^2] = 1,$

R2A $[Y_0 Y_1^{-1}, Y_0^{-1} S_{i,2} Y_0] = 1,$ for all i in $\boldsymbol{N}$,

R2B $[Y_0 Y_1^{-1}, Y_0^{-2} S_{i,2} Y_0^2] = 1,$ for all i in $\boldsymbol{N}$,

R3A $[Y_1, S_{i,1}] = 1,$ for all i in $\boldsymbol{N}$,

R3B $[Y_0^{-1} Y_1 Y_0, S_{i,1}] = 1,$ for all i in $\boldsymbol{N}$,

R3C $[Y_0^{-1} Y_1 Y_0, S_{i,2}] = 1,$ for all i in $\boldsymbol{N}$,

R3D $[Y_0^{-2} Y_1 Y_0^2, S_{i,2}] = 1,$ for all i in $\boldsymbol{N}$,

R4A $[Y_0, S_{i,1}^{-1}] = S_{(i)\delta,2},$ for all i in $\boldsymbol{N}$,

R4B $[Y_1, S_{i,2}^{-1}] = Y_0^{-1} S_{(i)\delta,2} Y_0,$ for all i in $\boldsymbol{N}$,

R5A $[Y_0 Y_1^{-1}, Y_0^{-1} T_{i,1} Y_0] = 1,$ for all i in $\boldsymbol{N}$,

R5B $[Y_0 Y_1^{-1}, Y_0^{-2} T_{i,1} Y_0^2] = 1,$ for all i in $\boldsymbol{N}$,

R6A $[S_{i,2}, T^{-1}_{(j)\delta,1} T_{j,0}] = 1,$ for all i and j in $\boldsymbol{N}$,

R6B $[Y_0^{-1} S_{i,2} Y_0, S_{j,1}] = 1,$ for all i and j in $\boldsymbol{N}$,

R6C $[Y_0^{-1} S_{i,2} Y_0, S_{j,2}] = 1,$ for all i and j in $\boldsymbol{N}$,

R6D $[Y_0^{-2} S_{i,2} Y_0^2, S_{j,2}] = 1,$ for all i and j in $\boldsymbol{N}$,

R7A $S_{i,1} S_{j,1} = S_{(i,j)\Delta,1},$ for all i and j in $\boldsymbol{N}$,

R7B $S_{i,2} S_{j,2} = S_{(i,j)\Delta,2},$ for all i and j in $\boldsymbol{N}$,

R8A $T_{i,0} T_{j,0} = T_{(i,j)\Delta,0},$ for all i and j in $\boldsymbol{N}$,

R8B $T_{i,1} T_{j,1} = T_{(i,j)\Delta,1},$ for all i and j in $\boldsymbol{N}$,

R9A $T_{i,0} = T_{(i)\delta,1} S_{i,1},$ for all i and j in $\boldsymbol{N}$,

R9B $T_{i,1} = Y_0^{-1} T_{(i)\delta,1} Y_0 S_{i,2},$ for all i and j in $\boldsymbol{N}$,

R10A $[S_{i,1}, T_{j,1}] = 1,$ for all i and j in $\boldsymbol{N}$,

R10B $[S_{i,1}, Y_0^{-1} T_{j,1} Y_0] = 1,$ for all i and j in $\boldsymbol{N}$,

R10C $[S_{i,2}, Y_0^{-1}T_{j,1}Y_0] = 1,$ for all i and j in $\boldsymbol{N}$,

R10D $[S_{i,2}, Y_0^{-2}T_{j,1}Y_0^2] = 1,$ for all i and j in $\boldsymbol{N}$,

R11A $S_{1,1} = 1,$

R11B $S_{1,2} = 1,$

R12A $T_{1,0} = 1,$

R12B $T_{1,1} = 1.$

Note that we name the sets comprising R_f based on the original set of relations in $\mathcal{F}^{inf}_{(G,\theta)}$. For the second part, we will take the finite presentation F_{fin} and reduce the number of generators by using the relations **R9A** and **R9B** to remove the generators of the form $S_{i,1}$ and $S_{i,2}$. The presentation produced after this process will be $\mathcal{F}^{fin}_{(G,\theta)}$.

Proposition 4.29. *The presentations $\mathcal{F}^{inf}_{(G,\theta)} = \langle X|R\rangle$ and $F_{fin} = \langle X_f|R_f\rangle$ represent isomorphic groups.*

Proof. The form of the proof will take the following outline. In **Stage 1** we will derive the set of relations R_f from the set R and use the transformation **T1** to create a new presentation $\widehat{F_1} = \langle X|R \cup R_f\rangle$. In **Stage 2** we will show that every relation in the presentation $\widehat{F_1}$ can be derived from the finite set R_f and three specially chosen infinite sets which we will call **A, B** and **C**. Then, by using the transformation **T2** on $\widehat{F_1}$, we will create a new presentation $\widehat{F_2} := \langle X|R_f \cup \mathbf{A} \cup \mathbf{B} \cup \mathbf{C}\rangle$. Finally, in **Stage 3** we turn our attention to the set of generators X. We will show that the three infinite sets of relations, **A, B** and **C** satisfy the conditions needed to remove a set of generators by the transformation **T4**. Therefore, by applying **T4** to $\widehat{F_2}$ we can remove all but finitely many generators from $\widehat{F_2}$ as well as the relations from **A, B** and **C**. The presentation that will have been created after Stage 3 will be the finite presentation $F_{fin} = \langle X_f|R_f\rangle$.

Stage 1

As stated in the introduction above, the purpose of this first stage is to transform the presentation $\mathcal{F}^{inf}_{(G,\theta)} = \langle X|R\rangle$ into the presentation $\widehat{F_1} = \langle X|R \cup R_f\rangle$ by using the Tietze transformation **T1**. To do so we must first show that each of the relations in R_f can be derived from the relations in R. The set R_f is split up into 30 finite subsets where each one is numbered according to the original relation in R from which it will be derived.

The sets R1A and R1B

The first two subsets from R_f are **R1A** and **R1B**. Both consist of just a single relation each and both can be be derived from relations in the set **R1**.

$$Y_0^{-1}Y_2Y_0 = Y_1^{-1}Y_2Y_1 \qquad\qquad Y_0^{-1}Y_3Y_0 = Y_1^{-1}Y_3Y_1$$
$$\Rightarrow Y_1Y_0^{-1}Y_2Y_0Y_1^{-1}Y_2 = 1 \qquad\qquad \Rightarrow Y_1Y_0^{-1}Y_3Y_0Y_1^{-1}Y_3 = 1$$

$$\Rightarrow [Y_0Y_1^{-1}, Y_2] = 1 \qquad \Rightarrow [Y_0Y_1^{-1}, Y_3] = 1$$
$$\Rightarrow [Y_0Y_1^{-1}, Y_0^{-1}Y_1Y_0] = 1 \qquad \Rightarrow [Y_0Y_1^{-1}, Y_0^{-2}Y_1Y_0^2] = 1$$

The sets R2A and R2B

The relations in **R2A** and **R2B** can be constructed from those in **R2** in the following way;

$$\begin{array}{ll}
Y_0^{-1}S_{i,3}Y_0 = Y_1^{-1}S_{i,3}Y_1 & Y_0^{-1}S_{i,4}Y_0 = Y_1^{-1}S_{i,4}Y_1 \\
\Rightarrow Y_1Y_0^{-1}S_{i,3}^{-1}Y_0Y_1^{-1}S_{i,3} = 1 & \Rightarrow Y_1Y_0^{-1}S_{i,4}^{-1}Y_0Y_1^{-1}S_{i,4} = 1 \\
\Rightarrow [Y_0Y_1^{-1}, S_{i,3}] = 1 & \Rightarrow [Y_0Y_1^{-1}, S_{i,4}] = 1 \\
\Rightarrow [Y_0Y_1^{-1}, Y_0^{-1}S_{i,2}Y_0] = 1 & \Rightarrow [Y_0Y_1^{-1}, Y_0^{-2}S_{i,2}Y_0^2] = 1.
\end{array}$$

The sets R3A, R3B, R3C and R3D

The set of relations **R3** can be written as the set of commutators

$$\{[Y_k, S_{i,n}] = 1| \text{ for all } i \text{ in } \boldsymbol{N},\ n, k \in \mathbb{N}_1 \text{ such that } 0 < n \leq k\}.$$

Thus **R3A** is simply a subset of **R3**. The set **R3B** is very close to the subset $\{[Y_2, S_{i,1}] = 1|$ for all i in $\boldsymbol{N}\}$ from **R3**. We use the relation $Y_2 = Y_0^{-1}Y_1Y_0$ from **R1** to replace every occurance of Y_2 and thus produce **R3B**. A similar process produces **R3C** and **R3D**.

The sets R4A and R4B

The set **R4A** is simply a subset of **R4**. The set **R4B** is derived from the subset $\{[Y_2, S_{i,2}^{-1}] = S_{i,3}|$ for all i in $\boldsymbol{N}\}$ of **R4**, where every $S_{i,3}$ is replaced with $Y_0^{-1}S_{(i)\delta,2}Y_0$ by the relation from **R2**.

The sets R5A and R5B

The sets **R5A** and **R5B** are derived from **R5** in the same manner that **R2A** and **R2B** were derived from **R2**.

$$\begin{array}{ll}
Y_0^{-1}T_{i,2}Y_0 = Y_1^{-1}T_{i,2}Y_1 & Y_0^{-1}T_{i,3}Y_0 = Y_1^{-1}T_{i,3}Y_1 \\
\Rightarrow Y_1Y_0^{-1}T_{i,2}^{-1}Y_0Y_1^{-1}T_{i,2} = 1 & \Rightarrow Y_1Y_0^{-1}T_{i,3}^{-1}Y_0Y_1^{-1}T_{i,3} = 1 \\
\Rightarrow [Y_0Y_1^{-1}, T_{i,2}] = 1 & \Rightarrow [Y_0Y_1^{-1}, T_{i,3}] = 1 \\
\Rightarrow [Y_0Y_1^{-1}, Y_0^{-1}T_{i,1}Y_0] = 1 & \Rightarrow [Y_0Y_1^{-1}, Y_0^{-2}T_{i,1}Y_0^2] = 1.
\end{array}$$

The sets R6A, R6B, R6C and R6D

The set of relations **R6** can be written as the set of commutators

$$\{[S_{i,m}, S_{j,n}] = 1| \text{ for all } i \text{ in } \boldsymbol{N},\ n, m \in \mathbb{N}_1 \text{ such that } 0 < m \leq n\}.$$

Thus **R6A** is simply a subset of **R6**. The set **R6B** is derived from the subset $\{[S_{i,3}, S_{j,1}] = 1|$ for all i in $\boldsymbol{N}\}$ of **R6**. We use the relation $S_{i,3} = Y_0^{-1} S_{i,2} Y_0$ from **R2** to replace every occurance of $S_{i,3}$ and thus produce **R6B**. A similar process produces **R6C** and **R6D**.

The sets R7A, R7B, R8A and R8B

All the relations from the sets **R7A, R7B** and **R8A** are taken directly from **R7** and **R8** respectively.

The sets R9A and R9B

The set **R9A** is taken directly from **R9**. The set **R9B** is derived from from the subset $\{T_{i,1} = T_{(i)\delta,2} S_{i,2}\}$ of **R9**. By using the relation $T_{(i)\delta,2} = Y_0^{-1} T_{(i)\delta,1} Y_0$ from **R5** we can replace every occurrence of $T_{(i)\delta,2}$ with $Y_0^{-1} T_{(i)\delta,1} Y_0$ and thus produce **R9B**.

The sets R10A, R10B, R10C and R10D

The set of relations **R10** can be written as the set of commutators

$$\{[S_{i,m}, T_{j,n}] = 1| \text{ for all } i \text{ in } \boldsymbol{N},\ n, m \in \mathbb{N}_1 \text{ such that } 0 < m \leq n\}.$$

Thus **R10A** is simply a subset of **R10**. The set **R10B** is derived from to the subset $\{[S_{i,1}, T_{j,2}] = 1|$ for all i in $\boldsymbol{N}\}$ of **R10**. We use the relation $T_{i,2} = Y_0^{-1} T_{i,1} Y_0$ from **R2** to replace every occurance of $T_{i,2}$ with $, T_{j,2}] = 1|$ for all i in $\boldsymbol{N}\}$ of **R10**. We use the relation $T_{i,2} =$ and thus produce **R10B**. A similar process produces **R10C** and **R10D**.

The sets R11A, R11B, R12A and R12B

All the relations from the sets **R11A, R11B, R12A** and **R12B** are taken directly from **R11** and **R12** respectively.

Having established that all the relations in $\mathcal{F}^{fin}_{(G,\theta)}$ can either be derived from or taken directly from the relations in $\widehat{F}$, we then use the Tietze transformation **T1** to add the new relations to $\widehat{F}$ and create a new presentation. We call this presentation $\widehat{F}_1 := \langle X | R \cup R_f \rangle$.

Stage 2

The second stage of the proof involves removing a large number of relations from the presentation $\widehat{F}_1$ by making use of the Tietze transformation **T2**. By the end of this stage we will have constructed a presentation that has the same generating set X but whose relations have been reduced to the finite set R_f and three distinct infinite sets which will be defined throughout the stage. In the process that follows we consider each set of relations from R in turn, these will in turn provide sub-stages for Stage 2.

***The set of relations* R1**

First consider the set $\mathbf{R1} = \{Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1 | \text{for all } k \text{ and } n \text{ in } \mathbb{N}_0 \text{ such that } k < n\}$.

We split this relation into two disjoint subsets **R1a** and **R1b** which are defined as

$$\begin{aligned}\mathbf{R1a} :=& \{Y_0^{-1}Y_n^{-1}Y_0Y_{n+1} = 1| \text{ for all } n \geq 1\},\\ \mathbf{R1b} :=& \mathbf{R1} \setminus \mathbf{R1a} = \{Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1| \text{ for all } 0 < k < n\}.\end{aligned}$$

We will first show that every relation $Y_0^{-1}Y_nY_0 = Y_{n+1}$ from the set **R1a** can be rewritten as $Y_0^{-n}Y_1Y_0^n = Y_{n+1}$ for some n. The proof proceeds by induction on n. For the case when $n = 1$ the relation $Y_2 = Y_0^{-1}Y_1Y_0$ automatically satisfies the claim. Now suppose that the relation $Y_0^{-1}Y_nY_0 = Y_{n+1}$ can be written as $Y_0^{-n}Y_1Y_0^n = Y_{n+1}$ for some $n \geq 1$. Consider the relation $Y_0^{-1}Y_{n+1}Y_0 = Y_{n+2}$. By our assumption this becomes $Y_{n+2} = Y_0^{-1}(Y_0^{-n}Y_1Y_0^n)Y_0$, from which follows the relation $Y_{n+2} = Y_0^{-(n+1)}Y_1Y_0^{(n+1)}$. Thus, by induction, the claim is true. We can therefore use the Tietze transformations **T1** and **T2** to replace every relation from the set **R1a** that is of the form $Y_{n+1} = Y_0^{-1}Y_nY_0^{-1}$ with the relation $Y_{n+1} = Y_0^{-n}Y_1Y_0^n$. First the transformation **T1** adds all relations of the form $Y_{n+2} = Y_0^{-(n+1)}Y_1Y_0^{(n+1)}$ to the presentation. Following this **T2** then removes all the relations $Y_{n+1} = Y_0^{-1}Y_nY_0^{-1}$. The new set of relations that has replaced **R1a** we call **A**;

$$\mathbf{A} := \{Y_0^{-n}Y_1Y_0^n = Y_{n+1}| \text{ for all } n \geq 1\}.$$

This set **A** is the first of the infinite sets mentioned in the introduction to the second stage of the proof. Every relation in **A** associates a generator Y_n with a finite string constructed from the set $\{Y_0, Y_0^{-1}, Y_1\}$ and **A** will be used in Stage 3, in conjuction with the transformation **T4**, to remove from the presentation the generators Y_n for all $n \geq 2$.

We will now prove that the infinite set of relations **R1b** can be derived from **A** together with the two relations **R1A** and **R1B**. Consider the relation $Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1$ from **R1b**. Using the relations from **A** this can be rewritten in the following way;

$$\begin{aligned}&Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1\\ &\quad\Leftrightarrow (Y_0^{-(k-1)}Y_1^{-1}Y_0^{(k-1)})(Y_0^{-(n-1)}Y_1^{-1}Y_0^{(n-1)})(Y_0^{-(k-1)}Y_1Y_0^{(k-1)})(Y_0^{-n}Y_1Y_0^n) = 1\\ &\quad\Leftrightarrow Y_0^{-(k-1)}Y_1^{-1}Y_0^{(k-n)}Y_1^{-1}Y_0^{(n-k)}Y_1Y_0^{(k-n-1)}Y_1Y_0^n = 1\\ &\quad\Leftrightarrow Y_1^{-1}Y_0^{(k-n)}Y_1^{-1}Y_0^{(n-k)}Y_1Y_0^{(k-n-1)}Y_1Y_0^{(n-k+1)} = 1\\ &\quad\Leftrightarrow Y_1^{-1}Y_0Y_0^{(k-n-1)}Y_1^{-1}Y_0^{(n-k+1)}Y_0^{-1}Y_1Y_0^{(k-n-1)}Y_1Y_0^{(n-k+1)} = 1\\ &\quad\Leftrightarrow Y_1^{-1}Y_0Y_{n-k+2}^{-1}Y_0^{-1}Y_1Y_{n-k+2} = 1\\ &\quad\Leftrightarrow [Y_0^{-1}Y_1, Y_{n-k+2}] = 1\end{aligned}$$

$$\Leftrightarrow [Y_1^{-1}Y_0, Y_{n-k+2}] = 1.$$

This gives us the important double implication

$$Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1 \Leftrightarrow [Y_1^{-1}Y_0, Y_{n-k+2}] = 1. \qquad (\star)$$

By $(\star)$, any result we prove for the relation $[Y_1^{-1}Y_0, Y_{n-k+2}] = 1$ will also be true for $Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1$. Therefore as $n > k$, by proving that for all $m \geq 3$ the relation $[Y_1^{-1}Y_0, Y_m] = 1$ can be derived from **A**, **R1A** and **R1B** we will have also proved that the relations in **R1b** can be derived from the same three sets. The proof proceeds by induction on m. Consider the initial case when $m = 3$,

$$\begin{aligned}
Y_1^{-1}Y_0Y_3 &= Y_1^{-1}Y_0Y_0^{-2}Y_1Y_0^2 && \text{(from } \mathbf{A}\text{)}\\
&= Y_1^{-1}Y_0^{-1}Y_1Y_0^2 \\
&= (Y_0^{-1}Y_0)Y_1^{-1}Y_0^{-1}Y_1Y_0^2 && \text{(multiplication by the identity)}\\
&= Y_0^{-1}(Y_0Y_1^{-1})(Y_0^{-1}Y_1Y_0)Y_0 \\
&= Y_0^{-1}(Y_0^{-1}Y_1Y_0)(Y_0Y_1^{-1})Y_0 && \text{(from } \mathbf{R1A}\text{)}\\
&= Y_0^{-2}Y_1Y_0^2Y_1^{-1}Y_0 \\
&= Y_3Y_1^{-1}Y_0 && \text{(from } \mathbf{A}\text{)}
\end{aligned}$$

as required.

Therefore the relation

$$[Y_1^{-1}Y_0, Y_3] = 1 \qquad (4.15)$$

can be derived from **A**, **R1A** and **R1B**.

Likewise for $m = 4$,

$$\begin{aligned}
Y_1^{-1}Y_0Y_4 &= Y_1^{-1}Y_0Y_0^{-3}Y_1Y_0^3 && \text{(from } \mathbf{A}\text{)}\\
&= Y_1^{-1}Y_0^{-2}Y_1Y_0^3 \\
&= (Y_0^{-1}Y_0)Y_1^{-1}Y_0^{-2}Y_1Y_0^3 && \text{(multiplication by the identity)}\\
&= Y_0^{-1}(Y_0Y_1^{-1})(Y_0^{-2}Y_1Y_0^2)Y_0 \\
&= Y_0^{-1}(Y_0^{-2}Y_1Y_0^2)(Y_0Y_1^{-1})Y_0 && \text{(from } \mathbf{R1B}\text{)}\\
&= Y_0^{-3}Y_1Y_0^3Y_1^{-1}Y_0 \\
&= Y_4Y_1^{-1}Y_0 && \text{(from } \mathbf{A}\text{)}
\end{aligned}$$

as required.

Recall that because the relation (4.15) satisfies the claim, by $(\star)$ it follows that for all

$n \geq 1$ the relation $Y_{n+1} = Y_{n-1}^{-1} Y_n Y_{n-1}$ can also be derived from **A**, **R1A** and **R1B**. This fact will be used in the inductive step below.

Suppose that $[Y_1^{-1}Y_0, Y_m] = 1$ and $[Y_1^{-1}Y_0, Y_{m+1}] = 1$ satisfy the claim for some $m \geq 3$. Consider the product $Y_1^{-1}Y_0Y_{m+2}$,

$$\begin{aligned} Y_1^{-1}Y_0Y_{m+2} &= Y_1^{-1}Y_0Y_m^{-1}Y_{m+1}Y_m && \text{(from the consequences of (4.15))} \\ &= Y_m^{-1}Y_{m+1}Y_mY_1^{-1}Y_0 && \text{(by our assumptions)} \\ &= Y_{m+2}Y_1^{-1}Y_0. && \text{(from the consequences of (4.15))} \end{aligned}$$

Thus the relation $[Y_1^{-1}Y_0, Y_{m+2}] = 1$ can also be derived from **A**, **R1A** and **R1B**. Therefore, by induction, the relation $[Y_1^{-1}Y_0, Y_m] = 1$ satisfies the claim for all $m \geq 3$. Hence, by $(\star)$, the same is true for all relations of the form $Y_k^{-1}Y_n^{-1}Y_kY_{n+1} = 1$ where $0 < k < n$ and thus we have proved the proposition regarding **R1b**.

***The set of relations* R2**

The second set of relations from the set R to consider is **R2**, which we recall as the set

$$\{Y_k^{-1}S_{i,n}^{-1}Y_kS_{i,n+1} = 1 | \text{for all } i \text{ in } \boldsymbol{N},\ k \text{ in } \mathbb{N}_0 \text{ and } n \text{ in } \mathbb{N}_2 \text{ such that } k < n-1\}.$$

We will show that this set can be derived from four finite sets together with two infinite sets, where the outline of the proof will be similar to the **R1** case. To begin with we partition **R2** into two disjoint subsets,

$$\begin{aligned} \mathbf{R2a} &:= \{Y_0^{-1}S_{i,n}Y_0 = S_{i,n+1} | \text{ for all } i \in \boldsymbol{N} \text{ and } n \geq 2\} \\ \mathbf{R2b} := \mathbf{R2} \setminus \mathbf{R2a} &= \{Y_k^{-1}S_{i,n}Y_k = S_{i,n+1} | \text{ for all } i \in \boldsymbol{N} \text{ and } 0 < k < n-1\}. \end{aligned}$$

A simple induction argument, similiar to the derivation of **A**, proves that we can replace any relation $Y_0^{-1}S_{i,n}Y_0 = S_{i,n+1}$ in **R2a** with the relation $Y_0^{-(n-2)}S_{i,2}Y_0^{(n-2)} = S_{i,n}$. Thus we use the transformations **T1** and **T2** to replace **R2a** with the set

$$\mathbf{B} := \{Y_0^{-(n-2)}S_{i,2}Y_0^{(n-2)} = S_{i,n} | \text{for all } i \in \boldsymbol{N} \text{ and } n \geq 2\}.$$

This set **B** is the second of the three infinite sets mentioned in the introduction to Stage 2.

Our aim is to show that the relations from the set **R2b** can be derived from **R1A, R1B, R2A, R2B** together with the two infinite sets **A** and **B**. We can use the relations from **B** and the relation $Y_0^{-(k-1)}Y_1Y_0^{(k-1)} = Y_k$ from **A** to rewrite each $Y_k^{-1}S_{i,n}Y_k = S_{i,n+1}$

from **R2b** as follows;

$$
\begin{aligned}
&Y_k^{-1}S_{i,n}Y_k = S_{i,n+1}\\
&\Leftrightarrow (Y_0^{-(k-1)}Y_1^{-1}Y_0^{(k-1)})(Y_0^{-(n-2)}S_{i,2}^{-1}Y_0^{(n-2)})(Y_0^{-(k-1)}Y_1Y_0^{(k-1)})(Y_0^{-(n-1)}S_{i,2}Y_0^{(n-1)}) = 1\\
&\Leftrightarrow Y_1^{-1}Y_0^{(k-n+1)}S_{i,2}^{-1}Y_0^{(n-k-1)}Y_1Y_0^{(k-n)}S_{i,2}Y_0^{(n-k)} = 1\\
&\Leftrightarrow Y_1^{-1}Y_0Y_0^{(k-n)}S_{i,2}^{-1}Y_0^{(n-k)}Y_0^{-1}Y_1Y_0^{(k-n)}S_{i,2}Y_0^{(n-k)} = 1\\
&\Leftrightarrow [Y_0^{-1}Y_1, Y_0^{(k-n)}S_{i,2}Y_0^{(n-k)}] = 1\\
&\Leftrightarrow [Y_0^{-1}Y_1, S_{i,n-k+2}] = 1\\
&\Leftrightarrow [Y_1^{-1}Y_0, S_{i,n-k+2}] = 1.
\end{aligned}
$$

This gives us another important double implication

$$Y_k^{-1}S_{i,n}^{-1}Y_kS_{i,n+1} = 1 \Leftrightarrow [Y_1^{-1}Y_0, S_{i,n-k+2}] = 1. \tag{$\star\star$}$$

Therefore, by ($\star\star$), if we can prove that for all $m \geq 4$ the relations $[Y_1^{-1}Y_0, S_{i,m}] = 1$ can be derived from the relations **R1A, R1B, R2A, R2B** and the two infinite sets **A** and **B**, then we will have proved the same claim for the relations of the form $Y_k^{-1}S_{i,n}Y_k = S_{i,n+1}$ for all $0 < k < n-1$. Consider the case when $m = 4$. Using **B** and **R2A** we can derive $[Y_1^{-1}Y_0, S_{i,4}] = 1$ in the following way;

$$
\begin{aligned}
Y_1^{-1}Y_0S_{i,4} &= Y_1^{-1}Y_0Y_0^{-2}S_{i,2}Y_0^2 && \text{(from } \mathbf{B}\text{)}\\
&= Y_1^{-1}Y_0^{-1}S_{i,2}Y_0Y_0\\
&= (Y_0^{-1}Y_0)Y_1^{-1}Y_0^{-1}S_{i,2}Y_0Y_0 && \text{(insertion of the identity)}\\
&= Y_0^{-1}(Y_0^{-1}S_{i,2}Y_0)(Y_0Y_1^{-1})Y_0 && \text{(by } \mathbf{R2A}\text{)}\\
&= Y_0^{-2}S_{i,2}Y_0^2Y_1^{-1}Y_0\\
&= S_{i,4}Y_1^{-1}Y_0 && \text{(from } \mathbf{B}\text{).}
\end{aligned}
$$

Therefore by using the relations **B** and **R2A** one can derive the relation

$$[Y_1^{-1}Y_0, S_{i,4}] = 1. \tag{4.16}$$

Recall that as the relation (4.16) satisfies the claims, by ($\star\star$) every relation of the form $Y_{n-2}^{-1}S_{i,n}^{-1}Y_{n-2}S_{i,n+1} = 1$ can also be derived from the relations **B** and **R2A**.

The cases when $m \geq 5$ are proved by induction. In the proof we will make use of the fact that the relations $[Y_1^{-1}Y_0, Y_m]$ have already been proven to be derivable from **A, R1A** and **R1B**, for any $m \geq 3$. First consider the initial case, $m = 5$.

$$Y_1^{-1}Y_0S_{i,5} = Y_1^{-1}Y_0Y_0^{-3}S_{i,2}Y_0^3 \qquad \text{(from } \mathbf{B}\text{)}$$

$$
\begin{aligned}
&= Y_1^{-1}Y_0^{-2}S_{i,2}Y_0^3 && \\
&= (Y_0^{-1}Y_0)Y_1^{-1}Y_0^{-2}S_{i,2}Y_0^3 && \text{(insertion of the identity)} \\
&= Y_0^{-1}(Y_0Y_1^{-1})(Y_0^{-2}S_{i,2}Y_0^2)Y_0 && \\
&= Y_0^{-1}(Y_0^{-2}S_{i,2}Y_0^2)(Y_0Y_1^{-1})Y_0 && \text{(by } \mathbf{R2B}\text{)} \\
&= Y_0^{-3}S_{i,2}Y_0^3Y_1^{-1}Y_0 && \\
&= S_{i,5}Y_1^{-1}Y_0. && \text{(from } \mathbf{B}\text{)}
\end{aligned}
$$

Thus the relation $[Y_1^{-1}Y_0, S_{i,5}] = 1$ can be derived from **B** and **R2B**.

Suppose $[Y_1^{-1}Y_0, S_{i,m}] = 1$ satisfies the claim for some $m \geq 5$. Then consider the product $Y_1^{-1}Y_0, S_{i,m+1}$;

$$
\begin{aligned}
Y_1^{-1}Y_0S_{i,m+1} &= (Y_1^{-1}Y_0)Y_{m-2}^{-1}S_{i,m}Y_{m-2} && \text{(from the consequence of (4.16))} \\
&= Y_{m-2}^{-1}(Y_1^{-1}Y_0)S_{i,m}Y_{m-2} && \text{(from } \mathbf{A, R1A} \text{ and } \mathbf{R1B}\text{)} \\
&= Y_{m-2}^{-1}S_{i,m}(Y_1^{-1}Y_0)Y_{m-2} && \text{(by our assumption)} \\
&= Y_{m-2}^{-1}S_{i,m}Y_{m-2}(Y_1^{-1}Y_0) && \text{(from } \mathbf{A, R1A} \text{ and } \mathbf{R1B}\text{)} \\
&= S_{i,m+1}Y_1^{-1}Y_0. && \text{(from the consequence of (4.16))}
\end{aligned}
$$

Thus, by induction, any relation of the form $[Y_1^{-1}Y_0, S_{i,m}] = 1$ for $m \geq 5$ can be derived using relations from the sets **R1A, R1B, R2A, R2B** and the infinite sets **A** and **B**. Therefore by ($\star\star$) we conclude that the claim is true for every relation in **R2b**.

***The set of relations* R3**

The next set of relations to consider is **R3**, which we recall as the set

$$\{[Y_kS_{i,n}] = 1 | \text{for all } i \text{ in } \boldsymbol{N},\ k \text{ in } \mathbb{N}_0 \text{ and } n \text{ in } \mathbb{N}_1 \text{ such that } k \geq n\}.$$

The aim of the next part of the proof is to show that that this set can be constructed from the six relations **R1A, R1B, R3A, R3B, R3C** and **R3D** and the two infinite sets **A** and **B**. To do so we consider the set **R3** as the union of two disjoint subsets, **R3a** and **R3b**, and consider each subset separately.

$$
\begin{aligned}
\mathbf{R3a} &:= \{[Y_k, S_{i,1}] = 1 | \text{ for all } i \text{ in } \boldsymbol{N},\ k \text{ in } \mathbb{N}_1\} \\
\mathbf{R3b} &:= \mathbf{R3} \setminus \mathbf{R3a} = \{[Y_k, S_{i,n}] = 1 | \text{ for all } i \text{ in } \boldsymbol{N},\ n, k \text{ in } \mathbb{N}_2 \text{ such that } n \leq k\}.
\end{aligned}
$$

We first prove the claim for **R3a** by induction on k. The case, when $k = 1$, is exactly the relation **R3A**. The second case, when $k = 2$, is the relation **R3B** where we have used the relation $Y_2 = Y_0^{-1}Y_1Y_0$ from **A** to replace the string $Y_0^{-1}Y_1Y_0$. Suppose that for some $k \geq 1$ the relations $[Y_k, S_{i,1}] = 1$ and $[Y_{k+1}, S_{i,1}] = 1$ can be constructed using relations from **R1A, R1B, R3A** and **R3B** together with the infinite set **A**. Consider the product $Y_{k+2}S_{i,1}$. Using relations from the five sets above we have the following;

$$\begin{aligned}
Y_{k+2}S_{i,1} &= (Y_k^{-1}Y_{k+1}Y_k)S_{i,1} && \text{(from } \mathbf{R1A,\ R1B} \text{ and } \mathbf{A}\text{)}\\
&= S_{i,1}Y_k^{-1}Y_{k+1}Y_k && \text{(by our assumptions)}\\
&= S_{i,1}Y_{k+2} && \text{(from } \mathbf{R1A,\ R1B} \text{ and } \mathbf{A}\text{).}
\end{aligned}$$

Thus, by induction, the relations $[Y_k, S_{i,1}] = 1$ for all $k \geq 1$ can be constructed from **R1A, R1B**, **R3A, R3B** and **A**.

This leaves us with the second subset **R3b** which we will show can be constructed from the four finite sets of relations **R1A, R1B, R3C** and **R3D** together with the two infinite sets **A** and **B**. Recall that the indexes k and n now must satisfy the inequality $2 \leq n \leq k$. Let $[Y_k, S_{i,n}] = 1$ be a relation from **R3b**. By using relations from **A** and **B** it can be rewritten in the following way;

$$\begin{aligned}
&[Y_k, S_{i,n}] = 1\\
&\Leftrightarrow Y_k^{-1}S_{i,n}^{-1}Y_kS_{i,n} = 1\\
&\Leftrightarrow (Y_0^{-(k-1)}Y_1^{-1}Y_0^{(k-1)})(Y_0^{-(n-2)}S_{i,2}^{-1}Y_0^{(n-2)})(Y_0^{-(k-1)}Y_1Y_0^{(k-1)})(Y_0^{-(n-2)}S_{i,2}Y_0^{(n-2)}) = 1\\
&\Leftrightarrow Y_0^{-(k-1)}Y_1^{-1}Y_0^{(k-n+1)}S_{i,2}^{-1}Y_0^{-(k-n+1)}Y_1Y_0^{(k-n+1)}S_{i,2}Y_0^{(n-2)} = 1\\
&\Leftrightarrow (Y_0^{-(k-n+1)}Y_1^{-1}Y_0^{(k-n+1)})S_{i,2}^{-1}(Y_0^{-(k-n+1)}Y_1Y_0^{(k-n+1)})S_{i,2} = 1\\
&\Leftrightarrow Y_{k-n+2}^{-1}S_{i,2}^{-1}Y_{k-n+2}S_{i,2} = 1\\
&\Leftrightarrow [Y_{k-n+2}, S_{i,2}] = 1.
\end{aligned}$$

Thus, by the double implication above, proving that the set of relations $\{[Y_m, S_{i,2}] = 1 | \text{for all } m \geq 2\}$ can be derived from **R1A, R1B, R3C, R3D** and **A** will also prove our original claim regarding **R3b**. The proof proceeds by induction on m. The two initial cases, when $m = 2$ and $m = 3$, are taken directly from the relations **R3C** and **R3D**. Suppose the relations $[Y_m, S_{i,2}] = 1$ and $[Y_{m+1}, S_{i,2}] = 1$ satisfy the claim for some $m \geq 2$. Consider the product $Y_{m+2}S_{i,2}$;

$$\begin{aligned}
Y_{m+2}S_{i,2} &= Y_m^{-1}Y_{m+1}Y_mS_{i,2} && \text{(by } \mathbf{R1A,\ R1B} \text{ and } \mathbf{A}\text{)}\\
&= S_{i,2}Y_m^{-1}Y_{m+1}Y_m && \text{(by our assumptions)}\\
&= S_{i,2}Y_{m+2} && \text{(by } \mathbf{R1A,\ R1B} \text{ and } \mathbf{A}\text{)}
\end{aligned}$$

as required.

Therefore the set of relations in **R3b** can be derived from the six relations **R1A, R1B, R3A, R3B, R3C** and **R3D** and the two infinite sets **A** and **B**.

***The set of relations* R4**

The set of relations **R4** can be written as $\{[Y_n, S_{i,n+1}^{-1}] = S_{(i)\delta,n+2} |$ for all n in $\mathbb{N}_0\}$. Our claim is that the two sets **R4A** and **R4B**, and the two infinite sets **A** and **B**, are enough to generate all the relations in **R4**. We split the relation into three cases. In the first case we consider the subset of relations for when $n = 0$. This is exactly the set **R4A** and thus automatically satisfies the claim. The second case is the subset of **R4** for when $n = 1$. Each relation of the form $[Y_1, S_{i,2}^{-1}] = S_{(i)\delta,3}$ can be derived from the relation $[Y_1, S_{i,2}^{-1}] = Y_0^{-2} S_{(i)\delta,2} Y_0$ from **R4B** and the relation $S_{(i)\delta,3} = Y_0^{-2} S_{(i)\delta,2} Y_0$ from **B**. Thus every relation in **R4** for when $n = 1$ is derived from **R4B** and **B**. Consider the product $Y_n S_{i,n+1}^{-1}$ for some $n \geq 2$. Using relations from the three sets **R4B, A** and **B** we have the following;

$$\begin{aligned}
Y_n S_{i,n+1}^{-1} =& Y_0^{-(n-1)} Y_1 Y_0^{n-1} Y_0^{-(n-1)} S_{i,2}^{-1} Y_0^{n-1} && \text{(from } \mathbf{A} \text{ and } \mathbf{B}\text{)}\\
=& Y_0^{-(n-1)} Y_1 S_{i,2}^{-1} Y_0^{n-1}\\
=& Y_0^{-(n-1)} S_{i,2}^{-1} Y_1 S_{(i)\delta,3} Y_0^{n-1} && \text{(from } n = 1 \text{ case)}\\
=& Y_0^{-(n-1)} S_{i,2}^{-1} (Y_0^{n-1} Y_0^{-(n-1)}) Y_1 (Y_0^{n-1} Y_0^{-(n-1)}) S_{(i)\delta,3} Y_0^{n-1}\\
& \text{(insertion of identity elements)}\\
=& S_{i,n+1}^{-1} Y_n S_{(i)\delta,n+2} && \text{(from } \mathbf{A} \text{ and } \mathbf{B}\text{)}
\end{aligned}$$

Therefore the relation $[Y_n, S_{i,n+1}^{-1}] = S_{(i)\delta,n+2}$ satisfies the claim and we conclude that the set of relations **R4** can be derived from **R4A**, **R4B** and the two infinite sets **A** and **B**.

***The set of relations* R5**

Recall that **R5** is the set

$$\{Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1 | \text{ for all } i \text{ in } \boldsymbol{N},\ k \text{ in } \mathbb{N}_0 \text{ and } n \text{ in } \mathbb{N}_1 \text{ such that } k < n\}.$$

To begin with we split the set into two disjoint subsets;

$$\begin{aligned}
&\mathbf{R5a} := \{Y_0^{-1} T_{i,n}^{-1} Y_0 T_{i,n+1} = 1 | \text{ for all } i \text{ in } \boldsymbol{N},\ n \text{ in } \mathbb{N}_1\}\\
&\mathbf{R5b} := \mathbf{R5} \backslash \mathbf{R5a} = \{Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1 | \text{ for all } i \text{ in } \boldsymbol{N},\ k, n \text{ in } \mathbb{N}_0 \text{ such that } k < n\}.
\end{aligned}$$

The relations in the set **R5a** can be replaced in a manner similar to those that were in

R2a and thus we replace **R5a** with

$$\mathbf{C} := \{Y_0^{-(n-1)} T_{i,1} Y_0^{(n-1)} = T_{i,n+1} | \text{for all } i \text{ in } \boldsymbol{N} \text{ and } n \text{ in } \mathbb{N}_1\}.$$

This set **C** is the third and final infinite set mentioned in the introduction to Stage 2.

Now consider $Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1$ from **R5b**, where $0 < k < n$. We can use the relations from **C** and the relation $Y_0^{-(k-1)} Y_1 Y_0^{(k-1)} = Y_k$ from **A** to rewrite $Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1$ as follows;

$$\begin{aligned}
&Y_k^{-1} T_{i,n} Y_k = T_{i,n+1} \\
&\Leftrightarrow (Y_0^{-(k-1)} Y_1^{-1} Y_0^{(k-1)})(Y_0^{-(n-1)} T_{i,1}^{-1} Y_0^{(n-1)})(Y_0^{-(k-1)} Y_1 Y_0^{(k-1)})(Y_0^{-n} T_{i,1} Y_0^{n}) = 1 \\
&\Leftrightarrow Y_0^{-(k-1)} Y_1^{-1} Y_0^{(k-n)} T_{i,1}^{-1} Y_0^{(n-k)} Y_1 Y_0^{(k-n-1)} T_{i,1} Y_0^{(n)} = 1 \\
&\Leftrightarrow Y_1^{-1} Y_0^{(k-n)} T_{i,1}^{-1} Y_0^{(n-k)} Y_1 Y_0^{(k-n-1)} T_{i,1} Y_0^{(n-k+1)} = 1 \\
&\Leftrightarrow Y_1^{-1} Y_0 Y_0^{(k-n-1)} T_{i,1}^{-1} Y_0^{(n-k+1)} Y_0^{-1} Y_1 Y_0^{(k-n-1)} T_{i,1} Y_0^{(n-k+1)} = 1 \\
&\Leftrightarrow [Y_0^{-1} Y_1, Y_0^{(k-n-1)} T_{i,1} Y_0^{(n-k+1)}] = 1 \\
&\Leftrightarrow [Y_0^{-1} Y_1, T_{i,n-k+2}] = 1 \\
&\Leftrightarrow [Y_1^{-1} Y_0, T_{i,n-k+2}] = 1.
\end{aligned}$$

This gives the double implication

$$Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1 \Leftrightarrow [Y_1^{-1} Y_0, T_{i,n-k+2}] = 1. \tag{$\star\star\star$}$$

Thus by $(\star\star\star)$, if we can prove that for all $m \geq 3$ the relation $[Y_1^{-1} Y_0, T_{i,m}] = 1$ can be derived from the relations **R1A, R1B, R5A, R5B** together with the two infinite sets **A** and **C**, then we will have proved the same claim for $Y_k^{-1} T_{i,n}^{-1} Y_k T_{i,n+1} = 1$ for all $0 < k < n$. Consider the case when $m = 3$. Using **C** and **R5A** we can derive $[Y_1^{-1} Y_0, T_{i,3}] = 1$ in the following way;

$$\begin{aligned}
Y_1^{-1} Y_0 T_{i,3} &= Y_1^{-1} Y_0 Y_0^{-2} T_{i,1} Y_0^2 && \text{(from } \mathbf{C}\text{)} \\
&= Y_1^{-1} Y_0^{-1} T_{i,1} Y_0^2 \\
&= (Y_0^{-1} Y_0) Y_1^{-1} Y_0^{-1} T_{i,1} Y_0^2 && \text{(insertion of the identity)} \\
&= Y_0^{-1} (Y_0 Y_1^{-1})(Y_0^{-1} T_{i,1} Y_0) Y_0 \\
&= Y_0^{-1} (Y_0^{-1} T_{i,1} Y_0)(Y_0 Y_1^{-1}) Y_0 && \text{(by } \mathbf{R5A}\text{)} \\
&= Y_0^{-2} T_{i,1} Y_0^2 Y_1^{-1} Y_0 \\
&= T_{i,3} Y_1^{-1} Y_0. && \text{(from } \mathbf{C}\text{)}
\end{aligned}$$

Thus by using relations from **C** and **R5A** we can derive the relation

$$[Y_1^{-1}Y_0, T_{i,3}] = 1. \tag{4.17}$$

Therefore, as (4.17) satisfies the claims, by $(\star\star\star)$ every relation $Y_{n-1}^{-1}T_{i,n}Y_{n-1} = T_{i,n+1}$ where $n \geq 1$ can be derived from the relations **C** and **R5A**.

The cases when $m \geq 4$ are proved by induction. In the proof we will again make use of the fact that for any $m \geq 3$ the relations $[Y_1^{-1}Y_0, Y_m]$ have already been proven to be derivable from **A, R1A** and **R1B**. First consider the initial case, $m = 4$.

$$\begin{aligned}
Y_1^{-1}Y_0T_{i,4} &= Y_1^{-1}Y_0Y_0^{-3}T_{i,1}Y_0^3 && \text{(from } \mathbf{C}\text{)}\\
&= Y_1^{-1}Y_0^{-2}T_{i,1}Y_0^2Y_0 && \\
&= (Y_0^{-1}Y_0)Y_1^{-1}Y_0^{-2}T_{i,1}Y_0^2Y_0 && \text{(insertion of the identity)}\\
&= Y_0^{-1}(Y_0^{-2}T_{i,1}Y_0^2)(Y_0Y_1^{-1})Y_0 && \text{(by } \mathbf{R5B}\text{)}\\
&= Y_0^{-3}T_{i,1}Y_0^3Y_1^{-1}Y_0 && \\
&= T_{i,4}Y_1^{-1}Y_0 && \text{(from } \mathbf{C}\text{).}
\end{aligned}$$

Thus the relation $[Y_1^{-1}Y_0, S_{i,5}] = 1$ can be derived from **C** and **R5B**.

Suppose $[Y_1^{-1}Y_0, T_{i,m}] = 1$ satisfies the claim for some $m \geq 4$. Then consider the product $Y_1^{-1}Y_0T_{i,m+1}$;

$$\begin{aligned}
Y_1^{-1}Y_0T_{i,m+1} &= (Y_1^{-1}Y_0)Y_{m-1}^{-1}T_{i,m}Y_{m-1} && \text{(from the consequence of (4.17))}\\
&= Y_{m-1}^{-1}(Y_1^{-1}Y_0)T_{i,m}Y_{m-1} && \text{(from } \mathbf{A,\ R1A} \text{ and } \mathbf{R1B}\text{)}\\
&= Y_{m-1}^{-1}T_{i,m}(Y_1^{-1}Y_0)Y_{m-1} && \text{(by our assumption)}\\
&= Y_{m-1}^{-1}T_{i,m}Y_{m-1}(Y_1^{-1}Y_0) && \text{(from } \mathbf{A,\ R1A} \text{ and } \mathbf{R1B}\text{)}\\
&= T_{i,m+1}Y_1^{-1}Y_0 && \text{(from the consequence of (4.17))}
\end{aligned}$$

Thus, by induction, any relation of the form $[Y_1^{-1}Y_0, T_{i,m}] = 1$ for some $m \geq 4$ can be derived using relations from the sets **R1A, R1B, R5A** and **R5B** together with the infinite sets **A** and **C**. Therefore we conclude that the claim is true for every relation in **R5b**.

***The set of relations* R6**

Our attention now turns to the set of relations **R6**

$$\{S_{i,n}S_{j,m} = S_{j,m}S_{i,n} | \text{for all } i \text{ and } j \text{ in } \boldsymbol{N} \text{ and } n \text{ and } m \text{ in } \mathbb{N}_1 \text{ such that } n > m\},$$

which we will show can be derived from the seven relations **R2A, R3A, R3C, R6A, R6B, R6C** and **R6D**, together with the infinite set of relations **B**. We consider the set **R6** as two disjoint subsets;

$$\begin{aligned}\mathbf{R6a} :=& \{[S_{i,n}, S_{j,1}] = 1 | \text{for all } n \geq 2\} \\ \mathbf{R6b} :=& \mathbf{R6} \setminus \mathbf{R6a} = \{[S_{i,n}, S_{j,m}] = 1 | \text{for all } 2 \leq m < n\}.\end{aligned}$$

Our proposal is that the relations in **R6a** can be derived from the four finite sets of relations **R2A, R3A, R6A** and **R6B**, together with the infinite set of relations **B**. When $n = 2$ the relations are exactly the set **R6A**. We will prove by induction on n that all the relations of the form $[S_{i,n}, S_{j,1}] = 1$ for when $n \geq 3$ can be derived from the four sets **R2A, R3A, R6B** and **B**. The initial case when $n = 3$ is constructed directly by combining **R6B** with relations from **B**. Suppose the relation $[S_{i,n}, S_{j,1}] = 1$ satisfies the proposition for some for some i and j in $\boldsymbol{N}$ and $n \geq 3$. Consider the product $S_{i,n+1}S_{j,1}$;

$$\begin{aligned}S_{i,n+1}S_{j,1} &= (Y_1^{-1}S_{i,n}Y_1)S_{j,1} && \text{by } \mathbf{R2A} \text{ and } \mathbf{B} \\ &= Y_1^{-1}S_{i,n}S_{j,1}Y_1 && \text{by } \mathbf{R3A} \\ &= Y_1^{-1}S_{j,1}S_{i,n}Y_1 && \text{by our assumption} \\ &= S_{j,1}Y_1^{-1}S_{i,n}Y_1 && \text{by } \mathbf{R3A} \\ &= S_{i,n+1}S_{j,1} && \text{by } \mathbf{R2A} \text{ and } \mathbf{B}.\end{aligned}$$

Thus the relation $[S_{i,n+1}, S_{j,1}] = 1$ can be constructed from **R2A, R3A, R6B** and the infinite set of relations **B**. Hence by induction our claim is true.

The subset **R6b** consists of all the relations in **R6** that are not in **R6a**. The proposal is that this subset can be derived from the relations **R2A, R3C, R6C, R6D** and the infinite set **B**. As we have done in previous cases, we first rewrite the set of relations using **R2A** and **B**. Consider the relation $[S_{i,n}S_{j,m}] = 1$ in **R6b**;

$$\begin{aligned}&S_{i,n}^{-1}S_{j,m}^{-1}S_{i,n}S_{j,m} = 1 \\ &\Leftrightarrow (Y_0^{-(n-2)}S_{i,2}^{-1}Y_0^{n-2})(Y_0^{-(m-2)}S_{j,2}^{-1}Y_0^{m-2})(Y_0^{-(n-2)}S_{i,2}Y_0^{n-2})(Y_0^{-(m-2)}S_{j,2}Y_0^{m-2}) = 1 \\ &\Leftrightarrow Y_0^{m-n}S_{i,2}^{-1}Y_0^{n-m}S_{j,2}^{-1}Y_0^{m-n}S_{i,2}Y_0^{n-m}S_{j,2} = 1 \\ &\Leftrightarrow S_{i,n-m+2}^{-1}S_{j,2}^{-1}S_{i,n-m+2}S_{j,2} = 1.\end{aligned}$$

Therefore, by the double implication

$$S_{i,n}^{-1}S_{j,m}^{-1}S_{i,n}S_{j,m} = 1 \Leftrightarrow S_{i,n-m+2}^{-1}S_{j,2}^{-1}S_{i,n-m+2}S_{j,2} = 1,$$

by proving that the set of relations $\{[S_{i,n}S_{j,2}] = 1| \text{ for all } n \geq 3\}$ can be constructed from **R2A, R3C, R6C, R6D** and the infinite set **B** we will have proved our original proposal. The proof follows by a similar method to our proof involving **R6a**. The relations in **R6b** where $n = 3$ are constructed from those in the set **R6C** combining with **B**. The rest of the cases will be proved by induction on n. The initial case when $n = 4$ is constructed directly by combining **R6D** with relations from **B**. Suppose the relation $[S_{i,n}S_{j,2}] = 1$ can be constructed using **R2A, R3C, R6D** and the infinite set **B** for some for some i and j in $\boldsymbol{N}$ and $n \geq 4$. Consider then the product $S_{i,n+1}S_{j,2}$;

$$\begin{aligned}
S_{i,n+1}S_{j,2} &= (Y_2^{-1}S_{i,n}Y_2)S_{j,2} && \text{by } \mathbf{R2A} \text{ and } \mathbf{B} \\
&= Y_2^{-1}S_{i,n}S_{j,2}Y_2 && \text{by } \mathbf{R3C} \\
&= Y_2^{-1}S_{j,2}S_{i,n}Y_2 && \text{by our assumption} \\
&= S_{j,2}Y_2^{-1}S_{i,n}Y_2 && \text{by } \mathbf{R3C} \\
&= S_{i,n+1}S_{j,2} && \text{by } \mathbf{R2A} \text{ and } \mathbf{B}.
\end{aligned}$$

Thus the relation $[S_{i,n+1}S_{j,2}] = 1$ can be constructed from **R2A, R3C, R6D** and the infinite set of relations **B**. Hence by induction our claim is true. Therefore the set of relations **R6** can be construced from the seven relations **R2A, R3A, R3C, R6A, R6B, R6C** and **R6D**, together with the infinite set of relations **B**.

***The set of relations* R7**

The next set of relations to consider is **R7**, which we recall as the set

$$\{S_{i,n}S_{j,n} = S_{(i,j)\Delta} | \text{for all } i \text{ and } j \text{ in } \boldsymbol{N} \text{ and } n \text{ in } \mathbb{N}_1\}.$$

This set can easily be seen to be derived from the two relations **R7A** and **R7B** coupled with the infinite set **B**. All the relations in **R7** for when $n = 1$ and $n = 2$ are contained within the sets **R7A** and **R7B** respectively. Consider the product $S_{i,n}S_{j,n}$ for some for some i and j in $\boldsymbol{N}$ and $n \geq 3$;

$$\begin{aligned}
S_{i,n}S_{j,n} &= Y_0^{-(n-2)}S_{i,2}Y_0^{(n-2)}Y_0^{-(n-2)}S_{j,2}Y_0^{(n-2)} && \text{from } \mathbf{B} \\
&= Y_0^{-(n-2)}S_{i,2}S_{j,2}Y_0^{(n-2)} \\
&= Y_0^{-(n-2)}S_{(i,j)\Delta,2}Y_0^{(n-2)} && \text{by } \mathbf{R7B} \\
&= S_{(i,j)\Delta,n}. && \text{from } \mathbf{B}
\end{aligned}$$

Thus **R7** can be constructed from the two relations **R7A** and **R7B** together with the infinite set **B**.

***The set of relations* R8**

The claim for **R8** is that this set of relations can be constructed from **R8A, R8B** and the infinite set **C**. We leave the proof to the reader as it is identical in its construction to the one just given for **R7**.

***The set of relations* R9**

The next set of relations to consider is **R9**, which we recall as the set

$$\{T_{i,n} = T_{(i)\delta,n+1}S_{i,n+1} | \text{for all } i \text{ in } \boldsymbol{N} \text{ and } n \text{ in } \mathbb{N}_0\}.$$

The relations in **R9** for when $n = 0$ are exactly those in **R9A**. Likewise, the relations in **R9** where $n = 1$ are simply those in the set **R9A** where we have replaced the string $Y_0^{-1}T_{i,1}Y_0$ with $T_{i,2}$ by the corresponding relation in **C**. From **R9A, R9B** and the two infinite sets of relations **B** and **C** the rest of **R9** can be constructed as follows. Consider the element $T_{i,n}$ for some i in $\boldsymbol{N}$ and $n \geq 2$;

$$\begin{aligned}
T_{i,n} &= Y_0^{-(n-1)}T_{i,1}Y_0^{(n-1)} && \text{by } \mathbf{C} \\
&= Y_0^{-(n-1)}T_{(i)\delta,2}S_{i,2}Y_0^{(n-1)} && \text{by } \mathbf{R9A} \\
&= Y_0^{-(n-1)}T_{(i)\delta,2}(Y_0^{(n-1)}Y_0^{-(n-1)})S_{i,2}Y_0^{(n-1)} && \text{by insertion of the identity} \\
&= T_{,n+1}S_{i,n+1}. && \text{by } \mathbf{C} \text{ and } \mathbf{B}
\end{aligned}$$

Thus the set of relations **R9** can be constructed from **R9A, R9B** and the two infinite sets **B** and **C**.

***The set of relations* R10**

The next set of relations to consider is **R10**,

$$\{[S_{i,k}, T_{j,n}] = 1 | \text{for all } i \text{ and } j \text{ in } \boldsymbol{N} \text{ and } k \text{ and } n \text{ in } \mathbb{N}_1, \text{ such that } k \leq n\},$$

which we will show can be constructed from the seven relations **R3A, R3C, R5A, R10A, R10B, R10C, R10D** together with the two infinite sets **B** and **C**. To prove this claim we split the set **R10** into two distinct subsets, **R10a** and **R10b**, and consider each in turn;

$$\begin{aligned}
\mathbf{R10a} &:= \{[S_{i,1}, T_{j,n}] = 1| \text{ for all } n \geq 1\} \\
\mathbf{R10b} &:= \mathbf{R10} \setminus \mathbf{R10a} = \{[S_{i,m}, T_{j,n}] = 1| \text{ for all } 1 \leq m \leq n\}.
\end{aligned}$$

We first consider the set **R10a**. When $n = 1$ the relations from **R10a** are exactly those from the set **R10A**. When $n = 2$ the relations from **R10a** are simply those from **R10B** where the string $Y_0^{-1}T_{i,1}Y_0$ is replaced with $T_{i,2}$ by the relation in **C**. Consider then the product $S_{i,1}T_{j,n}$ for some i and j in $\boldsymbol{N}$ and some natural number $n \geq 3$. Using relations from the sets **R3A, R5A, R10B** and **C** this product becomes the following;

$$\begin{aligned}
S_{i,1}T_{j,n} &= S_{i,1}(Y_1^{-(n-2)}T_{j,2}Y_1^{(n-2)}) && \text{by } \mathbf{R5A} \text{ and } \mathbf{C}\\
&= Y_1^{-(n-2)}S_{i,1}T_{j,2}Y_1^{(n-2)} && \text{by } \mathbf{R3A}\\
&= Y_1^{-(n-2)}T_{j,2}S_{i,1}Y_1^{(n-2)} && \text{by } \mathbf{R10B}\\
&= Y_1^{-(n-2)}T_{j,2}Y_1^{(n-2)}S_{iX_n^{-b_n}X_{n-1}^{-b_{n-1}}...X_0^{-b_0},1} && \text{by } \mathbf{R3A}\\
&= T_{j,n}S_{i,1}. && \text{by } \mathbf{R5A} \text{ and } \mathbf{C}
\end{aligned}$$

Thus the relation $[S_{i,1}T_{j,n}] = 1$ can be constructed from **R3A, R5A, R10B** and the infinite set **C**, and hence our claim is true for **R10a**.

The subset **R10b** contains all the relations of **R10** that are not found in **R10a**. Our claim is that we can construct the relations in **R10b** using the relations **R3C, R5A, R10C, R10D** and the two infinite sets **B** and **C**. Using a technique similar to that used in many of the proofs that have gone before, we first rewrite the relations in **R10b** using the infinite sets **B** and **C**. Consider the relation $[S_{i,m}, T_{j,n}] = 1$ for some natural numbers m and n such that $2 \leq m \leq n$;

$$\begin{aligned}
&S_{i,m}^{-1}T_{j,m}^{-1}S_{i,m}T_{j,m} = 1\\
&\Leftrightarrow (Y_0^{-(m-2)}S_{i,2}^{-1}Y_0^{m-2})(Y_0^{-(n-2)}T_{j,2}^{-1}Y_0^{n-2})(Y_0^{-(m-2)}S_{i,2}Y_0^{m-2})(Y_0^{-(n-2)}T_{j,2}^{-1}Y_0^{n-2}) = 1\\
&\Leftrightarrow S_{i,2}^{-1}Y_0^{m-n}T_{j,2}^{-1}Y_0^{n-m}S_{i,2}Y_0^{m-n}T_{j,2}^{-1}Y_0^{n-m} = 1\\
&\Leftrightarrow S_{i,2}^{-1}T_{j,n-m+2}^{-1}S_{i,2}T_{j,n-m+2} = 1.
\end{aligned}$$

Thus, by the double implication $S_{i,m}^{-1}T_{j,m}^{-1}S_{i,m}T_{j,m} = 1 \Leftrightarrow S_{i,2}^{-1}T_{j,n-m+2}^{-1}S_{i,2}T_{j,n-m+2} = 1$, to prove our claim regarding **R10b** it is enough to prove the claim for the set $\{[S_{i,2}, T_{j,n}] = 1| \text{ for } n \geq 2\}$.

The first two cases, when $n = 2$ and $n = 3$, are taken from the relations **R10C** and **R10D** and combining them with relations from **C**. Consider then the product $S_{i,2}T_{j,n}$ for some i and j in $\boldsymbol{N}$ and $n \geq 4$. Using the relations **R3C, R5A, R10D** and the infinite set **C** we have the following;

$$\begin{aligned}
S_{i,2}T_{j,n} &= S_{i,2}(Y_2^{-(n-3)}T_{j,3}Y_2^{(n-3)}) && \text{by } \mathbf{R5A} \text{ and } \mathbf{C}\\
&= Y_2^{-(n-3)}S_{i,2}T_{j,3}Y_2^{(n-3)} && \text{by } \mathbf{R3C}\\
&= Y_2^{-(n-3)}T_{j,3}S_{i,2}Y_2^{(n-3)} && \text{by } \mathbf{R10D}\\
&= Y_2^{-(n-3)}T_{j,3}Y_2^{(n-3)}S_{i,2} && \text{by } \mathbf{R3C}\\
&= T_{j,n}S_{i,2}. && \text{by } \mathbf{R5A} \text{ and } \mathbf{C}
\end{aligned}$$

Therefore the claim for **R10b** is true and thus the set of relations **R10** can be construed from the seven relations **R3A, R3C, R5A, R10A, R10B, R10C, R10D** and the two infinite sets **B** and **C**.

***The set of relations* R11**

The penultimate set of relations from R is **R11**=$\{S_{1,n} = 1|$ for all $n \geq 1\}$. The proposal is that this set can be constructed from the two relations **R11A, R11B** and the infinite set **B**. The two initial cases, when $n = 1$ and $n = 2$, are exactly the relations **R11A** and **R11B**. Consider the element $S_{1,n}$ for some $n \geq 3$. By **B** and **R11B** we have the following;

$$\begin{aligned} S_{1,n} &= Y_0^{-(n-2)} S_{i,2} Y_0^{(n-2)} && \text{by } \mathbf{B} \\ &= Y_0^{-(n-2)} Y_0^{(n-2)} && \text{by } \mathbf{R11B} \\ &= 1 \end{aligned}$$

Thus **R11** can be constructed from the two relations **R11A, R11B** and the infinite set **B**.

***The set of relations* R12**

The final relation to consider is **R12** $= \{T_{1,n} = 1|$ for all n in $\mathbb{N}_0\}$, which can be constructed from the relations **R12A, R12B** and the infinite set **C**. The proof follows the same outline as that for **R11** above.

Therefore, beginning from the presentation $\widehat{F}_1 = \langle X|R \cup R_f\rangle$, we have shown that the relations in R can be constructed from the relations in R_f and the three infinite sets **A, B** and **C**. Thus we can define a presentation

$$\widehat{F}_2 := \langle X|R_f \cup \mathbf{A} \cup \mathbf{B} \cup \mathbf{C}\rangle$$

such that $\widehat{F}_2 \cong \widehat{F}_1$.

Stage 3

We have now reached the third and final stage in our proof. The presentation $\widehat{F}_2$ is still an infinite presentation with an infinite generating set X and an infinite set of relations. However, in this final stage we will use the Tietze transformation **T4** to create a finite presentation from $\widehat{F}_2$. Our first step is to break down the generating set X from the presentation $\widehat{F}_2$ into three subsets which we consider separately.

X1 $\{Y_n|n \in \mathbb{N}_0\}$,

X2 $\{S_{i,n} | 1 \leq i \leq N, n \in \mathbb{N}_1\}$,

X3 $\{T_{i,n} | 1 \leq i \leq N, n \in \mathbb{N}_1\}$.

We first consider the subset **X1**. Define the set **X1a** as

$$\mathbf{X1a} := \{Y_n | \text{for all } n \geq 2\}.$$

Recall that the set of relations **A** consists solely of the relations $Y_n = Y_0^{-(n-1)} Y_1 Y_0^{(n-1)}$ for every $n \geq 2$. Therefore the set of generators **X1a** and the set of relations **A** satisfy the conditions that are required to apply the Tietze transformation **T4** to $\widehat{F}_2$ with respect to **X1a** and **A**. Noting that the relations in **B** and **C** do not use Y_n or Y_n^{-1} for any $n \geq 1$, the Tietze transformation **T4** allows us to produce a presentation of a new group $\widehat{F}_3$ isomorphic to $\widehat{F}_2$ as follows.

$$\widehat{F}_3 := \langle X \setminus \mathbf{X1a} | R_f \cup \mathbf{B} \cup \mathbf{C} \rangle.$$

A similar process happens between the set of generators **X2** and the set of relations **B**. Define the set **X2a** as

$$\mathbf{X2a} := \{S_{i,n} | \text{for all } i \text{ in } \boldsymbol{N} \text{ and } n \geq 3\}.$$

The set **B** consists solely of relations of the form $S_{i,n} = Y_0^{-(n-2)} S_{i,2} Y_0^{(n-2)}$ for every $n \geq 3$, where each generator $S_{i,n}$ is expressed as a string constructed from the set of generators $\{Y_0, Y_0^{-1}, S_{i,2}\}$. Therefore, by applying the transformation **T4** to the presentation $\widehat{F}_3$ with respect to the set of generators **X2a** and the set of relations **B**, we produce the presentation

$$\widehat{F}_4 := \langle X \setminus (\mathbf{X1a} \cup \mathbf{X2a}) | R_f \cup \mathbf{C} \rangle.$$

Likewise, for the final subset of generators $\mathbf{X3} = \{T_{i,n} | 1 \leq i \leq N, n \in \mathbb{N}_1\}$ we define the set

$$\mathbf{X3a} := \{T_{i,n} | \text{for all } i \text{ in } \boldsymbol{N} \text{ and } n \geq 2\}.$$

The relations from **C** facilitate the removal of all generators $T_{i,n}$ where $n \geq 2$ from $\widehat{F}_4$, the verification of which we leave to the reader as it follows the same method used in the previous two cases. The group presentation that remains after the third stage is the finite presentation $\langle X \setminus (\mathbf{X1a} \cup \mathbf{X2a} \cup \mathbf{X3a}) | R_f \rangle = \langle X_f | R_f \rangle = F_{fin}$. □

Now we have shown that $F_{(G,\theta)}$ has a finite presentation given by F_{fin}, it remains for us to remove the generators of the form $S_{i,1}$ and $S_{i,2}$ to complete the proof of Theorem 4.3.

Proof of Theorem 4.3. The relations **R9A** and **R9B** can be rearranged to read;

$$T^{-1}_{(i)\delta,1} T_{i,0} = S_{i,1} \qquad\qquad Y_0^{-1} T^{-1}_{(i)\delta,1} Y_0 T_{i,1} = S_{i,2},$$

therefore one can use the Tietze transformation **T4** to remove the generators $S_{i,1}$ and $S_{i,2}$ in F_{fin}. The transformation replaces all the appearances of $S_{i,1}$ and $S_{i,2}$ in the relations of F_{fin} with their respective products given above. After these replacements are made, one has produced the presentation $\mathcal{F}^{fin}_{(G,\theta)}$ appearing in Theorem 4.3. For example, the relations in **R2A** become the set of relations of the form

$$[Y_0 Y_1^{-1}, Y_0^{-1}(T^{-1}_{(i)\delta,1})^{Y_0} T_{i,1} Y_0] = 1,$$

which is the set **S3** in $\mathcal{F}^{fin}_{(G,\theta)}$. $\square$

Chapter 5

Embedding Groups from $\mathfrak{F}_{aug}$ into V

5.1 Statement of results

In this final chapter we will prove the following theorem.

Theorem 5.1. *Suppose G is a finite, abelian group and θ is an idempotent endomorphism of G. Then $F_{(G,\theta)}$ embeds into Thompson's group V.*

Recall that an endomorphism $\theta : G \to G$ is idempotent if and only if $\theta^2 = \theta$. The proof of Theorem 5.1 follows from two results. The first is an isomorphism result for groups in $\mathfrak{F}_{aug}$.

Theorem 5.2. *Suppose $F_{(G,\theta)}$ and $F_{(G,\phi)}$ are groups from $\mathfrak{F}_{aug}$ such that G is abelian and the endomorphisms θ and ϕ are both idempotent. Then $F_{(G,\theta)} \cong F_{(G,\phi)}$.*

The proof of Theorem 5.2 follows from the presentation of the groups given in the previous chapter. We will show that when G is abelian and θ is idempotent the infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$ for $F_{(G,\theta)}$ can be rewritten to be independent of θ. This result is somewhat surprising and we give an example illustrating the isomorphism between two such groups.

The second result that is needed to prove Theorem 5.1 is found in [3]. In Remark 2.12 of [3] the authors state that if G is a finite group and θ is the identity endomorphism of G (the endomorphism that sends every group element to itself,) then the group $V_{(G,\theta)}$ embeds into V. We provide a proof this statement in the second half of the chapter, together with an example.

Since $F_{(G,\theta)}$ is a subgroup of $V_{(G,\theta)}$ this also provides an embedding of $F_{(G,\theta)}$ into V. As the identity homomorphism is clearly idempotent, by Theorem 5.2 every $F_{(G,\theta)}$ for which G is abelian and θ is idempotent must also embed into V. Thus the result follows for Theorem 5.1.

5.2 Isomorphism result

In what follows $G = \{g_i\}_{i \in \boldsymbol{N}}$ is a finite group of order N, where g_1 is the identity. We define θ to be an idempotent endomorphism of G, that is, $(g)\theta^2 = (g)\theta$ for all $g \in G$. Much of the notation we use is carried forward from Chapter 4. Recall that $\delta : \boldsymbol{N} \to \boldsymbol{N}$ is a function defined by $(i)\delta = j$ iff $(g_i)\theta = g_j$. Note that if θ is idempotent then so is δ. Additionally $\Delta : \boldsymbol{N} \times \boldsymbol{N} \to \boldsymbol{N}$ is a function designed to translate group multiplication from G to the set $\boldsymbol{N}$, where $(i,j)\Delta = k$ iff $g_i \cdot g_j = g_k$.

As was mentioned above, we use the infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$ to prove Theorem 5.2. The proof proceeds by using Tietze transformations on $\mathcal{F}^{inf}_{(G,\theta)}$ until we have produced a group presentation that represents the same group as $\mathcal{F}^{inf}_{(G,\theta)}$ but without the dependence on the endomorphism θ of G.

Theorem 5.2. *Suppose $F_{(G,\theta)}$ and $F_{(G,\phi)}$ are groups from $\mathfrak{F}_{aug}$ such that G is abelian and the endomorphisms θ and ϕ are both idempotent. Then $F_{(G,\theta)} \cong F_{(G,\phi)}$.*

Proof of Theorem 5.2. We begin the proof by using the set of relations in **R9**, in conjunction with the Tietze transformation **T4**, to remove all the S-generators (those of the form $S_{i,n}$) from the infinite presentation $\mathcal{F}^{inf}_{(G,\theta)}$, in similar fashion to the replacement we carried out for the finite presentation $\mathcal{F}^{fin}_{(G,\theta)}$ at the end of the last chapter. Thus every appearance of the generator $S_{i,n}$ in the relations of $\mathcal{F}^{inf}_{(G,\theta)}$ is replaced with the word $T^{-1}_{(i)\delta,n}T_{i,n-1}$. We call the resulting presentation $\widehat{P}_1 = \langle X_1 | R_1 \rangle$, where the sets X_1 and R_1 are given below.

Set of generators: $X_1 := \mathbf{X1} \sqcup \mathbf{X3}$:

$\mathbf{X1} := \{Y_n | n \in \mathbb{N}_0\}$,

$\mathbf{X3} := \{T_{i,n} | 1 \leq i \leq N, n \in \mathbb{N}_0\}$.

Set of relations: R_1 is the disjoint union of the following sets;

R1′ $Y_k^{-1} Y_n Y_k = Y_{n+1}$,

for all k and n in $\mathbb{N}_0$ such that $k < n$,

R2′ $Y_k^{-1} T^{-1}_{(i)\delta,n} T_{i,n-1} Y_k = T^{-1}_{(i)\delta,n+1} T_{i,n}$,

for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n-1$,

R3′ $Y_k^{-1}T_{(i)\delta,n}^{-1}T_{i,n-1}Y_k = T_{(i)\delta,n}^{-1}T_{i,n-1}$,

for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k \geq n$,

R4′ $Y_{n-1}^{-1}T_{(i)\delta,n}^{-1}T_{i,n-1}Y_{n-1} = T_{\delta^2(i),n+1}^{-1}T_{i,n-1}$,

for all i in $\boldsymbol{N}$ and n in $\mathbb{N}_1$,

R5′ $Y_k^{-1}T_{i,n}Y_k = T_{i,n+1}$,

for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n$,

R6′ $T_{(i)\delta,n}^{-1}T_{i,n-1}T_{(j)\delta,m}^{-1}T_{j,m-1} = T_{(j)\delta,m}^{-1}T_{j,m-1}T_{(i)\delta,n}^{-1}T_{i,n-1}$,

for all i and j in $\boldsymbol{N}$ and n and m in $\mathbb{N}_1$ such that $n > m$,

R7′ $T_{(i)\delta,n}^{-1}T_{i,n-1}T_{(j)\delta,n}^{-1}T_{j,n-1} = T_{((i)\delta,(j)\delta)\Delta,n}^{-1}T_{(i,j)\Delta,n-1}$,

for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_1$,

R8′ $T_{i,n}T_{j,n} = T_{(i,j)\Delta,n}$,

for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_0$,

R10′ $T_{(i)\delta,k}^{-1}T_{i,k-1}T_{j,n} = T_{j,n}T_{(i)\delta,k}^{-1}T_{i,k-1}$,

for all i and j in $\boldsymbol{N}$ and k and n in $\mathbb{N}_1$, such that $k \leq n$,

R12′ $T_{1,n} = 1$,

for all n in $\mathbb{N}_0$.

Removing the relations from R2′

One may observe that the relations in **R2′** immediately arise as consequences of the relations in **R5′**.

$$\begin{aligned}
Y_k^{-1}T_{(i)\delta,n}^{-1}T_{i,n-1}Y_k &= Y_k^{-1}T_{(i)\delta,n}^{-1}(Y_kY_k^{-1})T_{i,n-1}Y_k && \text{inserting the identity}\\
&= (Y_k^{-1}T_{(i)\delta,n}^{-1}Y_k)(Y_k^{-1}T_{i,n-1}Y_k)\\
&= T_{(i)\delta,n+1}^{-1}T_{i,n} && \text{by } \mathbf{R5}.
\end{aligned}$$

Thus we can rewrite $\widehat{P}_1$ by removing the relations from **R2′** using **T2**. We call the new presentation $\widehat{P}_1 = \langle X_1 | R_2 \rangle$, where R_2 is the new set of relations with **R2′** removed.

Deriving the set of relations D

Note that none of the steps taken in the proof so far have relied on the fact that G is abelian and θ is idempotent. In fact, the steps above can be taken for arbitrary finite group and endomorphism. However, this changes in the proceeding steps and the rest of the proof is reliant on the properties of G and θ. In what follows we will take care to point out when our assumptions on G and θ are used.

The next two steps in the proof involve adding two new sets of relations, **D** and **E**, to $\widehat{P}_2$ defined as;

$$\begin{aligned}
\mathbf{D} &:= \{Y_k^{-1} T_{i,n} Y_k = T_{i,n} : \text{for all } i \text{ in } \boldsymbol{N},\ k \text{ and } n \text{ in } \mathbb{N}_0 \text{ such that } n \leq k\},\\
\mathbf{E} &:= \{T_{i,n} T_{j,k} = T_{j,k} T_{i,n} : \text{for all } i, j \text{ in } \boldsymbol{N},\ k \text{ and } n \text{ in } \mathbb{N}_0 \text{ such that } n \leq k\}.
\end{aligned}$$

We first show that the relations from **D** can be derived from those in R_2.

The proof proceeds by induction. Let $\mathcal{P}(m)$ be the statement that $Y_k^{-1} T_{i,n} Y_k = T_{i,n}$ can be derived from relations in $\mathcal{F}^{inf}_{(G,\theta)}$ where $k - n = m$ for some $m \geq 0$. Consider the initial case when $m = 0$ and thus $k = n$. We begin with the left hand side of the proposed relation.

$$\begin{aligned}
Y_n^{-1} T_{i,n} Y_n &= Y_n^{-1} (T_{(i)\delta,n+1} T^{-1}_{(i)\delta,n+1}) T_{i,n} Y_n && \text{inserting the identity}\\
&= (Y_n^{-1} T_{(i)\delta,n+1} Y_n)(Y_n^{-1} T^{-1}_{(i)\delta,n+1} T_{i,n} Y_n)\\
&= T_{(i)\delta,n+2} (Y_n^{-1} T^{-1}_{(i)\delta,n+1} T_{i,n} Y_n) && \text{by } \mathbf{R5'}\\
&= T_{(i)\delta,n+2} T^{-1}_{(i)\delta^2,n+2} T_{i,n} && \text{by } \mathbf{R4'}\\
&= T_{(i)\delta,n+2} T^{-1}_{(i)\delta,n+2} T_{i,n} && \text{by idempotency of } \delta\\
&= T_{i,n}.
\end{aligned}$$

Thus $\mathcal{P}(0)$ is true. Suppose then that $\mathcal{P}(m)$ is true for some $m \geq 0$. Consider the case $\mathcal{P}(m+1)$.

$$\begin{aligned}
Y_k^{-1} T_{i,n} Y_k &= Y_k^{-1} (T_{(i)\delta,n+1} T^{-1}_{(i)\delta,n+1}) T_{i,n} Y_k && \text{inserting the identity}\\
&= (Y_k^{-1} T_{(i)\delta,n+1} Y_k)(Y_k^{-1} T^{-1}_{(i)\delta,n+1} T_{i,n} Y_k)\\
&= T_{(i)\delta,n+1} (Y_k^{-1} T^{-1}_{(i)\delta,n+1} T_{i,n} Y_k) && \text{by } \mathcal{P}(m)\\
&= T_{(i)\delta,n+1} T^{-1}_{(i)\delta,n+1} T_{i,n} && \text{by } \mathbf{R3'}\\
&= T_{i,n}.
\end{aligned}$$

Thus by induction $\mathcal{P}(m)$ must be true for all $m \geq 0$. Hence the relations in **D** are consequence of the relations of R_2.

Deriving the set of relations E

We now prove that the relations from **E** are also consequences of the relations from R_2.

The proof proceeds by induction. Let $\mathcal{P}(m)$ be the statement that $T_{i,n} T_{j,k} = T_{j,k} T_{i,n}$ can be derived from the relations in $\mathcal{F}^{inf}_{(G,\theta)}$ where $k - n = m$ for some $m \geq 0$. We begin with the case $\mathcal{P}(0)$, when $n = k$. By $\mathbf{R8'}$, $T_{i,n} T_{j,n} = T_{(i,j)\Delta,n}$ for all i and j. As G is abelian $(i,j)\Delta = (j,i)\Delta$, thus $T_{i,n}T_{j,n} = T_{(j,i)\Delta,n} = T_{j,n}T_{i,n}$ and $\mathcal{P}(0)$ is true.

Suppose $\mathcal{P}(m)$ is true for some $m \geq 0$. Consider $\mathcal{P}(m+1)$.

$$\begin{aligned}
T_{i,n}T_{j,k} &= (T_{(i)\delta,n+1}T^{-1}_{(i)\delta,n+1})T_{i,n}T_{j,k} && \text{inserting the identity}\\
&= T_{(i)\delta,n+1}(T^{-1}_{(i)\delta,n+1}T_{i,n}T_{j,k}) && \text{rebracketing}\\
&= T_{(i)\delta,n+1}(T_{j,k}T^{-1}_{(i)\delta,n+1}T_{i,n}) && \text{by } \mathbf{R10'}\\
&= T_{j,k}T_{(i)\delta,n+1}T^{-1}_{(i)\delta,n+1}T_{i,n} && \text{by } \mathcal{P}(m)\\
&= T_{j,k}T_{i,n}.
\end{aligned}$$

Thus by induction $\mathcal{P}(m)$ is true for all $m \geq 0$. Hence the relations in **E** are consequence of the relations of R_2.

Therefore, we add the relations from **D** and **E** to $\widehat{P}_2$ by using **T1**, and create the presentation $\widehat{P}_3 = \langle X_1 | R_3 \rangle$, where $R_3 := R_2 \sqcup \boldsymbol{D} \sqcup \boldsymbol{E}$.

Removing relations from R_3 as consequences of D and E

Many of the relations in R_3 now follow as consequences of **D** and **E**. First notice that the relations in $\mathbf{R3'}$ follow directly from the relations in **D** in the following way.

$$\begin{aligned}
Y_k^{-1}T^{-1}_{(i)\delta,n}T_{i,n-1}Y_k &= Y_k^{-1}T^{-1}_{(i)\delta,n}(Y_kY_k^{-1})T_{i,n-1}Y_k && \text{inserting the identity}\\
&= (Y_k^{-1}T^{-1}_{(i)\delta,n}Y_k)(Y_k^{-1}T_{i,n-1}Y_k)\\
&= T^{-1}_{(i)\delta,n}T_{i,n-1} && \text{by } \mathbf{D}.
\end{aligned}$$

Thus we can remove $\mathbf{R3'}$ from the set of relations R_3 using the transformation **T2**.

Secondly notice that the relations in $\mathbf{R4'}$ follow from a combination of the relations in $\mathbf{R5'}$ and **D**.

$$\begin{aligned}
Y_{n-1}^{-1}T^{-1}_{(i)\delta,n}T_{i,n-1}Y_{n-1} &= Y_{n-1}^{-1}T^{-1}_{(i)\delta,n}(Y_{n-1}Y_{n-1}^{-1})T_{i,n-1}Y_{n-1} && \text{inserting the identity}\\
&= (Y_{n-1}^{-1}T^{-1}_{(i)\delta,n}Y_{n-1})(Y_{n-1}^{-1}T_{i,n-1}Y_{n-1})\\
&= T^{-1}_{(i)\delta,n+1}(Y_{n-1}^{-1}T_{i,n-1}Y_{n-1}) && \text{by } \mathbf{R5'}\\
&= T^{-1}_{(i)\delta,n+1}T_{i,n-1} && \text{by } \mathbf{D}
\end{aligned}$$

Thus we can also remove $\mathbf{R4'}$ from R_3 using the transformation **T2**.

The relations from **E** tells us that all T-generators from **X3** commute. Thus the relations

R6$'$, **R7**$'$ and **R10**$'$ immediately follow as consequences and can be removed from R_3 by the transformation **T2**.

Therefore the presentation $\widehat{P}_3$ been rewritten as the presentation $\widehat{P}_4 = \langle X_1 | R_4 \rangle$, where the sets X_1 and R_4 are given below.

Set of generators, $X_1 := \mathbf{X1} \sqcup \mathbf{X3}$, where:

X1 $\{Y_n | n \in \mathbb{N}_0\}$,

X3 $\{T_{i,n} | 1 \leq i \leq N, n \in \mathbb{N}_0\}$.

Set of relations: R_4 is the union of the following set;:

R1$'$ $Y_k^{-1} Y_n Y_k = Y_{n+1}$, for all k and n in $\mathbb{N}_0$ such that $k < n$,

R5$'$ $Y_k^{-1} T_{i,n} Y_k = T_{i,n+1}$, for all i in $\boldsymbol{N}$, k in $\mathbb{N}_0$ and n in $\mathbb{N}_1$ such that $k < n$,

R8$'$ $T_{i,n} T_{j,n} = T_{(i,j)\Delta,n}$, for all i and j in $\boldsymbol{N}$ and n in $\mathbb{N}_0$,

R12$'$ $T_{1,n} = 1$ for all n in $\mathbb{N}_0$,

D $Y_k^{-1} T_{i,n} Y_k = T_{i,n}$, for all i in $\boldsymbol{N}$, k and n in $\mathbb{N}_0$ such that $n \leq k$,

E $T_{i,n} T_{j,k} = T_{j,k} T_{i,n}$, for all i, j in $\boldsymbol{N}$, k and n in $\mathbb{N}_0$ such that $n \leq k$.

Therefore if G is a finite abelian group and θ and ϕ are two idempotent endomorphisms of G then $F_{(G,\theta)}$ and $F_{(G,\phi)}$ share a presentation and thus must be isomorphic.

$\square$

Recall that the group $F_{(G,\theta)}$ can be generated from the two infinite sets $\{X_n : n \in \mathbb{N}_0\}$ and $\{\Gamma_{i,n} : 1 \leq i \leq |G|, n \in \mathbb{N}_0\}$, which are formally defined by the tree pair representatives in Figures 4.6 and 4.8 of Chapter 4. Given two groups $F_{(G,\theta)}$ and $F_{(G,\phi)}$, where G is abelian and both θ and ϕ are idempotent, one can construct an isomorphism ψ between the two groups by mapping the generators of one to the generators of the other. Formally, if $F_{(G,\theta)} = \langle \{X_n\} \cup \{\Gamma_{i,n}\} \rangle$ and $F_{(G,\phi)} = \langle \{X'_n\} \cup \{\Gamma'_{i,n}\} \rangle$ then the isomorphism ψ between the two groups is defined on the generators as $(X_n)\psi = X'_n$ and $(\Gamma_{i,n})\psi = \Gamma'_{i,n}$.

Below we give an example of the isomorphism ψ between two such groups $F_{(G,\theta)}$ and $F_{(G,\phi)}$ in $\mathfrak{F}_{aug}$. We take two elements from $F_{(G,\theta)}$ and show that their product under ψ is well defined in $F_{(G,\phi)}$.

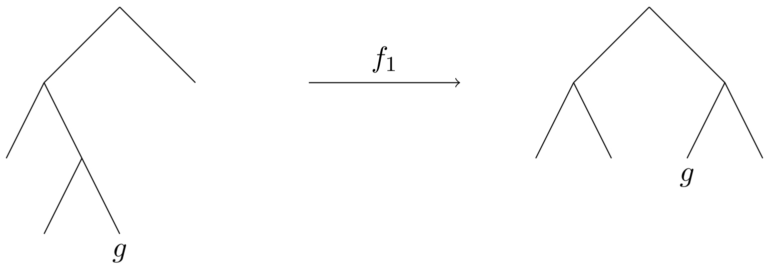
Example of the isomorphism

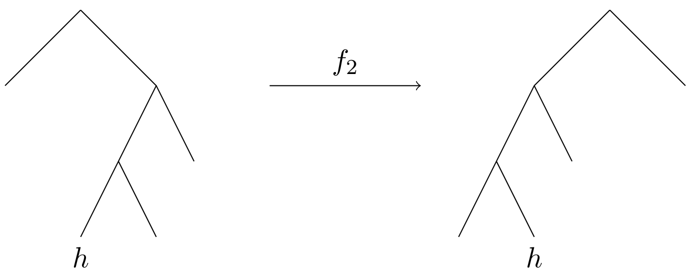
Let $G = C_2 \times C_4$, the direct product of the cyclic group of order two and the cyclic group of order four. Define the generators of G to be $(a, 1)$ and $(1, b)$, where $(a, 1)$ is the generator of the canonical embedding of C_2 inside G, and $(1, b)$ is the generator of the canonical embedding of C_4 inside G. Let θ and ϕ be two idempotent endomorphisms of G defined by

$$(x, y)\theta = (1, y) \tag{5.1}$$
$$(x, y)\phi = (1, 1). \tag{5.2}$$

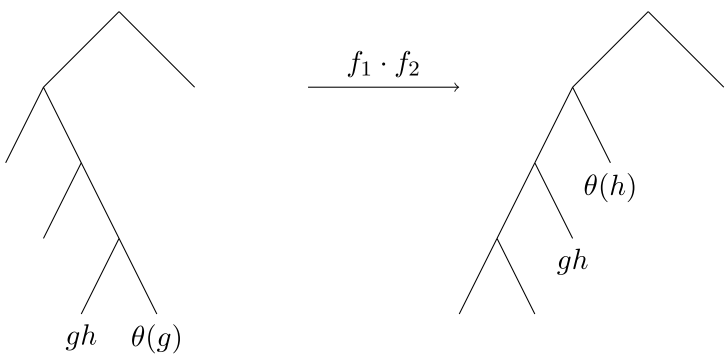
Let $\psi : F_{(G,\theta)} \to F_{(G,\phi)}$ be the isomorphism between the two groups as previously defined. Thus we refer to the generators of $F_{(G,\theta)}$ by the set $\{X_n\} \cup \{\Gamma_{i,n}\}$, and the generator of $F_{(G,\phi)}$ by $\{X'_n\} \cup \{\Gamma'_{i,n}\}$, where the isomorphism ψ maps $X_n \mapsto X'_n$ and $\Gamma_{i,n} \mapsto \Gamma'_{i,n}$.

Let $g = (a, b)$ and $h = (1, b^3)$ be two group elements of G. Consider two elements, f_1 and f_2, of $F_{(G,\theta)}$, defined by their reduced tree pair representatives below.





The product $f_1 \cdot f_2$ is given by the augmented tree pair below.



To map these elements into the group $F_{(G,\phi)}$ we first need to write them as a product of generators. This is achieved by putting them into PAN form and then replacing all the elements from the set $\{\Sigma_{i,n}\}$ with those from $\{\Gamma_{i,n}\}$ using the relation $\Sigma_{i,n} = \Gamma_{i,n-1}\Gamma^{-1}_{\theta(i),n}$.

$$\begin{aligned}
f_1 &= X_0X_1\Sigma_{g,3}X_0^{-1} \\
f_2 &= X_1\Sigma_{h,2}X_0^{-2} \\
f_1 \cdot f_2 &= X_0X_1X_2\Sigma_{gh,3}\Sigma_{\theta(g),4}X_0^{-3}.
\end{aligned}$$

Replacing the elements from $\{\Sigma_{i,n}\}$ in f_1, f_2 and $f_1 \cdot f_2$ gives the following;

$$\begin{aligned}
f_1 &= X_0X_1\Gamma_{g,2}\Gamma^{-1}_{\theta(g),3}X_0^{-1} \\
f_2 &= X_1\Gamma_{h,1}\Gamma^{-1}_{\theta(h),2}X_0^{-2} \\
f_1 \cdot f_2 &= X_0X_1X_2\Gamma_{gh,2}\Gamma^{-1}_{\theta(gh),3}\Gamma_{\theta(g),3}\Gamma^{-1}_{\theta^2(g),4}X_0^{-3}.
\end{aligned}$$

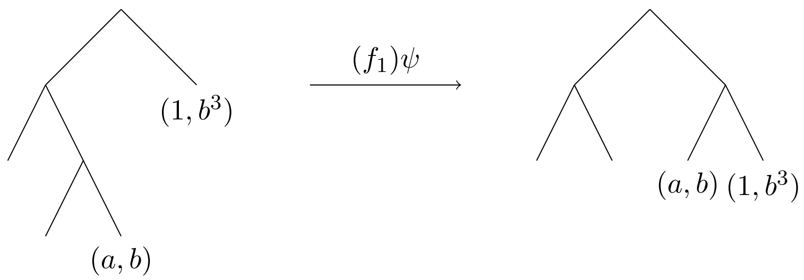
Under the isomorphism ψ from $F_{(G,\theta)}$ to $F_{(G,\phi)}$, the elements f_1, f_2 and $f_1 \cdot f_2$ get mapped to the following products of generators;

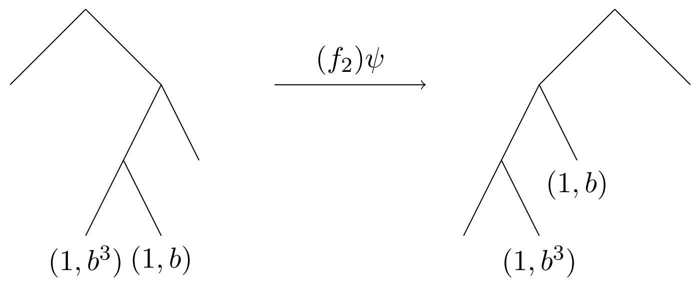
$$\begin{aligned}
(f_1)\psi &= X'_0X'_1\Gamma'_{g,2}{\Gamma'}^{-1}_{\theta(g),3}{X'_0}^{-1} \\
(f_2)\psi &= X'_1\Gamma'_{h,1}{\Gamma'}^{-1}_{\theta(h),2}{X'_0}^{-2} \\
(f_1 \cdot f_2)\psi &= X'_0X'_1X'_2\Gamma'_{gh,2}{\Gamma'}^{-1}_{\theta(gh),3}\Gamma'_{\theta(g),3}{\Gamma'}^{-1}_{\theta^2(g),4}{X'_0}^{-3}.
\end{aligned}$$

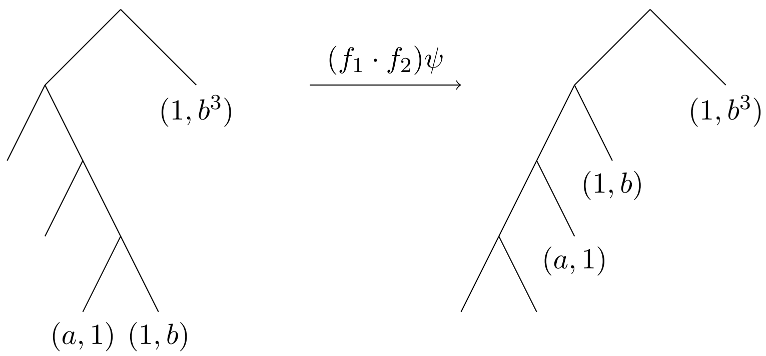
To find augmented tree pair representatives for $(f_1)\psi$, $(f_2)\psi$ and $(f_1 \cdot f_2)\psi$, we can multiply the reduced tree pair representatives of the products given above. Doing so will give the same domain and range trees as their equivalent elements f_1, f_2 and $f_1 \cdot f_2$ respectively, the difference will be in the decorations of each tree pair. We will refer to these decorations as γ_1, γ_2 and γ_3 respectively. After doing the tree pair multiplication one gets the following decorations on the resulting augmented tree pairs.

$$\begin{aligned}
\gamma_1 &= (1, 1, g, \phi(g)\theta(g)^{-1}) \\
\gamma_2 &= (1, h, \phi(h)\theta(h)^{-1}, \phi^2(h)\phi(\theta(h)^{-1})) \\
\gamma_3 &= (1, 1, gh, \phi(gh)\theta(gh)^{-1}\theta(g), \phi^2(gh)\phi(\theta(gh)^{-1})\phi(\theta(g))\theta^2(g)^{-1})
\end{aligned}$$

After resolving the homomorphisms and the products in the decorations above, one has the following augmented tree pair representatives for $(f_1)\psi$, $(f_2)\psi$ and $(f_1 \cdot f_2)\psi$.







To finish the example we multiply the augmented tree pairs of $(f_1)\psi$ and $(f_2)\psi$, and show that it is equal to the element $(f_1 \cdot f_)\psi$. As before, we already know that the domain and range trees for $(f_1)\psi \cdot (f_2)\phi$ will be the same as $(f_1 \cdot f_2)\psi$. The decoration, which we denote by $\gamma_{1,2}$, needs to be checked and is given by;

$$\gamma_{1,2} = (1, 1, gh, \phi(g)\phi(h)\theta(h)^{-1}, \phi(g)\theta(g)^{-1}\phi^2(h)\phi(\theta(h)^{-1})).$$

Which, after being resolved, becomes;

$$\gamma_{1,2} = (1, 1, (a, 1), (1, b), (1, b^3)).$$

Thus $\gamma_{1,2} = \gamma_3$ as expected and hence $(f_1 \cdot f_2)\psi = (f_1)\psi \cdot (f_2)\psi$.

5.3 Embedding results

In this final section we provide a proof for Remark 2.12 of [3]

Lemma 5.3 (based on Remark 2.12 of [3]). *Consider the group $V_{(G,\theta)}$ where θ is the identity endomorphism of G. Then $V_{(G,\theta)}$ embeds into Thompson's group V.*

Proof. Suppose $G = \{g_i\}_{i \in \boldsymbol{N}}$ is a finite group of order N, where g_1 is the identity element. Let $\mathcal{A} = \{a_1, a_2, \ldots, a_N\}$ be any antichain of $\mathcal{T}_2$ with N elements. Let $T_{\mathcal{A}}$ be the finite binary tree defined by the antichain $\mathcal{A}$. We fix these objects for the rest of the proof.

Let $\mathfrak{T}$ be the **set of all tree pairs** and $\mathfrak{A}$ be the **set of all augmented tree pairs**. We define a map $\pi : \mathfrak{A} \to \mathfrak{T}$ in the following way. Suppose $v = (\mathcal{D}, \gamma, \sigma, \mathcal{R})$ is an n-leaved augmented tree pair. Let $L_d := \{d_i\}_{i \in \boldsymbol{n}}$ be the antichain given by $\mathcal{D}$, and $L_r := \{r_i\}_{i \in \boldsymbol{n}}$ be the antichain given by $\mathcal{R}$. Let $\bar{\gamma} \in \boldsymbol{N}^n$ be the n-tuple such that $\bar{\gamma}(k) = i$ if and only if $\gamma(k) = g_i \in G$. Then $v\pi = (\widehat{\mathcal{D}}, \widehat{\sigma}, \widehat{\mathcal{R}})$, where $\widehat{\mathcal{D}}$, $\widehat{\mathcal{R}}$ and $\widehat{\sigma}$ are defined as the following;

- $\widehat{\mathcal{D}}$ is the binary tree created by attaching copies of the domain tree $\mathcal{D}$ to each leaf of the tree $T_{\mathcal{A}}$,
- $\widehat{\mathcal{R}}$ is the binary tree created by attaching copies of the range tree $\mathcal{R}$ to each leaf of the tree $T_{\mathcal{A}}$,
- The permutation $\widehat{\sigma}$ defines the map that take the leaf with address $a_i d_j$ in $\widehat{\mathcal{D}}$ to the leaf with address $a_{(i,\bar{\gamma}(j))\Delta} r_{(j)\sigma}$ in $\widehat{\mathcal{R}}$.

Although this looks quite complicated, one can visualise the tree pair $v\pi$ in the following way. Suppose $(d, g_j) \mapsto (r, g_j)$ is a leaf map defined by v. If $g_i g_j = g_k$ then $a_i d \mapsto a_k r$ is a prefix replacement rule for $v\pi$. An example is provided at the end of the chapter, illustrating the map π.

Suppose $[v]$ is an element of $V_{(G,\theta)}$ represented by the augmented tree pair $v = (\mathcal{D}, \gamma, \sigma, \mathcal{R})$. We construct a map $\pi^* : V_{(G,\theta)} \to V$ such that $[v]\pi^* = [v\pi]$.

We claim that the map π^* is an injective homomorphism. There are three steps involved. First we show that the map is well defined on the choice of augmented tree pair representative. Secondly we show that π^* is a homomorphism. Thirdly we prove that π^* is injective.

The map π^* is well defined

Augmented tree pairs in $\mathfrak{A}$ are equivalent under symmetric transitive closure of the splitting operation. We will show that π^* is well defined under the splitting operation and the result will follow. Suppose $[v]$ is an element of $V_{(G,\theta)}$ represented by the n-leaved augmented tree pair $v = (\mathcal{D}, \gamma, \sigma, \mathcal{R})$. Let $L_d := \{d_i\}_{i\in\boldsymbol{n}}$ be the antichain given by $\mathcal{D}$, and let $L_r := \{r_i\}_{i\in\boldsymbol{n}}$ be the antichain given by $\mathcal{R}$. Let $[v]\pi^* = [v\pi]$ be the image of $[v]$ under π^*, where $v\pi = (\widehat{\mathcal{D}}, \widehat{\sigma}, \widehat{\mathcal{R}})$. Let $v' = (\mathcal{D}', \gamma', \sigma', \mathcal{R}')$ be the augmented tree pair created by splitting the k^{th} leaf of v. As the endomorphism θ is the identity endomorphism, the decoration on both leaves of the newly created caret of v' is $\gamma(k)$. That is, $\gamma'(k) = \gamma'(k+1) = \gamma(k)$.

Let $v'\pi = (\widehat{\mathcal{D}'}, \widehat{\sigma'}, \widehat{\mathcal{R}'})$. The binary tree $\widehat{\mathcal{D}'}$ can be created by adding a caret to each leaf of $\widehat{\mathcal{D}}$ with address $a_i d_k$, for all $a_i \in \mathcal{A}$. Likewise, the binary tree $\widehat{\mathcal{R}'}$ can be created by adding a caret to each leaf of $\widehat{\mathcal{R}}$ with address $a_i r_k$, for all $a_i \in \mathcal{A}$. The prefix replacement rules associated to each of the carets $a_i d_k 0$ and $a_i d_k 1$ of $\widehat{\mathcal{D}'}$ are given by

$$
\begin{aligned}
a_i d_k 0 &\mapsto a_{(i,\bar{\gamma}(k))\Delta} r_k 0 \\
a_i d_k 1 &\mapsto a_{(i,\bar{\gamma}(k))\Delta} r_k 1,
\end{aligned}
$$

for all $a_i \in \mathcal{A}$. Thus each pair of leaves $\{a_i d_k 0, a_i d_k 1\}$ in $\widehat{\mathcal{D}}$ forms an exposed caret with the pair of leaves $\{a_{(i,\bar{\gamma}(k))\Delta} r_k 0, a_{(i,\bar{\gamma}(k))\Delta} r_k 1\}$ in $\widehat{\mathcal{R}}$. Thus one can create $v'\pi$ by using the splitting operation on the $(i+k)^{th}$ leaves of $v\pi$ for all $0 \le i \le (N-1)$. Therefore $[v\pi] = [v'\pi]$ and hence the map π^* is well defined on representative.

The map π^* is a homomorphism

We now show that π^* is a homomorphism. Suppose $[u]$ and $[v]$ are two elements of $V_{(G,\theta)}$ where $u = (\mathcal{D}_u, \gamma_u, \sigma_u, \mathcal{R}_u)$ and $v = (\mathcal{D}_v, \gamma_v, \sigma_v, \mathcal{R}_v)$ are n-leaved augmented tree pair representatives such that $\mathcal{R}_u = \mathcal{D}_v$. We consider the product $[u]\cdot[v]$ by using the leaf maps of u and v.

Suppose $L_d := \{d_1, d_2, \ldots, d_n\}$ is the antichain defined by $\mathcal{D}_u$, $L_s := \{s_1, s_2, \ldots, s_n\}$ is the antichain defined by $\mathcal{R}_u$ (and hence also $\mathcal{D}_v$), and $L_r := \{r_1, r_2, \ldots, r_n\}$ is the antichain defined by $\mathcal{R}_v$. The product of the augmented tree pairs $uv = u \cdot v$ is given by the following leaf maps, for all $i \in \boldsymbol{n}$,

$$(d_i, \gamma_{uv}(i)) \mapsto (r_{i(\sigma_u\cdot\sigma_v)}, \gamma_{uv}(i)),$$

where $\gamma_{uv}(i) = \gamma_u(i)\cdot\gamma_v((i)\sigma_u)$.

We will consider the images of $[u]$, $[v]$ and $[uv]$ under π^*. By definition of π, the element $[u]\pi^*$ is represented by the tree pair $u\pi$ defined by all the prefix replacement maps of the form

$$a_i d_j \mapsto a_{(i,\bar{\gamma}_u(j))\Delta} s_{(j)\sigma_u}, \tag{5.3}$$

for all $i \in \boldsymbol{N}$ and $j \in \boldsymbol{n}$. Likewise the image $[v]\pi^*$ is represented by the tree pair $v\pi$

defined by all the prefix replacement maps of the form

$$a_i s_j \mapsto a_{(i,\bar{\gamma}_v(j))\Delta} r_{(j)\sigma_v}, \tag{5.4}$$

for all $i \in \boldsymbol{N}$ and $j \in \boldsymbol{n}$. Finally the product $[uv]\pi^*$ is represented by the tree pair $uv\pi$ defined by all leaf maps of the form

$$a_i d_j \mapsto a_{(i,\bar{\gamma}_{uv}(j))\Delta} r_{j(\sigma_u \cdot \sigma_v)}, \tag{5.5}$$

for all $i \in \boldsymbol{N}$ and $j \in \boldsymbol{n}$.

By composition of (5.3) and (5.4), the product $(u)\pi \cdot (v)\pi$ is then represented by the tree pair defined by all the leaf maps of the form

$$a_i d_j \mapsto a_{((i,\bar{\gamma}_u(j))\Delta, \bar{\gamma}_v((j)\sigma_u))\Delta} r_{(j)\sigma_u \sigma_v} \tag{5.6}$$

for all $i \in \boldsymbol{N}$ and $j \in \boldsymbol{n}$. As group multiplication is associative we have

$$a_{((i,\bar{\gamma}_u(j))\Delta, \bar{\gamma}_v((j)\sigma_u))\Delta} = a_{(i,(\bar{\gamma}_u(j), \bar{\gamma}_v((j)\sigma_u))\Delta)\Delta}.$$

By the definition $\gamma_{uv}(j) = \gamma_u(j) \cdot \gamma_v((j)\sigma_u)$ we know that $\bar{\gamma}_{uv}(j) = \bar{\gamma}_u(j) \cdot \bar{\gamma}_v((j)\sigma_u)$ and therefore (5.6) becomes

$$a_i d_j \mapsto a_{(i,\bar{\gamma}_{uv}(j))\Delta} r_{j(\sigma_u \cdot \sigma_v)}. \tag{5.7}$$

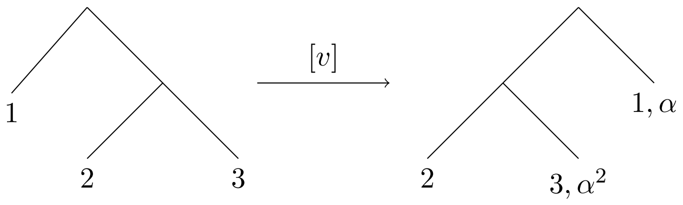
Therefore $[u]\pi^* \cdot [v]\pi^* = [uv]\pi^*$.

The map π^* is injective

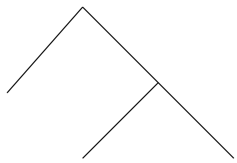
Finally we show that π^* is injective. Suppose $[v]$ is an element of $V_{(G,\theta)}$, represented by the augmented tree pair $v = (\mathcal{D}, \gamma, \sigma, \mathcal{R})$, such that $[v]\pi = 1_V$. Let $v\pi = (\widehat{\mathcal{D}}, \widehat{\sigma}, \widehat{\mathcal{R}})$. Then $\widehat{\mathcal{D}} = \widehat{\mathcal{R}}$ and hence $\mathcal{D} = \mathcal{R}$. Additionally, as $\widehat{\sigma}$ is the identity permutation on the leaves of $(\widehat{\mathcal{D}}, \widehat{\sigma}, \widehat{\mathcal{R}})$, σ must also be the identity permutation on the leaves of v. Furthermore, $(i,j)\Delta = i$ for all i, j, therefore $g_j = 1_G$ for all $g_j \in \gamma$. Putting all these consequences together gives us that $v = (\mathcal{D}, 1_{G^n}, 1_{S_n}, \mathcal{D})$ which represents the identity in $V_{(G,\theta)}$ and hence π^* is injective. $\square$

We finish the chapter by giving an example of an element of $V_{(G,\theta)}$ being embedded into V via the homomorphism π^*.

Example 5.4. Let $G = \langle \alpha \rangle$ be the cyclic group C_3 and θ be the identity endomorphism of C_3. Enumerate the elements in G as $g_1 = id_G$, $g_2 = \alpha$ and $g_3 = \alpha^2$. Consider the element $[v]$ of $V_{(G,\theta)}$ represented by the augmented tree pair v below.

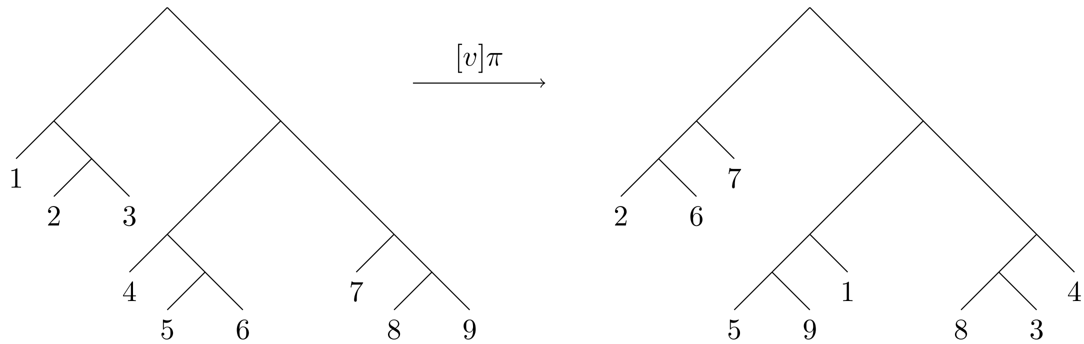


To implement the map π we must first define a complete antichain of $\mathcal{T}_2$ whose number of elements is equal to the order of G. There are two possibilities and we choose $\mathcal{A} = \{0, 10, 11\}$, drawn below.



We let $a_1 = 0$, $a_2 = 10$ and $a_3 = 11$. To create the domain tree for $v\pi$ we attach copies of the domain tree of v to the leaves of $T_{\mathcal{A}}$. Similarly, to create the range tree for $v\pi$ we attach copies of the range tree of v to the leaves of $T_{\mathcal{A}}$.

It is left to determine the bijection on the leaves of $v\pi$. Let us consider the first leaf. The address of this leaf is $a_1 0$, where a_1 is from the antichain $\mathcal{A}$, and the zero is the address of the first leaf in the domain tree of v. The tree pair v defines the leaf map $0\alpha \mapsto 1\alpha$, thus in $v\pi$ the leaf $a_1 0$ is mapped to the leaf $a_{(1,2)\Delta}1 = a_2 1 = 101$. We follow the method for the remainder of the leaves in $v\pi$ and we produce the following tree pair diagram.



The element $[v]\pi^*$ is then given by $[v\pi]$.

5.4 Future research

When considering counter examples to Lehnert's conjecture we have primarily focused our research on the groups discussed in [3], the augmented Thompson groups $V_{(G,\theta)}$ and $F_{(G,\theta)}$. The reason we have done so is two-fold. Firstly, they are a class of $co\mathcal{CF}$ groups which do not seem to arise from the four constructions of Holt, Rees, Röver and Thomas in [20] which we mentioned in Chapters 1-3. That is, there is no obvious way of taking the known $co\mathcal{CF}$ groups and creating groups from the class of $\mathfrak{V}_{aug}$ by using direct products or wreath products with $\mathcal{CF}$-groups. Secondly, the augmentation features that define the groups have no obvious dynamical representation in Thompson's group V. Using dynamics to prove that a group cannot embed into V is not a new approach. One can use dynamics to show that any infinite group which is finitely generated and torsion cannot embed into V, for example, the Grigorchuk group. Whilst groups from $\mathfrak{V}_{aug}$ do not satisfy this property, we believe that it is possible that a similar approach could be explored. The cases which we prove in this chapter to embed into V require strict conditions on both the finite groups used and the group endmorphisms we pair with them. By relaxing these conditions we believe it may be possible to find a counterexample to Lehnert's conjecture, that is, a group from $\mathfrak{V}_{aug}$ which cannot embed into V.

One possibility to consider would be the group $F_{(G,\theta)}$, where G is the abelian group C_3 and θ is the endomorphism which maps each non-identity element to the other. The endomorphism θ is not idempotent and therefore $F_{(G,\theta)}$ does not satisfy the conditions of Theorem 5.1. When considering the decoration on an augmented tree pair, each time a leaf that is decorated with a non-identity element is split, the new leaves are decorated with one of each non-identity element from C_3, the left leaf with one and the right leaf with the other. At the time of writing we were unable to find a natural way of expressing this relation using elements of V.

In summary, we propose that the class of groups $\mathfrak{V}_{aug}$ continues to be a source of plausible counterexamples to Lehnert's conjecture.

Bibliography

[1] Anatoly V. Anīsīmov. The group languages. *Kibernetika (Kiev)*, 4:18–24, 1971.

[2] Daniel Bennett and Collin Bleak. A dynamical definition of f.g. virtually free groups. *Internat. J. Algebra Comput.*, 26(1):105–121, 2016.

[3] Rose Berns-Zieze, Dana Fry, Johnny Gillings, Hannah Hoganson, and Heather Mathews. Groups with context-free co-word problem and embeddings into Thompson's group V. *arXiv:1407.7745v2*, December 2014.

[4] Jean-Camille Birget. The groups of Richard Thompson and complexity. *Internat. J. Algebra Comput.*, 14(5-6):569–626, 2004. International Conference on Semigroups and Groups in honor of the 65th birthday of Prof. John Rhodes.

[5] Jean-Camille Birget. The groups of Richard Thompson and complexity. *Internat. J. Algebra Comput.*, 14(5-6):569–626, 2004. International Conference on Semigroups and Groups in honor of the 65th birthday of Prof. John Rhodes.

[6] Collin Bleak. personal communication. June 2017.

[7] Collin Bleak, Francesco Matucci, and Max Neunhöffer. Embeddings into Thompson's group V and $co\mathcal{CF}$ groups. *J. Lond. Math. Soc. (2)*, 94(2):583–597, 2016.

[8] Collin Bleak and Martyn Quick. On small presentations of R. Thompson's group v. *in preparation*, 2015.

[9] Collin Bleak and Olga Salazar-Díaz. Free products in R. Thompson's group V. *Transactions of the American Mathematical Society*, 365(11):5967–5997, November 2013.

[10] Oleg Bogopolski. *Introduction to group theory.* EMS Textbooks in Mathematics. European Mathematical Society (EMS), Zürich, 2008. Translated, revised and expanded from the 2002 Russian original.

[11] José Burillo, Sean Cleary, Melanie Stein, and Jennifer Taback. Combinatorial and metric properties of Thompson's group T. *Trans. Amer. Math. Soc.*, 361(2):631–652, 2009.

[12] Jim W. Cannon, William J. Floyd, and William R. Parry. Introductory notes on Richard Thompson's groups. *Enseign. Math. (2)*, 42(3-4):215–256, 1996.

[13] Georg Cantor. Über unendliche, lineare punktmannigfaltigkeiten v. *Mathematische Annalen*, 21:545–591, 1883.

[14] N. Chomsky. Three models for the description of language. *IRE Transactions on Information Theory*, 2(3):113–124, September 1956.

[15] Pierre de la Harpe. *Topics in Geometric Group Theory*, chapter II.B, page 25. University of Chicago Press, 2000.

[16] M. Dehn. über unendliche diskontinuierliche Gruppen. *Math. Ann.*, 71(1):116–144, 1911.

[17] Patrick Dehornoy. Geometric presentations for Thompson's groups. *J. Pure Appl. Algebra*, 203(1-3):1–44, 2005.

[18] M. J. Dunwoody. The accessibility of finitely presented groups. *Invent. Math.*, 81(3):449–457, 1985.

[19] R. I. Grigorˇ cuk. On Burnside's problem on periodic groups. *Funktsional. Anal. i Prilozhen.*, 14(1):53–54, 1980.

[20] Derek F. Holt, Sarah Rees, Claas E. Röver, and Richard M. Thomas. Groups with context-free co-word problem. *J. London Math. Soc. (2)*, 71(3):643–657, 2005.

[21] John E. Hopcroft and Jeffery D. Ullman. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1979.

[22] Jörg Lehnert. *Gruppen von quasi-Automorphismen.* PhD thesis, Goethe Universität, Frankfurt, 2008.

[23] Jörg Lehnert and Paul Schweitzer. The co-word problem for the Higman-Thompson group is context-free. *Bull. Lond. Math. Soc.*, 39(2):235–241, 2007.

[24] David E. Muller and Paul E. Schupp. Groups, the theory of ends, and context-free languages. *Journal of Computer and System Sciences*, 26:295–310, 1983.

[25] David E. Muller and Paul E. Schupp. The theory of ends, pushdown automata, and second-order logic. *Theoretical Computer Science*, 37(1):51–75, 1985.

[26] Claas Röver. *Subgroups of finitely presented simple groups.* PhD thesis, Pembroke College, University of Oxford, 1999.

[27] Claas E. Röver. Constructing finitely presented simple groups that contain grigorchuk groups. *Journal of Algebra*, 220(1):284 – 313, 1999.

[28] Olga Patricia Salazar-Díaz. Thompson's group V from a dynamical viewpoint. *Internat. J. Algebra Comput.*, 20(1):39–70, 2010.

[29] Henry J. Stephen Smith. On the Integration of Discontinuous Functions. *Proc. London Math. Soc.*, S1-6:140–153, 1874/75.

[30] Slobodan Tanusevski. *Generalized Thompson groups.* ProQuest LLC, Ann Arbor, MI, 2014. Thesis (Ph.D.)–State University of New York at Binghamton.

[31] Richard J Thompson. Finitely presented groups of homeomorphisms. circulated handwritten notes, 1965.

[32] Walther von Dyck. Gruppentheoretische studien. *Mathematische Annalen*, 20(1):1–44, 1882.

[33] S. Witzel and M. C. B. Zaremsky. Thompson groups for systems of groups, and their finiteness properties. *ArXiv e-prints*, May 2014.

[34] M. C. B. Zaremsky. A user's guide to cloning systems. *ArXiv e-prints*, June 2016.