ALGEBRAIC PROPERTIES OF THE RICHARD THOMPSON'S GROUP F AND ITS APPLICATIONS IN CRYPTOGRAPHY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HAKAN YETER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MATHEMATICS

FEBRUARY 2021

Approval of the thesis:

ALGEBRAIC PROPERTIES OF THE RICHARD THOMPSON'S GROUP F AND ITS APPLICATIONS IN CRYPTOGRAPHY

submitted by **HAKAN YETER** in partial fulfillment of the requirements for the degree of **Master of Science in Mathematics Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Yıldıray Ozan
Head of Department, **Mathematics** _____

Assoc. Prof. Dr. Mustafa Gökhan Benli
Supervisor, **Mathematics Department, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Fatih Sulak
Mathematics Department, Atilim University _____

Assoc. Prof. Dr. Mustafa Gökhan Benli
Mathematics Department, METU _____

Assist. Prof. Dr. Burak Kaya
Mathematics Department, METU _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Hakan Yeter

Signature :

ABSTRACT

ALGEBRAIC PROPERTIES OF THE RICHARD THOMPSON'S GROUP F AND ITS APPLICATIONS IN CRYPTOGRAPHY

Yeter, Hakan
M.S., Department of Mathematics
Supervisor: Assoc. Prof. Dr. Mustafa Gökhan Benli

February 2021, 69 pages

Thompson's groups F, T and V, especially F, are widely studied groups in group theory. With their unique properties, they appear and become counterexamples for many general problems in different areas of mathematics. Developments in science and technology raise the importance of security for transmission and storage of private information, which can be provided by secure cryptosystems. In this thesis, we investigate algebraic properties of the Thompson's group F and its applications to cryptography.

Keywords: Piecewise Linear Homeomorphisms, Finitely Presented Groups, Simple Groups, Public Key Cryptography, Group Based Cryptography.

ÖZ

RİCHARD THOMPSON'IN GRUBU F'NİN CEBİRSEL ÖZELLİKLERİ VE KRİPTOGRAFİDEKİ UYGULAMALARI

Yeter, Hakan
Yüksek Lisans, Matematik Bölümü
Tez Yöneticisi: Doç. Dr. Mustafa Gökhan Benli

Şubat 2021 , 69 sayfa

Thompson'ın grupları F, T ve V, özellikle F, gruplar teorisi içinde çok önem arz eden gruplardandır. Bunlar kendilerine özgü benzersiz özellikleriyle matematiğin farklı alanlarında görülmekte olup bu alanlardaki birçok genel probleme karşı örnek verilebilirler. Bilim ve teknolojideki gelişmeler, güvenli kripto sistemlerle sağlanabilen gizli bilgilerin saklanması ve bu bilgilerin iletimindeki güvenliğin önemini arttırmıştır. Bu tezde, Thompson'ın grubu F'nin cebirsel özelliklerini ve kriptografideki uygulamalarını inceleyeceğiz.

Anahtar Kelimeler: Parçalı Lineer Homeomorfizmalar, Sonlu Sunulan Gruplar, Basit gruplar, Açık Anahtarlı Kriptografi, Grup Tabanlı Kriptografi.

To my family...

ACKNOWLEDGEMENTS

First of all, I owe my sincerest gratitude to my supervisor Mustafa Gökhan Benli for his endless support, detailed reviews, patience, motivation, and encouragement throughout the process of writing up this thesis. Without his patience and guidance, this study could not have been conducted. I had the chance to learn a lot from him throughout my graduate study.

Also, I have special thanks to Assoc. Prof. Dr. Mehmetcik Pamuk for his endless support and encouragement.

I would also like to thank all instructors, colleagues, and friends in the Mathematics Department of METU for teaching, sharing their broad knowledge and wide experience.

I wish to express my sincere thanks to my friends Oğuz Gözcü, Zehra Esin Dinçer, Erhan Ceylan, Mehmet Ali Batan, Ceren Karataş Batan, Yağmur Yılmaz, Kürşat Yılmaz, Recep Özkan, Derya Özkan, and Nazmi Oyar for their great support and encouragement.

Last but not least, I wish to express my very profound gratitude to my family for providing me with their unfailing love, support, and continuous encouragement throughout my whole life.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Richard Thompson introduced three groups, F, T, and V such that $F \subseteq T \subseteq V$ in some unpublished notes in 1965. While T and V are finitely presented infinite simple groups, F is not simple, but its commutator subgroup F' is simple. They were used by McKenzie and Thompson in [18] for constructing finitely presented groups with unsolvable word problems. These three groups have unusual properties, and therefore, are studied in many areas of mathematics. Especially, the group F plays an important role in infinite group theory related to amenability. The Von Neumann problem (answered negatively by Olshanskii in [24]) asks whether a nonamenable group necessarily contains a free group of rank bigger than or equal to 2. The group F does not contain such a free subgroup, and hence, whether F is amenable or not becomes important. This problem is one of the significant open problems in group theory.

Constructing secure cryptosystems makes use of different aspects of mathematics. Group based cryptography is a rather new research area aiming to build secure cryptosystems and key exchange protocols based on noncommutative groups. Thompson's group F is a suitable candidate for such cryptosystems.

This thesis will investigate the relations and applications of Thompson's group F to cryptography in two chapters.

In Chapter 2, we will first give some basic definitions and properties of F, such as tree representations, reducibility, and normal form of elements. Then, we will move on to the algebraic properties of F in Section 2.2. In this section, we will show that F is a finitely presented infinite group where the commutator subgroup of F consists

of elements, which are identity near 0 and 1, and F is just-nonabelian. Also, we will prove that F' is simple by using Higman's theorem [13]. Section 2.3 will investigate the word problem in F, and we will explain the algorithm of [27], which allows calculating normal forms of elements in F efficiently.

In Chapter 3, we will explain general notions of cryptography and its relations with group theory. Section 3.1 is devoted to public key cryptography and its rudiments. In Section 3.2, we will focus on group based cryptography. We will observe how non-commutative groups can be used in public key cryptography. We will then explain primary conditions for a group to be used as a platform group in cryptosystems. Also, we will give several key establishment protocols utilizing several search problems. In Section 3.3, we will analyze the protocol, suggested by V. Shpilrain and A. Ushakov [27], that uses F as a platform group. Finally, in Section 3.4, we will explore and explain the cryptanalysis of Matucci [17] against the protocol of Shpilrain and Ushokov [27]. This will show that the protocol is highly insecure.

CHAPTER 2

THOMPSON'S GROUP F AND SOME OF ITS PROPERTIES

2.1 Definition of F and Some Basic Properties

In this section, we will give some basic definitions and prove some properties of the Thompson's group F. We will use these in the coming sections. This section follows the standard reference [5] about F. The results proven in this section were taken again from [5].

Definition 1. *A homeomorphism $f : \mathbb{R} \to \mathbb{R}$ is piecewise linear if there are $x_1, x_2, \ldots,$ $x_n \in \mathbb{R}$ with $x_1 < x_2 < \cdots < x_n$ such that f is linear in each interval $(-\infty, x_1)$, $(x_1, x_2), \ldots, (x_{n-1}, x_n), (x_n, \infty)$. The real numbers $x_1, x_2, \ldots, x_n$ are the breakpoints of f.*

Definition 2. *The Thompson's group F is the group of piecewise linear homeomorphisms of the interval $[0, 1]$ with breakpoints at dyadic rational numbers, and the slope on each interval is an integer power of 2. (A real number of the form $\frac{a}{2^b}$ where $a, b \in \mathbb{Z}$ is called a dyadic rational number.)*

Derivatives are positive where they exist. So, the elements of F preserve orientation.

Example 1. *The function f is an element of F with the breakpoints at $x = \frac{1}{8}$ and $x = \frac{1}{2}$.*

$$f(x) = \begin{cases} 4x & 0 \leq x \leq \frac{1}{8} \\ x + \frac{3}{8} & \frac{1}{8} \leq x \leq \frac{1}{2} \\ \frac{x+3}{4} & \frac{1}{2} \leq x \leq 1 \end{cases}$$

In [2], Brown introduced the following representation of elements of F by tree diagrams.

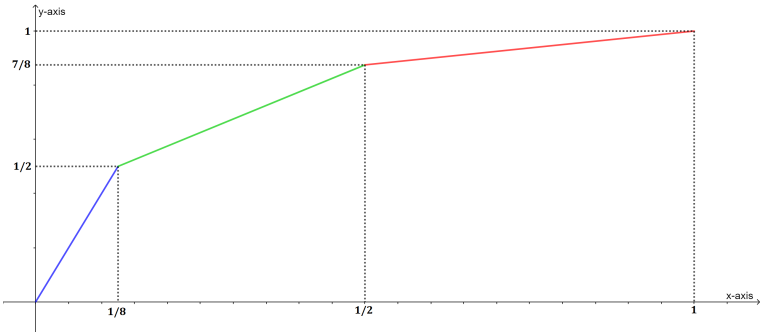


Figure 2.1: The graph of the element f

Definition 3. *A rooted tree is a graph with a specific vertex (called the root), which does not have any nontrivial cycles.*

A finite tree is called a rooted binary tree if
1) There is only one vertex with degree two (called root) and,
2) All the other vertices either are of degree one (called leaves) or are of degree three (called nodes).

If v is a node in the tree R, then there are precisely two edges $e_L^{\{v\}}$, $e_R^{\{v\}}$ which contain v and are not involved in the path from the root and v. The edge $e_L^{\{v\}}$ is called a left edge of R and $e_R^{\{v\}}$ is called a right edge of R. The left side (respectively, right side) of R is the maximal arc of the left edges (respectively, right edges) from the root.

There is a natural left-to-right linear ordering on the leaves of a rooted binary tree. Such a tree is called **"an ordered rooted binary tree"**.

Definition 4. *Let R be a rooted binary tree. A **caret** is a subtree of R consisting two leaves of R connected to a common vertex by an edge. (See Figure 2.3)*

Definition 5. *An interval of the form $[\frac{n}{2^m}, \frac{n+1}{2^m}]$ where $n, m \in \mathbb{Z}$ such that $n, m \geq 0$ with $n \leq 2^m - 1$ is called a standard dyadic interval in $[0, 1]$.*

The next lemma shows the interplay between dyadic partitions, rooted binary trees and elements of the group F.

Lemma 2.1.1 (Lemma 2.2, [5])**.** *For every element $f \in F$, there exists a standard dyadic partition $0 = x_0 < x_1 < x_2 < \cdots < x_n = 1$ such that f is linear on every*

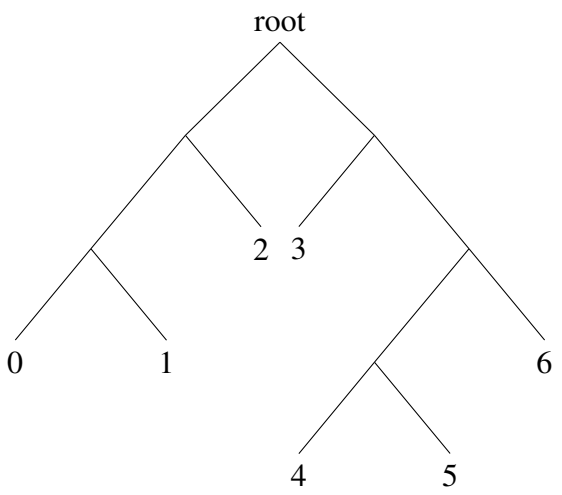


Figure 2.2: An ordered rooted binary tree

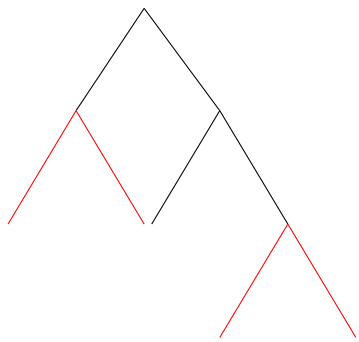


Figure 2.3: A rooted binary tree and two carets

interval of the partition and $0 = f(x_0) < f(x_1) < f(x_2) < \cdots < f(x_n) = 1$ *is a standard dyadic partiton.*

Proof. By the definition of elements of F, we can choose a partition P of the interval $[0, 1]$ such that the partition points are dyadic rationals and f is linear on these intervals. Let $[a, b]$ be an interval of P. Assume that the function f has the derivative 2^{-k} on the interval $[a, b]$ for some $k \in \mathbb{Z}$. In other words, on the interval $[a, b]$ $f(x) = 2^{-k}x + t$ for some dyadic rational number t. Now, we can find a non-negative integer m such that $m + k \geq 0$, $2^m a \in \mathbb{Z}$, $2^m b \in \mathbb{Z}$, $2^{m+k} f(a) \in \mathbb{Z}$, and $2^{m+k} f(b) \in \mathbb{Z}$. So, $a < a + \frac{1}{m} < a + \frac{2}{m} < \cdots < b$ partitions $[a, b]$ into standard dyadic intervals. Similarly, $f(a) < f(a) + \frac{1}{2^{m+k}} < f(a) + \frac{2}{2^{m+k}} < f(a) + \frac{3}{2^{m+k}} < \cdots < f(b)$ partitions $[f(a), f(b)]$ into standard dyadic intervals. $\square$

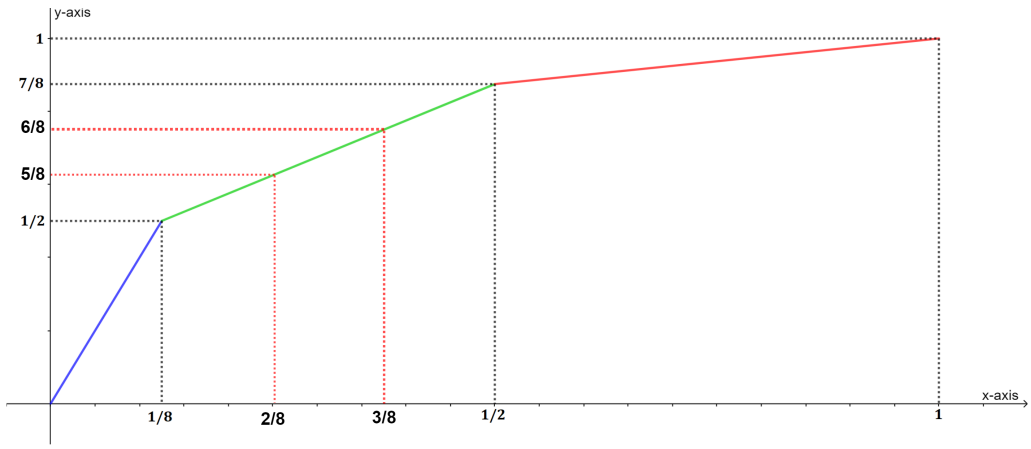


Figure 2.4: The element f of example 1 with a standard dyadic partition

In Figure 2.5, we show the infinite tree of standard dyadic intervals. A finite subtree with the root $[0, 1]$ of the tree of standard dyadic intervals is called a $\mathcal{T}$-tree.

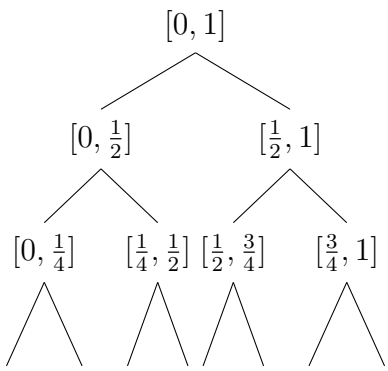


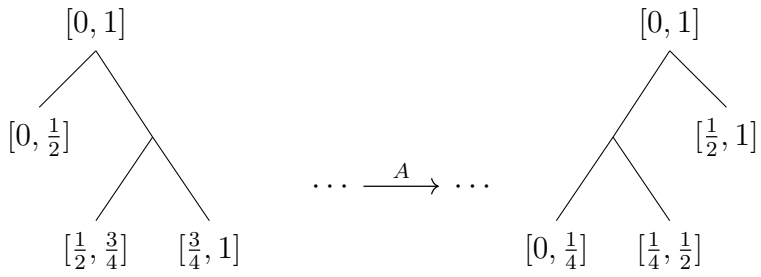
Figure 2.5: The infinite tree of standard dyadic intervals in $[0, 1]$

Definition 6. *Let R and S be two $\mathcal{T}$-trees with the same number of leaves. The ordered pair (R, S) is called a* ***tree diagram****. In this case, R is the domain tree, and S is the range tree.*

Example 2. *The function A is an element of F with breakpoints at $x = \frac{1}{2}$ and $x = \frac{3}{4}$.*

$$A(x) = \begin{cases} \frac{x}{2} & 0 \leq x \leq \frac{1}{2} \\ x - \frac{1}{4} & \frac{1}{2} \leq x \leq \frac{3}{4} \\ 2x - 1 & \frac{3}{4} \leq x \leq 1 \end{cases}$$

In this case, the domain tree R corresponds to the partition $\{0, \frac{1}{2}, \frac{3}{4}, 1\}$ and the range tree S corresponds to the partition $\{0, \frac{1}{4}, \frac{1}{2}, 1\}$. The function A maps the first interval



(a) the domain R (b) the range S

Figure 2.6: The tree diagram of the function A

$[0, \frac{1}{2}]$ in the domain to the first interval $[0, \frac{1}{4}]$ in the range. Similarly, A sends the intervals in R to the corresponding intervals in S according to the ordering on the leaves. In fact, this tree diagram is the unique tree diagram for the function A.

One can clearly see that if (P, Q) (respectively, (Q, R)) is a tree diagram for a function f (respectively, g) in F, then (P, R) is a tree diagram for the function $g \circ f$.

We see that any element of F can be represented using a tree diagram, i.e., an ordered pair of finite binary trees where the leaves are dyadic intervals and are linearly ordered. However, these tree diagrams are not always unique.

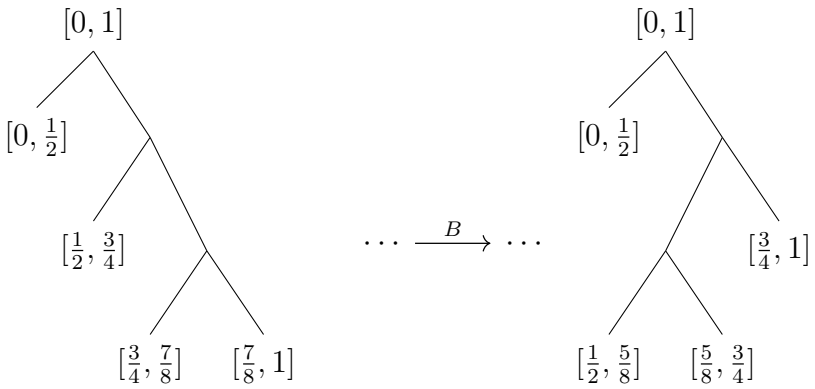
Let (R, S) be a tree diagram for an element $f \in F$. We can form another tree diagram by adding extra carets to the leaves in R and in S, where the leaves have the same label according to the ordering. Since f is linear on standard dyadic intervals, after adding extra carets, f sends the numbered leaves in R to the same numbered leaves in S. Hence, the resulting tree diagram becomes a new tree diagram for f.

Similarly, one can also remove redundant carets: Suppose there is a positive integer n so that both n^{th} and $(n+1)^{th}$ leaves appear as the vertices of the same caret in R, respectively in S, then eliminating all those carets but the roots from R and S gives a new tree diagram for the same function f. If there does not occur such carets in a tree diagram, then it is called **reduced**.

Hence, there exists a natural bijection between Thompson's group F and set of reduced tree diagrams.

Example 3. *The function B is an element of F with breakpoints at $x = \frac{1}{2}$, $x = \frac{3}{4}$ and $x = \frac{7}{8}$.*

$$B(x) = \begin{cases} x & 0 \leq x \leq \frac{1}{2} \\ \frac{x}{2} + \frac{1}{4} & \frac{1}{2} \leq x \leq \frac{3}{4} \\ x - \frac{1}{8} & \frac{3}{4} \leq x \leq \frac{7}{8} \\ 2x - 1 & \frac{7}{8} \leq x \leq 1 \end{cases}$$



(a) the domain tree (b) the range tree

Figure 2.7: Reduced tree diagram of the function B

Now, let's define elements $X_0, X_1, X_2, \ldots$ in F as follows: $X_0 = A, X_1 = B$ and $X_n = A^{-(n-1)}BA^{n-1}$ for $n \geq 1$. From Figure 2.6 and Figure 2.7, we can clearly see that diagram in Figure 2.8 is the tree representation of X_n.

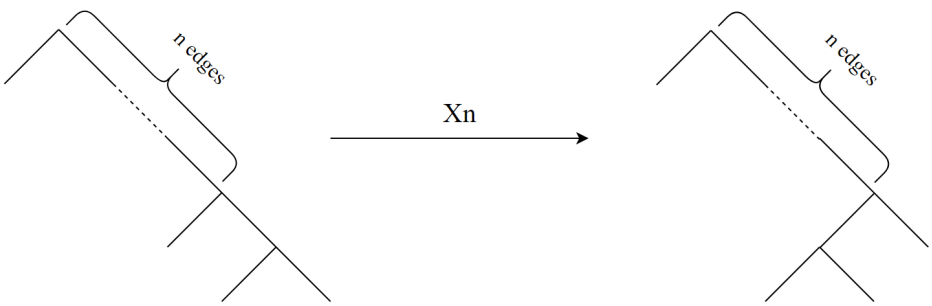


Figure 2.8: The reduced tree diagram for X_n

We will see that the sets $\{A, B\}$ and $\{X_0, X_1, X_2, \ldots\}$ are generating sets of the Thompson's group F. Moreover, we will see that each element in F has a unique normal form in terms of X_i's.

Also, observe that there is a geometric relation between the elements X_i for $i \geq 1$ and the element A. For example, the relation between $X_0 = A$ and $X_1 = B$ can be seen in Figure 2.9. Similarly, to obtain the element X_n, we simply take the identity function for the interval $[0, 1 - \frac{1}{2^n}]$ and we "compress" the function A into the interval $[1 - \frac{1}{2^n}, 1]$. See Figure 2.10 for the graphs of the functions X_2 and X_n.

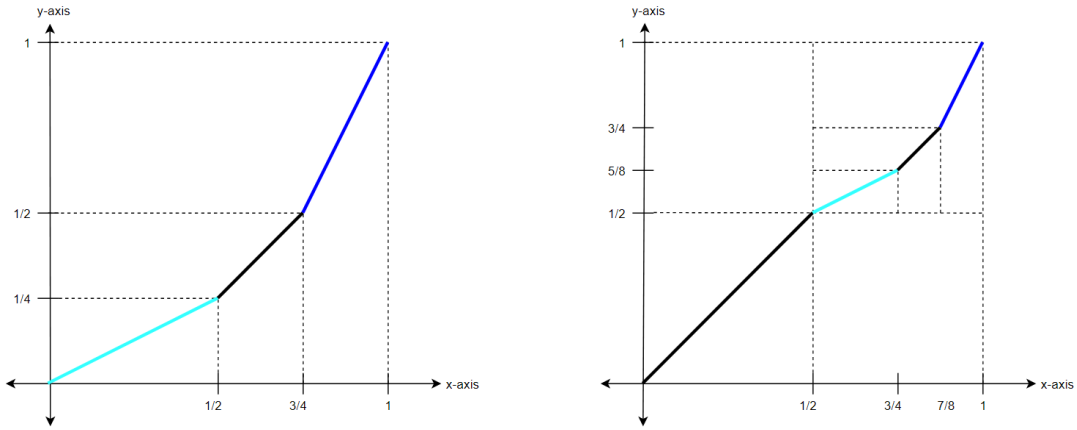


Figure 2.9: The elements A and B

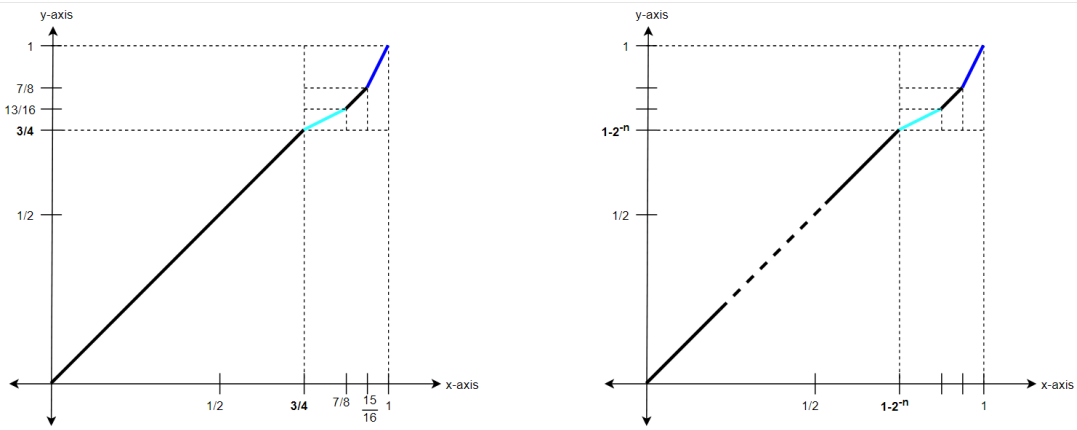


Figure 2.10: The elements X_2 and X_n

Definition 7. *Let $I_0, I_1, \ldots, I_n$ be the leaves of a $\mathcal{T}$-tree R in order. For any $k \in \mathbb{Z}$ with $0 \leq k \leq n$, let a_k be the maximal length of left edges starting from I_k which do not arrive at the right side of R. Then, a_k is called the* ***k^{th} exponent*** *of R.*

Example 4. *Let $\mathcal{T}$-tree R be given as in Figure 2.11. Then, the corresponding exponents of R are* $2, 0, 1, 0, 0, 1, 0, 0$.

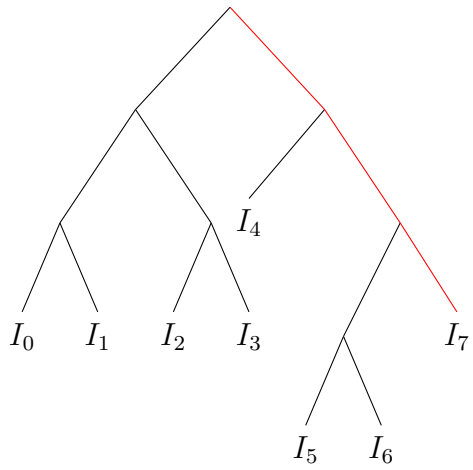


Figure 2.11: The $\mathcal{T}$-tree R

Definition 8. *For any integer $n \geq 0$, let $\mathcal{T}_n$ be the $\mathcal{T}$-tree with $n+1$ leaves whose right side has length n. Note that the domain tree of X_n is $\mathcal{T}_{n+2}$.*

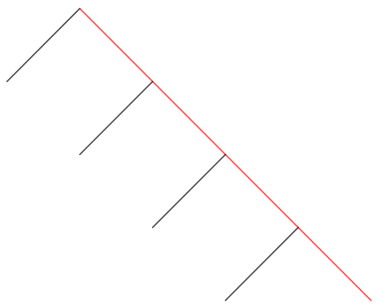


Figure 2.12: The $\mathcal{T}$-tree $\mathcal{T}_4$

Theorem 2.1.2 (Theorem 2.5, [5])**.** *Let R, S be $\mathcal{T}$-trees having $n+1$ leaves for some $n \geq 0$. Label the exponents of R as $a_0, a_1, \ldots, a_n$ and label the exponents of S as $b_0, b_1, \ldots, b_n$. Then, the element*

$$X_0{}^{b_0} X_1{}^{b_1} X_2{}^{b_2} \cdots X_n{}^{b_n} X_n{}^{-a_n} \cdots X_2{}^{-a_2} X_1{}^{-a_1} X_0{}^{-a_0}.$$

of F has tree diagram (R, S). The tree diagram (R, S) is reduced if and only if
$i)$ if the leaves I_{n-1} and I_n in R has the same root, then the leaf I'_{n-1} in S does not have the same root with I'_n, and

$ii)$ for any $k < n$ where $k \in \mathbb{Z}^+ \cup \{0\}$, if $a_k > 0$ and $b_k > 0$, then either $a_{k+1} > 0$ or $b_{k+1} > 0$.

Proof. For the first part of the theorem, it is enough to show that the element corresponding to the tree diagram $(R, \mathcal{T}_n)$ is

$$X_n{}^{-a_n} X_{n-1}{}^{-a_{n-1}} \cdots X_2{}^{-a_2} X_1{}^{-a_1} X_0{}^{-a_0}.$$

We do induction on $a = \sum_{i=0}^{n} a_i$, i.e., the exponent sum of R. For $a = 0$, then the element is identity function and $R = \mathcal{T}_n$. Now, for $a > 0$, let m be the smallest index so that $a_m > 0$. That is, R has exponents $a_i = 0$ for $0 \leq i \leq m-1$. So, R is a tree with a form like the one at the left of Figure 2.13. Let the $\mathcal{T}$-tree at the right of Figure 2.13 be given as R', where R'_1, R'_2, R'_3 and R_1, R_2, R_3 are isomorphic with each other as ordered rooted binary trees. As shown in Figure 2.8, the function with the tree diagram (R', R) is X_m. If $a'_0, a'_1, \ldots, a'_n$ are the exponents of R', then we have $a'_m = a_m - 1$ and $a'_k = a_k$ if $k \neq m$. Hence, by the induction hypothesis $(R', \mathcal{T}_n)$ is the tree diagram of the function $X_n{}^{-a'_n} \cdots X_1{}^{-a'_1} X_0{}^{-a'_0}$. Since $-a'_m = -a_m + 1$ and $-a'_k = -a_k$ if $k \neq m$, when we compose these two functions $X_m{}^{-1}$ and $X_n{}^{-a'_n} \cdots X_1{}^{-a'_1} X_0{}^{-a'_0}$, the element with tree diagram $(R, \mathcal{T}_n)$ becomes $X_n{}^{-a_n} \cdots X_2{}^{-a_2} X_1{}^{-a_1} X_0{}^{-a_0}$.

Now, we will prove the second part:
(i) Assume that the last two leaves, i.e., $(n-1)^{th}$ and n^{th} leaves of R and S lie in a caret. This means that there appears a caret at the right end of these trees. Thus, by definition of reducibility, this tree diagram is not reduced.
(ii) Assume that $a_k > 0, b_k > 0$, $a_{k+1} > 0$ and $b_{k+1} > 0$ for some integer k with $0 \leq k < n$. So, we have two consecutive leaves having nonzero exponents. Since the exponents of kth and $(k+1)^{th}$ leaves are nonzero and these are binary trees, we have $a_{k+2} = 0$. Thus, $(k+1)^{th}$ and $(k+2)^{th}$ leaves appear in the same caret. Hence, this diagram is not reduced.

For the other direction, if we suppose that the conditions (i) and (ii) are satisfied, then directly by definition of reducibility, we see that (R, S) is reduced. □

Corollary 2.1.3. *The elements A and B generate Thompson's group F.*

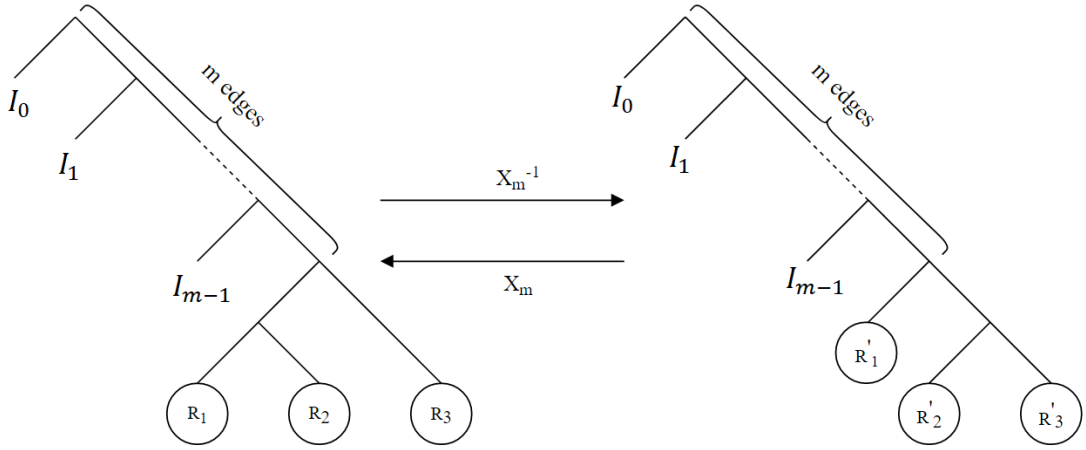


Figure 2.13: The $\mathcal{T}$-trees R and R'

Corollary 2.1.4. *Any nontrivial element of F can be written in a unique **normal form***

$$X_0{}^{b_0} X_1{}^{b_1} X_2{}^{b_2} \cdots X_n{}^{b_n} X_n{}^{-a_n} \cdots X_2{}^{-a_2} X_1{}^{-a_1} X_0{}^{-a_0}$$

where $n, a_0, \ldots, a_n, b_0, \ldots, b_n$ are nonnegative integers such that
$i)$ exactly one of a_n and b_n is nonzero, and
$ii)$ if both a_k and b_k are nonzero for some $k \in \mathbb{Z}$ with $0 \leq k < n$, then we have either $a_{k+1} > 0$ or $b_{k+1} > 0$. Moreover, any element of F in such a normal form is nontrivial.

As we shall see in Section 2.3, existence of the normal form provides a fast algorithm for the word problem in F.

2.2 Properties of F

In this section, we will prove various properties of F. Firstly, we will give two presentations of F and, in particular, prove that F is finitely presented. Next, we will prove results related to subgroups and quotients of F. As references in this section, we have [5], [13] and José Burillo, Introduction to Thompson's group F, available at: https://web.mat.upc.edu/pep.burillo/F%20book.pdf

We will define two groups F_1 and F_2 by generators and relations. We will then prove these are isomorphic to F. Let a, b be two elements of a group, then the commutator

of a and b is defined as $[a, b] = aba^{-1}b^{-1}$. The groups F_1 and F_2 are the following:

$$F_1 = \langle A, B \mid [AB^{-1}, A^{-1}BA], [AB^{-1}, A^{-2}BA^2] \rangle$$

$$F_2 = \langle X_0, X_1, X_2, \ldots \mid X_k^{-1}X_nX_k = X_{n+1} \; for \; k < n \rangle$$

Theorem 2.2.1 (Theorem 3.1, [5]). *F_1 and F_2 are isomorphic by an isomorphism sending A to X_0 and B to X_1.*

Proof. We will show that there is a homomorphism ϕ from F_1 to F_2 sending A to X_0 and B to X_1. Let F_3 be the free group generated by A and B. Let $\phi : F_3 \to F_2$ be the homomorphism with $\phi(A) = X_0$ and $\phi(B) = X_1$. Since $X_k^{-1}X_nX_k = X_{n+1}$ for $k < n$, if we fix $k = 0$ and $n = 1$, then we get that ${X_0}^{-1}X_1X_0 = X_2$. Similarly,

$${X_0}^{-1}X_2X_0 = X_3 \implies {X_0}^{-1}{X_0}^{-1}X_1X_0X_0 = {X_0}^{-2}X_1X_0^2 = X_3.$$

Assume that ${X_0}^{-(t-2)}X_1X_0^{t-2} = X_{t-1}$ for some $t \in \mathbb{N}$. Then,

$${X_0}^{-1}X_{t-1}X_0 = X_t \implies {X_0}^{-(t-1)}X_1X_0^{t-1} = X_t.$$

Hence, by induction, ${X_0}^{-(t-1)}X_1X_0^{t-1} = X_t$ for $t \geq 2$. So, for any $X_n \in F_2$, there is an element $A^{-(n-1)}BA^{n-1} \in F_3$ such that $\phi(A^{-(n-1)}BA^{n-1}) = X_n$. Thus, this homomorphism is surjective.

Also, we have ${X_1}^{-1}X_2X_1 = X_3 = {X_0}^{-1}X_2X_0$ since $X_k^{-1}X_nX_k = X_{n+1}$ for $k < n$. Hence,

$$\phi([AB^{-1}, A^{-1}BA]) = [\phi(AB^{-1}), \phi(A^{-1}BA)] = [X_0{X_1}^{-1}, X_2]$$

$$= X_0\underline{{X_1}^{-1}X_2X_1}{X_0}^{-1}{X_2}^{-1} = X_0\underline{{X_0}^{-1}X_2X_0}{X_0}^{-1}{X_2}^{-1} = 1.$$

Similarly, since ${X_1}^{-1}X_3X_1 = {X_0}^{-1}X_3X_0$, we have

$$\phi([AB^{-1}, A^{-2}BA^2]) = [\phi(AB^{-1}), \phi(A^{-2}BA^2)] = [X_0{X_1}^{-1}, X_3] = 1.$$

Therefore, we show that the defining relations of F_1 are in the kernel of ϕ.

Now, it is enough to show that there is a group homomorphism from F_2 to F_1 where X_0 is mapped to A, and X_1 is mapped to B. Let $Y_0 = A$ and $Y_n = A^{-(n-1)}BA^{n-1}$ for $n \geq 1$. It is sufficient to prove that

$$(i) \quad {Y_k}^{-1} Y_n Y_k = Y_{n+1} \text{ for } k < n \text{ and all } n \geq 1.$$

Consider the following:

$$(ii) \quad [A^{-1}B, Y_m] = 1 \text{ for } m \geq 3.$$

For $m = 3$, we know that $[AB^{-1}, A^{-1}BA] = 1$. So, $A^{-1}[AB^{-1}, A^{-1}BA]A = 1$. This implies that $[B^{-1}A, A^{-2}BA^2] = 1$. Since $[x, y] = 1$ implies $[x^{-1}, y] = 1$, we get $[A^{-1}B, A^{-2}BA^2] = [A^{-1}B, Y_3] = 1$. Similarly, as we have $[AB^{-1}, A^{-2}BA^2] = 1$, we get $[A^{-1}B, Y_4] = 1$.

Now, we show that if (ii) is true for $m = n-k+2$ then ${Y_k}^{-1} Y_n Y_k = Y_{n+1}$ for $k < n$ as follows:

$$\begin{aligned} Y_n Y_k &= A^{-(n-1)} B A^{n-1} A^{-(k-1)} B A^{k-1} = A^{-k+2} \underline{A^{-(n-k+1)} B A^{n-k+1}} A^{-1} B A^{k-1} \\ &= A^{-k+2} \underline{Y_{n-k+2} A^{-1} B} A^{k-1} = A^{-k+2} \underline{A^{-1} B Y_{n-k+2}} A^{k-1} \\ &= A^{-(k-1)} B A^{k-1} A^{-(k-1)} Y_{n-k+2} A^{k-1} = Y_k Y_{n+1} \end{aligned}$$

Hence, (i) holds for every $n \in \mathbb{Z}^+$ and $k = n - 1$ since (ii) is true for $m = 3$. In particular, ${Y_3}^{-1} Y_4 Y_3 = Y_5$. Since (ii) is satisfied when $m = 3$ and 4, i.e., $A^{-1}B$ commutes with Y_3 and Y_4, we have that $[A^{-1}B, Y_3^{-1} Y_4 Y_3] = [A^{-1}B, Y_5] = 1$, i.e., (ii) is valid for $m = 5$. A similar argument reveals that if $Y_4^{-1} Y_5 Y_4 = Y_6$, $[A^{-1}B, Y_4] = 1$, and $[A^{-1}B, Y_5] = 1$, then we get $[A^{-1}B, Y_6] = 1$. Therefore, by an inductive argument one sees $[A^{-1}B, Y_m] = 1$ for every $m \geq 3$, and hence (i) follows. $\square$

Suppose that $f : X \to X$ is a bijection, where X is a topological space. The closure of the set of the points in X that are not mapped to themselves is called the **support** of f, denoted by $supp(f)$. Since f is a bijection, we have

$$supp(f) = \overline{\{x \in X \mid f(x) \neq x\}} = supp(f^{-1}).$$

For bijections $f : X \to X$ and $g : X \to X$, we clearly have if $supp(f) \cap supp(g) = \varnothing$ then $f \circ g = g \circ f$.

Example 5. *The supports of the functions* AB^{-1}, $X_2 = A^{-1}BA$, $X_3 = A^{-2}BA^2$ *in* F *can be seen from Figure 2.14.*

$$supp(AB^{-1}) = [0, \frac{3}{4}], supp(X_2) = [\frac{3}{4}, 1] \text{ and } supp(X_3) = [\frac{7}{8}, 1].$$

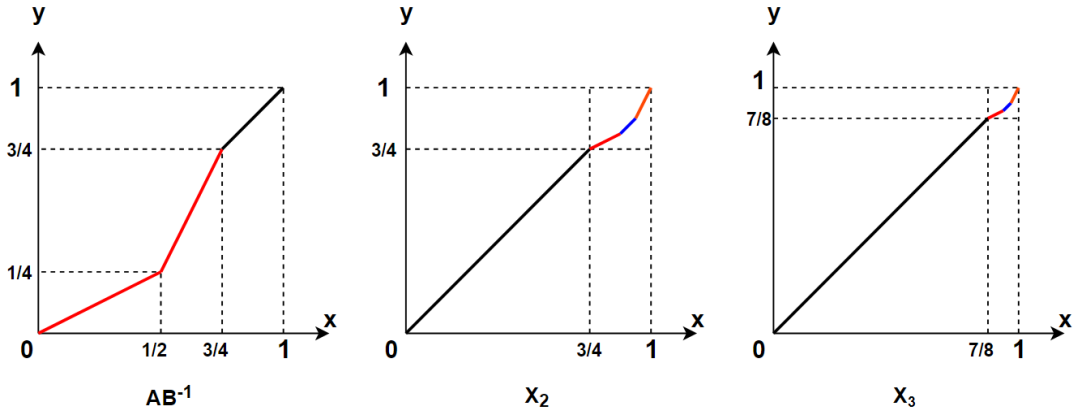


Figure 2.14: The functions AB^{-1}, X_2 and X_3

The elements in the form ${X_0}^{b_0}{X_1}^{b_1}{X_2}^{b_2}\cdots{X_n}^{b_n}$ with nonnegative b_i for $0 \leq i \leq n$ will be called **positive**. In the unique normal form, if we have all $a_i = 0$, then we have a positive element of F. So, the domain tree has all zero exponents, i.e., the domain tree is $\mathcal{T}_n$ for some nonnegative integer n. Inverses of positive elements in F are called **negative**.

Theorem 2.2.2 (Theorem 3.4, [5])**.** *The groups* F_1 *and* F_2 *are isomorphic to* F *by isomorphisms sending the formal symbols* $A, B, X_0, X_1, X_2, \ldots$ *in* F_1 *and* F_2 *to the corresponding elements in* F.

Proof. We know from Example 5 that intersection of the support of the function AB^{-1} in F with the support of $A^{-1}BA$, and $A^{-2}BA^2$ is the empty set. So, the functions $A^{-1}BA$ and $A^{-2}BA^2$ commute with AB^{-1}. Hence, the elements A, B in F satisfy the defining relations of F_1. Thus, we have a group homomorphism from F_1 to F, which sends the symbols A, B to the corresponding elements in F. Since F is generated by A and B, this homomorphism is onto. Hence, by Theorem 2.2.1, we have a surjective group homomorphism from F_2 to F, which sends the symbols $X_0, X_1, X_2, \ldots$ to the corresponding elements in F.

Now, it is enough to prove that this latter homomorphism is one-to-one. We know by Corollary 2.1.4 that any nontrivial element in F can be expressed uniquely as the product of a positive element and a negative element. From the defining relations of F_2, we have the following equalities:

$$X_k^{-1}X_n = X_{n+1}X_k^{-1}, \quad X_n^{-1}X_k = X_kX_{n+1}^{-1}, \quad X_nX_k = X_kX_{n+1} \quad for\ k < n.$$

These equalities show that any nontrivial element x in F_2 can be written as a product of a positive element and a negative element as in Corollary 2.1.4. If X_k appears in both the positive and negative part of x, but X_{k+1} does not appear in both, then by the equality $X_kX_{n+1}X_k^{-1} = X_n$ for $k < n$, we can eliminate X_k from both the positive part and the negative part, and we can change X_{n+1} with X_n. Hence, every nontrivial element of F_2 can be put in normal form as in Corollary 2.1.4. It follows from Corollary 2.1.4 that nontrivial elements of F_2 map to nontrivial elements of F. □

Definition 9. *Let G be any group and $a, b \in G$. The **commutator subgroup** of G is defined as $G' = [G, G] = \langle [g, h] \mid g, h \in G \rangle$. Clearly, $G' \lhd G$.*

Let $G = \langle X \rangle$ and $N \trianglelefteq G$. One can easily observe that $[x, y] \in N$ for all $x, y \in X$, then $G' \leq N$. Also, G/N is abelian if and only if $G' \leq N$.

Suppose $G = \langle x, y \rangle$. If $g = x^{k_1}y^{m_1} \cdots x^{k_n}y^{m_n} \in G$ with $\sum k_i = \sum m_i = 0$, then $q(g) = 0$ where $q : G \to G/G'$ is the quotient map. Thus, $g \in G'$.

It is also easy to observe that if a normal subgroup N contains $[x, y^{-1}]$, then $G' \leq N$.

The next theorem describes elements in F'.

Theorem 2.2.3 (Theorem 4.1, [5])**.** *The elements of F that are identity in a neighborhood of 0 and 1 constitute the commutator subgroup F'. Also, $F/F' \cong \mathbb{Z} \oplus \mathbb{Z}$.*

Proof. Define a map $\phi : F \to \mathbb{Z} \oplus \mathbb{Z}$ so that ϕ sends an element $f \in F$ to an element $(a, b) \in \mathbb{Z} \oplus \mathbb{Z}$ where the right-slope of f at 0 is 2^a, and the left-slope of f at 1 is 2^b. The composition of two elements in F results in the product of the slopes. Thus, ϕ is a homomorphism. Since $\phi(A) = (-1, 1)$, and $\phi(B) = (0, 1)$, ϕ is surjective.

Claim: $Ker\phi = F'$

Since $\mathbb{Z} \oplus \mathbb{Z}$ is abelian, we immediately obtain $F' \subseteq Ker\phi$. Conversely, let f be an arbitrary element of $Ker\phi$. Then if $f = A^{k_1}B^{l_1}A^{k_2}B^{l_2}\cdots A^{k_m}B^{l_m}$ we have

$$\phi(f) = (-\sum_{i=1}^{m} k_i,\ (\sum_{i=1}^{m} k_i) + (\sum_{i=1}^{m} l_i)) = (0,0).$$

Hence, $\sum_{i=1}^{m} k_i = 0$ and $\sum_{i=1}^{m} l_i = 0$. Thus $f \in F'$. □

The next lemma shows us a relation between dyadic rational intervals and an element of F'.

Lemma 2.2.4 (Theorem 3.2.3, [4]). *Let $0 = x_0 < x_1 < x_2 < \cdots < x_n = 1$ and $0 = y_0 < y_1 < y_2 < \cdots < y_n = 1$ be dyadic partitions of $[0,1]$. Then, there is an element $f \in F'$ such that $f(x_i) = y_i$ for any i. Moreover, if $[x_{i-1}, x_i] = [y_{i-1}, y_i]$ for some i, then f can be chosen as identity on the interval $[x_{i-1}, x_i]$.*

Proof. Suppose that the length of the interval $[x_{i-1}, x_i]$ is $a_i/2^{n_i}$ and the length of $[y_{i-1}, y_i]$ is $b_i/2^{m_i}$. Without loss of generality, suppose that $a_i \leq b_i$. Let us divide $[y_{i-1}, y_i]$ into b_i subintervals of length $1/2^{m_i}$. Also, let us divide the interval $[x_{i-1}, x_i]$ first into a_i subintervals of length $1/2^{n_i}$. In other words, $I_1, I_2, \ldots, I_{a_i}$ are the subintervals of $[x_{i-1}, x_i]$ where $|I_j| = 1/2^{n_i}$ for every $j = 1, 2, \ldots, a_i$. Observe that if we take an interval I_k for some $k = 1, 2, \ldots, a_i$, and divide it into halves, we obtain two subintervals, say I_{k_1}, I_{k_2}, of I_k with the length $1/2^{n_i+1} = 2^{-(n_i+1)}$. So, dividing an interval into two halves gives us one more interval. Also, with this division again, each subinterval has length a power of 2. Therefore, if we divide $(b_i - a_i)$-many intervals from $I_1, I_2, \ldots, I_{a_i}$, we totally get b_i subintervals of $[x_{i-1}, x_i]$ with length a power of 2. Hence, we can construct a linear map from each subinterval of $[x_{i-1}, x_i]$ to each subinterval of $[y_{i-1}, y_i]$. This procedure gives us a piecewise linear map with breakpoints at dyadic rational numbers, and the slope on each interval is a power of 2. By applying this procedure to all intervals $[x_{i-1}, x_i]$ for $i = 1, 2, \ldots, n$, we obtain an element $f \in F$ such that $f(x_i) = y_i$ for $i = 0, 1, 2, \ldots, n$. If we rearrange the first and the last intervals so that f becomes identity near 0 and 1, then we have $f \in F'$. □

Definition 10. *Let G be a group. G is called **just-nonabelian** if G is nonabelian, and G/N is abelian for every nontrivial normal subgroup of G.*

Theorem 2.2.5 (Theorem 4.3, [5]). *F is just-nonabelian.*

Proof. First, we will show that the center of F is trivial. Let f be an element of the center of F. So, f commutes with every element of F. The fixed point set of f is $Fix(f) := \{x \in [0,1] \mid f(x) = x\}$. For any g in F, since $fg = gf$, we have that for every $x \in Fix(g)$, we have $f(x) \in f(Fix(g))$, and $f(g(x)) = g(f(x))$. Therefore, $f(x) = g(f(x))$. So, we have $f(x) \in Fix(g)$. Hence, we get $f(Fix(g)) \subseteq Fix(g)$. Conversely, suppose that $x \notin f(Fix(g))$. Then, $f^{-1}(x) \notin Fix(g)$. So, we have $g(f^{-1}(x)) \neq f^{-1}(x)$. Since f is in the center, f^{-1} is also in the center. Hence, we get $f^{-1}(g(x)) \neq f^{-1}(x)$. Therefore, $g(x) \neq x$. Thus, $x \notin Fix(g)$. So, we obtain $Fix(g) \subseteq f(Fix(g))$. Thus, we have shown that $f(Fix(g)) = Fix(g)$. In other words, f stabilizes the fixed point set of any element g. We know that the fixed point set of B is

$$Fix(B) = \{x \in [0,1] \mid B(x) = x\} = [0, \frac{1}{2}] \cup \{1\}.$$

So, if we take $g = B$, we get $f([0, \frac{1}{2}] \cup \{1\}) = [0, \frac{1}{2}] \cup \{1\}$. Hence, $f([0, \frac{1}{2}]) = [0, \frac{1}{2}]$. This shows that $f(\frac{1}{2}) = \frac{1}{2}$. We already know by definition, $f(0) = 0$ and $f(1) = 1$. So, we have $\{0, \frac{1}{2}, 1\} \subseteq Fix(f)$. For any dyadic rational k, there is $h \in F$ such that $h(\frac{1}{2}) = k$. Since $fh = hf$, $f(h(\frac{1}{2})) = h(f(\frac{1}{2})) = h(\frac{1}{2})$. Therefore, we get $f(k) = k$. So, f fixes every dyadic rational number in $[0,1]$. Since dyadic rational numbers are dense in $[0,1]$ and f is continuous, f must be the identity function.

Next, let N be a nontrivial normal subgroup of F, and we will show that F/N is abelian. Since the center is trivial and $\{1\} \neq N \lhd F$, there exist $g \in N$, $g \neq e$, and $h \in F$ such that $gh \neq hg$. This implies that $1 \neq ghg^{-1}h^{-1} \in N$ as N is normal. So, N contains a nontrivial commutator, call it f. Suppose that f has the normal form $f = X_0^{b_0} X_1^{b_1} X_2^{b_2} \cdots X_n^{b_n} X_n^{-a_n} \cdots X_2^{-a_2} X_1^{-a_1} X_0^{-a_0}$. Recall the map from the proof of Theorem 2.2.3 that $\phi : F \to \mathbb{Z} \oplus \mathbb{Z}$ such that $\phi(A) = (-1, 1)$ and $\phi(B) = (0, 1)$. Since $Ker\phi = [F, F]$, $f \in [F, F]$ and so $\phi(f) = (0, 0)$. Hence,

$$\phi(f) = \phi(X_0^{b_0} X_1^{b_1} X_2^{b_2} \cdots X_n^{b_n} X_n^{-a_n} \cdots X_2^{-a_2} X_1^{-a_1} X_0^{-a_0}) = (0, 0).$$

Since ϕ is a homomorphism and $X_n = A^{-(n-1)} B A^{n-1}$ for $n \geq 1$, we get that $\phi(A^{k_0}) = (-k_0, k_0)$ and $\phi(B^{l_0}) = (0, l_0)$. So,

$$\phi(X_n) = \phi(A^{-(n-1)}) + \phi(B) + \phi(A^{(n-1)}) = \phi(B)$$

because of $(n-1,-(n-1))+(0,1)+(-(n-1),n-1)=(0,1)$. Hence, we have $\phi(X_n^{b_i})=\phi(B^{b_i})=(0,b_i)$. Therefore,

$$(0,0)=\phi(f)=(-b_0,b_0)+(0,b_1)+\cdots+(0,b_n)+(0,-a_n)+\cdots+(0,-a_1)+(a_0,-a_0).$$

So, we have $(0,0)=(a_0-b_0,\sum_{i=0}^{n}(b_i-a_i))$, which implies that $a_0=b_0$. Let k be the smallest index such that $a_k\neq b_k$. Without loss of generality, suppose $b_k>a_k$. So,

$$f=X_0^{a_0}X_1^{-a_1}\cdots X_{k-1}^{a_{k-1}}X_k^{b_k}\cdots X_n^{b_n}X_n^{-a_n}\cdots X_{k+1}^{-a_{k+1}}\underline{X_k^{-a_k}\cdots X_1^{a_1}X_0^{-a_0}}.$$

Take $h^{-1}=X_k^{-a_k}\cdots X_2^{-a_2}X_1^{-a_1}X_0^{-a_0}$. Then, $h^{-1}fh\in N$ since N is normal, and

$$h^{-1}fh=X_k^{b_k-a_k}\cdots X_n^{b_n}X_n^{-a_n}\cdots X_{k+1}^{-a_{k+1}}.$$

We replace f by $h^{-1}fh$ above to get new f as

$$f=X_k^{b_k}X_{k+1}^{b_{k+1}}\cdots X_n^{b_n}X_n^{-a_n}\cdots X_{k+2}^{-a_{k+2}}X_{k+1}^{-a_{k+1}}.$$

So, we can assume that $b_0=b_1=\cdots=b_{k-1}=0,\ a_0=a_1=\cdots=a_{k-1}=a_k=0$ and $b_k>0$. Now, by using the identities $X_kX_{n+1}=X_nX_k$ and $X_{n+1}X_k^{-1}=X_k^{-1}X_n$ for $k<n$, we change f by

$$X_0^{k-1}fX_0^{-(k-1)}=X_0^{k-1}X_k^{b_k}\cdots X_n^{b_n}X_n^{-a_n}\cdots X_{k+1}^{-a_{k+1}}X_0^{-(k-1)}.$$

We have new $f=X_1^{b_k}X_2^{b_{k+1}}\cdots X_{n-k+1}^{b_n}X_{n-k+1}^{-a_n}\cdots X_3^{-a_{k+2}}X_2^{-a_{k+1}}$. So, we can assume that

$$f=X_1^{b_1}X_2^{b_2}\cdots X_{n-k+1}^{b_{n-k+1}}X_{n-k+1}^{-a_{n-k+1}}\cdots X_3^{-a_3}X_2^{-a_2},\ \text{and } a_0=a_1=b_0=0 \text{ and } b_1>0.$$

In this case, by using the identities

$$X_k^{-1}X_n=X_{n+1}X_k^{-1} \text{ and } X_n^{-1}X_k=X_kX_{n+1}^{-1} \text{ for } k<n,$$

we obtain

$$(X_0^{-1}fX_0)(X_1^{-1}fX_1)^{-1}=X_2^{b_1}X_1^{-b_1}.$$

Hence, N contains $X_1^{-b}(X_2^bX_1^{-b})X_1^b=X_1^{-b}X_2^b$ for some positive integer b. Therefore, we have

$$X_0X_2^{b-1}\underline{[X_2,X_1^{-b}X_2^b]}X_2^{-(b-1)}X_0^{-1}=X_0X_2^{b-1}\underline{(X_2X_1^{-b}X_2^bX_2^{-1}X_2^{-b}X_1^b)}X_2^{-(b-1)}X_0^{-1}$$

$$X_0X_2^{b-1}(X_2X_{b+2}^{-1})X_2^{-(b-1)}X_0^{-1} = X_0X_2X_3^{-1}X_0^{-1} = X_1X_2^{-1}$$
$$= BA^{-1}B^{-1}A = [B, A^{-1}] \in N.$$

Thus, we get $F^{'} \leq N$ since N contains $[B, A^{-1}]$, and hence, F/N is abelian. □

Lemma 2.2.6 (Lemma 4.4, [5]). *Given two dyadic rational numbers a, b with $0 \leq a < b \leq 1$ such that $b - a = 2^k$ for some k. All functions in F with support in $[a, b]$ constitute a subgroup denoted by $F_{[a,b]}$. Moreover, F and $F_{[a,b]}$ are isomorphic by an isomorphism sending any $f \in F$ to a conjugate in $F_{[a,b]}$.*

Proof. Define the linear homeomorphism $\phi : [a, b] \to [0, 1]$ by $\phi(x) = \frac{1}{b-a}x - \frac{a}{b-a}$. Since a, b are dyadic rational numbers and $b-a$ is a power of 2, ϕ maps dyadic rational numbers to dyadic rational numbers. Therefore, the inverse $\phi^{-1} : [0, 1] \to [a, b]$ defined by $\phi^{-1}(x) = x(b - a) + a$ also sends dyadic rational numbers to dyadic rational numbers. Now, define the isomorphism such that for every $f \in F$, it takes f to $\phi^{-1}f\phi \in F_{[a,b]}$. Since f, ϕ and ϕ^{-1} map dyadic rational numbers to dyadic rational numbers, so does $\phi^{-1}f\phi$. Also, by Chain Rule,

$$(\phi^{-1}f\phi)^{'}(x) = [\phi^{-1}(f(\phi(x)))]^{'}[f^{'}(\phi(x))][\phi^{'}(x)]$$
$$= (b - a)(f^{'}(\phi(x)))(\frac{1}{b - a}) = f^{'}(\phi(x)), \text{ where it exists.}$$

Next, we need to show that $\phi^{-1}f\phi$ has breakpoints at dyadic rational numbers. Suppose $\phi(m) = n$ for some dyadic rational number n. Then, $\phi(m) = \frac{m-a}{b-a} = n$ implies $m = n(b - a) + a$. Since $a, (b - a)$ and n are dyadic rationals, m is also a dyadic rational. Therefore, $\phi^{-1}f\phi$ has breakpoints at dyadic rationals since $f \in F$ has breakpoints at dyadic rational numbers. Also, $\phi^{-1}f\phi$ has a slope of a power of 2 when $f^{'}(\phi)$ exists. Because f is linear, and f has a slope of a power of 2. Thus, the result follows. □

In this section, we will prove that $F^{'}$ is a simple group. We will use a criterion of Higman from [13] and follow ideas of [4].

Let G be a permutation group of a set X. For any $\alpha \in G$, the **support** of α is $supp(\alpha) = \{x \in X \mid \alpha(x) \neq x\}$. Regarding the support of an element, we have the following:

1. $supp(\alpha) = supp(\alpha^{-1})$

2. $\alpha = \beta\gamma \implies supp(\alpha) \subseteq supp(\beta) \cup supp(\gamma)$

3. $supp(g\alpha g^{-1}) = g(supp(\alpha))$

As an example, we will prove (3):
Assume that $x \in supp(g\alpha g^{-1})$. Then, by definition of support of a function, we get $g\alpha g^{-1}(x) \neq x$. If we compose from left by g^{-1}, we see $\alpha(g^{-1}(x)) \neq g^{-1}(x)$. So, $g^{-1}(x) \in supp(\alpha)$. This implies that $x \in g(supp(\alpha))$. Conversely, suppose by contrapositive that $x \notin supp(g\alpha g^{-1})$. Then, again by definition, we obtain $g\alpha g^{-1}(x) = x$. This implies that $\alpha(g^{-1}(x)) = g^{-1}(x)$. So, we get $g^{-1}(x) \notin supp(\alpha)$. Hence, $x \notin g(supp(\alpha))$.

Definition 11. *Let $Y \subseteq X$ and $\alpha \in G$. If the sets Y and $\alpha(Y)$ are disjoint, then we say* ***α moves Y****.*

Theorem 2.2.7 (Theorem 1, [13])**.** *Let $a, b, f (f \neq 1)$ be elements of G. If there exists an element g such that f moves $g(supp(a) \cup supp(b))$, then G' is simple.*

Proof. Firstly, we will deduce that if $1 \neq N \trianglelefteq G$, then $G' \leq N$. Let nonidentity $f \in N$ and $a, b \in G$. Suppose that there exists an element $g \in G$ such that f moves $g(supp(a) \cup supp(b))$. Since f is an element in the normal form, we have $g^{-1}fg$, say h, is in the normal as well. Since $f(g(supp(a) \cup supp(b))) \cap g(supp(a) \cup supp(b)) = \varnothing$, clearly we have $f(g(supp(a))) \cap g(supp(b)) = \varnothing$. So, if we compose from left by g^{-1}, we get $h(supp(a)) \cap supp(b) = \varnothing$. Since $h(supp(a)) = supp(hah^{-1})$, we see that hah^{-1} and b commute. If x and y commute, then x and y^{-1} commute too. Therefore, we obtain $hah^{-1}b^{-1}ha^{-1}h^{-1}b = 1$, which is equivalent to $hah^{-1}b^{-1}ha^{-1}h^{-1} = b^{-1}$. Hence, we indicate that

$$[a^{-1}, b^{-1}] = a^{-1}\underline{b^{-1}}ab = a^{-1}\underline{hah^{-1}b^{-1}ha^{-1}h^{-1}}ab$$

$$= a^{-1}hah^{-1}b^{-1}h\underline{bb^{-1}}a^{-1}h^{-1}ab = \underline{a^{-1}ha}h^{-1}\underline{b^{-1}hb}\,\underline{b^{-1}a^{-1}h^{-1}ab}.$$

Since N is a normal subgroup of G and $h \in N$, we acquire that $a^{-1}ha, b^{-1}hb$, and $a^{-1}h^{-1}a$ are elements in N. If $a^{-1}h^{-1}a \in N$, then we get $b^{-1}a^{-1}h^{-1}ab \in N$, too. Therefore, $[a^{-1}, b^{-1}] \in N$, and this implies $[a, b^{-1}] = [a, hah^{-1}b^{-1}ha^{-1}h^{-1}] \in N$. Thus, since for any $a, b \in G$, $[a, b] \in N$, we have that $G' \leq N$.

Next, we will show that $G^{''} = G^{'}$ for nontrivial $G^{'}$. There are two cases depending on whether $G^{''}$ is trivial or not:

Case 1: $G^{''} \neq 1$

We know that $1 \neq G^{''} \trianglelefteq G$. So, the previous result shows that $G^{'} \leq G^{''}$. Thus, we have $G^{''} = G^{'}$.

Case 2: $G^{''} = 1$

In this case, we choose the elements above as $1 \neq a = b = f \in G^{'}$. Then, there exists an element g such that $h = g^{-1}fg \in G^{'}$. Similar to above, we have that $supp(hah^{-1}) \cap supp(b) = \varnothing$. This implies that $supp(hfh^{-1}) \cap supp(f) = \varnothing$. Since they have different supports, they cannot be the same element. Hence, $hfh^{-1} \neq f$. Therefore, $1 \neq hfh^{-1}f^{-1} = [h, f]$. As $h, f \in G^{'}$, we obtain $1 \neq [h, f] \in G^{''}$. This contradicts to $G'' = 1$. Thus, $G^{''} = G^{'}$.

Now, we will prove that assumptions for G are true for $G^{'}$. Let $a, b, f \in G$ with $f \neq 1$. There exists $g \in G$ such that

$$f(g(supp(a) \cup supp(b))) \cap g(supp(a) \cup supp(b)) = \varnothing. \quad (\star)$$

So, we can understand from $(\star)$ that f moves $g(supp(a) \cup supp(b))$. Therefore, $(\star)$ implies that $g(supp(a) \cup supp(b)) \subseteq supp(f)$. Also, if we take previously given $a, b, f \in G$, respectively as $f, g, f \in G$ with $f \neq 1$ now, then there exists $m \in G$ such that

$$f(m(supp(f) \cup supp(g))) \cap m(supp(f) \cup supp(g)) = \varnothing.$$

Therefore, when we put $n = m^{-1}fm$, we obtain that

$$supp(f) \cap supp(ngn^{-1}) = \varnothing. \quad (\triangle)$$

Hence, $(\triangle)$ implies that ngn^{-1} is trivial on $supp(f)$. Otherwise, the intersection in $(\triangle)$ would not be the empty set. So, if we combine $(\star)$ and $(\triangle)$, we obtain that ngn^{-1} is trivial on $g(supp(a) \cup supp(b))$, and also is the inverse $ng^{-1}n^{-1}$. Hence, we have $ng^{-1}n^{-1}g(supp(a) \cup supp(b)) = g(supp(a) \cup supp(b))$. Therefore, if we call $ng^{-1}n^{-1}g = k$, then we clearly see that $[n, g^{-1}] = k \in G^{'}$ and by $(\star)$

$$f(k(supp(f) \cup supp(g))) \cap k(supp(f) \cup supp(g)) = \varnothing.$$

In other words, we can select $k \in G^{'}$ instead of $g \in G$.

Finally, we will show that G' is simple. Let $1 \neq N \trianglelefteq G'$. Then, we have $G'' \leq N$ by the first part of the proof. This implies that $G' \leq N$ since $G'' = G'$. Thus, $G' = N$, which means that G' is simple. $\square$

Lemma 2.2.8 (Proposition 3.3.2, [4]). *For Thompson's group F, we have $F' = F''$ where $F'' = [F', F']$.*

Proof. Let $f \in F'$, and let $supp(f) \subseteq [a, b]$. We have $f \in F_{[a,b]}$ where $0 < a, b < 1$ since F' consists of all elements of F that are identity near 0 and 1, by Theorem 2.2.3. Choose two dyadic rational numbers c and d so that $0 < c < a$ and $b < d < 1$. So, we have $f \in F_{[a,b]} \subset F_{[c,d]} \subset F'$. Since we know that $F_{[c,d]}$ is isomorphic to F by Lemma 2.2.6, we obtain $F_{[a,b]} \subset F'_{[c,d]}$. Hence, we have $f \in F'_{[c,d]}$. Since $F_{[c,d]} \subset F'$, we observe that $F'_{[c,d]} \subset F''$. Therefore, $f \in F'_{[c,d]} \subset F''$. Thus, we have $F'' = F'$. $\square$

Theorem 2.2.9 (Theorem 4.5, [5]). *The subgroup F' is simple.*

Proof. We will apply Higman's theorem, Theorem 2.2.7, to the group F'. Since the elements in F' have their supports strictly inside $[0, 1]$, this allows us to move them into a smaller interval, and hence we can apply Higman's condition.

Let $a, b \in F'$. So, we have $supp(a) \cup supp(b) \subsetneq [\epsilon, 1-\epsilon]$ for some $\epsilon > 0$. Let $f \in F'$ and $f \neq 1$. Since $f \neq 1$, there exists an interval I such that $f(I) \cap I = \varnothing$. Therefore, by Lemma 2.2.4, we can choose an element $g \in F'$ which takes $supp(a) \cup supp(b)$ inside I. This satisfies Higman's condition since $f(I) \cap I = \varnothing$. Hence, we conclude that F'' is simple. Since we know by Lemma 2.2.8 that $F'' = F'$, we obtain that F' is simple. $\square$

Let us call a word w in A, B, B^{-1} **reduced**, if in w, B and B^{-1} are not adjacent.

Theorem 2.2.10 (Theorem 4.6, [5]). *Distinct reduced words in A, B, B^{-1} correspond to distinct elements of F.*

Proof. We will work with reduced words in A, B, B^{-1}. Let w be a reduced word in A, B, B^{-1}, and $\overline{w}$ be the corresponding element in F. Denote length of a word w by

$|w|$. We want to show that if we have $\overline{w_1} = \overline{w_2}$ in F for two reduced words w_1, w_2 in A, B, B^{-1}, then we obtain $w_1 = w_2$.

Assume that $\overline{w_1} = \overline{w_2}$ for two reduced words w_1, w_2 in A, B, B^{-1}, and $w_1 \neq w_2$. Select these words such that $|w_1| + |w_2|$ is minimal. Without loss of generality, suppose that w_1 ends with B^{-1}, and w_2 ends with B. Then, we get $\overline{w_1 B} = \overline{w_2 B}$ since $\overline{w_1} = \overline{w_2}$, but $w_1 B \neq w_2 B$. Also, we have $|w_1 B| + |w_2 B|$ is still minimal since $|w_1 B| = |w_1| - 1$ and $|w_2 B| = |w_2| + 1$. Therefore, if we multiply w_1 and w_2 from the right by B^k for some suitable integer k, then we can say that w_1 ends with A.

Assume without loss of generality that w_1 ends with A. Similar to the previous paragraph, since $|w_1| + |w_2|$ is minimal, w_2 must end with B or B^{-1}. Otherwise, there might be some cancellations, and this gives us a shorter length. There is a group homomorphism ϕ from F to $\mathbb{Z}$, sending A to 1 and B to 0. Therefore, we have $\phi(\overline{w_1}) = \phi(\overline{w_2})$. Since $\phi(A) = 1$ and $\phi(B) = 0$, the number of A's appearing in w_1 is as same as the number of A's appearing in w_2. Let n be this number of A's. Since w_1 ends with A, we have $n > 0$. Now, we know that $A(\frac{3}{4}) = \frac{1}{2}$ and $A^2(\frac{3}{4}) = A(\frac{1}{2}) = \frac{1}{4}$. So, by the definition of A, we see that A takes the interval $[\frac{1}{2}, \frac{3}{4}]$ to the interval $[\frac{1}{4}, \frac{1}{2}]$, and $A^k(\frac{3}{4}) = (\frac{1}{2})^k = 2^{-k}$ for some integer k. Since B and B^{-1} are identity on the interval $[0, \frac{1}{2}]$, and w_1 ends with A, we obtain that $\overline{w_1}(\frac{3}{4}) = 2^{-n}$.

Now, assume that w_2 ends with B. Then, w_2 ends with AB^m for some $m \in \mathbb{Z}^+$. We know that $B(\frac{3}{4}) = \frac{5}{8}$ and $B^2(\frac{3}{4}) = B(\frac{5}{8}) = \frac{9}{16}$. So, by the definition of B, we see that B takes the interval $[\frac{1}{2}, \frac{3}{4}]$ to the interval $[\frac{1}{2}, \frac{5}{8}]$. In other words, B "pushes" $\frac{3}{4}$ to $\frac{1}{2}$ but it never is equal to $\frac{1}{2}$. Therefore, we have $\frac{1}{2} < B^m(\frac{3}{4}) < \frac{3}{4}$, and so, $2^{-2} = \frac{1}{4} < AB^m(\frac{3}{4}) < \frac{1}{2} = 2^{-1}$. Again, since B and B^{-1} are identity on $[0, \frac{1}{2}]$, we have $\overline{w_2}(\frac{3}{4}) = AB^m(\frac{3}{4}) \in (\frac{1}{4}, \frac{1}{2}) = (2^{-2}, 2^{-1})$. This implies that $\overline{w_2}(\frac{3}{4})$ is not a power of 2, which is a contradiction to the fact that $2^{-n} = \overline{w_1}(\frac{3}{4}) = \overline{w_2}(\frac{3}{4})$. Thus, w_2 ends with B^{-1}.

Now, we know that $\frac{7}{8} \leq B^{-1}(x) = A^{-1}(x)$ for every $x \in [\frac{3}{4}, 1]$. In other words, w_2 ends with B^{-1}, and B^{-1} "pushes" $\frac{3}{4}$ to 1. Then, since $\overline{w_2}(\frac{3}{4}) = 2^{-n}$, we obtain that $w_2 = w_3 w_4$ for the reduced words w_3, w_4 in A, B, B^{-1} with $w_4(\frac{3}{4}) = \frac{3}{4}$, and w_3 ends with A or B. If w_3 ends with A, then since B and B^{-1} are trivial on $[0, \frac{1}{2}]$, we have $\overline{w_2}(\frac{3}{4}) = \overline{w_3}(\frac{3}{4}) = 2^{-n'}$ where n' is the number of A's in w_3. Since $w_4(\frac{3}{4}) = \frac{3}{4}$ and

B^{-1} "pushes" $\frac{3}{4}$ to 1, we have A in w_4, which "pulls" back to $\frac{3}{4}$. This implies that $n' < n$, but we should have $n' = n$ since the number of A's in w_2 and w_3 are equal. This contradiction implies that w_3 ends with B. However, we already know from the previous paragraph that $\overline{w_2}(\frac{3}{4})$ is not even a power of 2. This contradiction completes the proof. □

Definition 12. *Let G be a group generated by a finite set S. For $g \in G$, define the length of g with respect to S by the following:*

$$|g|_S = \min\{n \mid g = s_1 \cdot s_2 \cdots s_n,\ s_i \in S^{\mp 1}\}.$$

Also, the ***growth of a function*** *of G with respect to S is*

$$\Phi_{(G,S)}(n) = \big|\{g \in G : |g|_S \leq n\}\big|.$$

It can be shown that the growth rate of such a function is independent of S.

A group has **exponential growth** if $\Phi_{(G,S)}$ grows exponentially. (See [16] for details.)

As a consequence of Theorem 2.2.10 we have:

Corollary 2.2.11. *Thompson's group F grows exponentially.*

Theorem 2.2.12 (Theorem 4.8, [5]). *Let $H \leq F$ be a nonabelian subgroup. Then, H contains a subgroup isomorphic to $\bigoplus_{i=1}^{\infty} \mathbb{Z}$.*

Proof. Let $K = \langle f, g \mid [f, g] \neq 1\rangle$. Denote the set of interior points of an interval J by $int(J)$. Let $I_1, I_2, \ldots, I_n$ be the closed intervals in $[0, 1]$ with every $int(I_k) \neq \varnothing$ such that for every integer k with $1 \leq k \leq n$, we have if I_k has an endpoint x, then $f(x) = g(x) = x$, and if $x \in int(I_k)$, then either $f(x) \neq x$ or $g(x) \neq x$.

We will show for every integer k with $1 \leq k \leq n$ that the endpoints of I_k are cluster points of the K-orbit, $K_x = \{k(x) \mid k \in K\}$, for all $x \in int(I_k)$. Let $x \in int(I_k)$ and $0 \leq y = inf(K_x)$ where $K_x = \{k(x) \mid k \in K\} \subseteq [0, \frac{1}{2}] = I_1$. Assume that y is not the left endpoint of I_k. Then, we have either $f(y) \neq y$ or $g(y) \neq y$. Assume that $f(y) \neq y$. In this case, we get either f or f^{-1} is decreasing, i.e., either $f(y) < y$ or $f^{-1}(y) < y$. Therefore, we "push" y through the point 0 by using f or f^{-1}. Hence, y is the left endpoint of I_k. Similarly, the least upper bound, i.e., supremum, of K_x

becomes the right endpoint of I_k. Thus, we see for every $k \in \mathbb{Z}$ with $1 \leq k \leq n$ that cluster points of K_x for every $x \in I_k$ become the endpoints of I_k.

Let $h_1 = [f, g]$. We know that $h_1 \in K$ and $h_1 \neq 1$. Suppose that $supp(h_1) = [a, b]$ for $0 < a, b < 1$. Similar to the commutators in F, we obtain that h_1 is trivial near the endpoints of I_1. So, $supp(h_1) = [a, b] \subsetneqq [0, \frac{1}{2}] = I_1$. By the previouos paragraph, there exists $k \in K$ such that $k([a, b]) \subseteq [a^{'}, b^{'}]$, where $[a^{'}, b^{'}] \cap [a, b] = \varnothing$. In other words, we can "push" $[a, b]$ through the point 0 in I_1. Since k's are increasing, we have $k(b) < a$, and also $k(a) < k(b)$ when $a < b$. We know from a fact that $k(supp(h_1)) = supp(kh_1k^{-1})$. Therefore, if we call the conjugate $kh_1k^{-1} = h_2$, we obtain $supp(kh_1k^{-1}) = supp(h_2) \subseteq [a^{'}, b^{'}]$. In other words, there exists $k \in K$ such that $kh_1k^{-1} = h_2$ and $supp(h_2) \cap supp(h_1) = \varnothing$ in I_1. Similarly, $h_3 = kh_2k^{-1}$, and $h_4 = kh_3k^{-1}$, and so on. Hence, there exists an infinite sequence of functions $h_1, h_2, h_3, \ldots$ in K such that $supp(h_i) \cap supp(h_j) = \varnothing$ in I_1 for every $i, j \in \mathbb{Z}^+$. Thus, we obtain $[h_i, h_j] = 1$ in I_1 for every $i, j \in \mathbb{Z}^+$. If $[h_i, h_j] = 1$ for every $i, j \in \mathbb{Z}^+$, then $h_1, h_2, h_3, \ldots$ form a basis of a free abelian subgroup of K.

We know that for all $i, j \in \mathbb{Z}^+$, $[h_i, h_j]$ is trivial in I_1, but it may be nontrivial in some other closed intervals. Now, suppose that $[h_2, h_3]$ is trivial in I_1 and I_2, but it is not trivial in I_3. We know from the previous paragraphs that by using $k \in K$, we can "push" any interior points to the left endpoint of the interval I_1, and we obtain conjugate $kh_1k^{-1} = h_2$ from nontrivial h_1. Hence, in this case, if we change h_1 with nontrivial $[h_2, h_3]$ and I_1 with I_3, and if we repeat the same procedure as before, then we get a trivial commutator in I_3. So, call $1 \neq [h_2, h_3] = t_1 \in I_3$. Since t_1 is a commutator, it is trivial near the endpoints of I_3. Let $supp(t_1) = [c, d]$. There exists $k \in K$ such that $k([c, d]) \subseteq [c^{'}, d^{'}]$ in I_3, where $[c, d] \cap [c^{'}, d^{'}] = \varnothing$. Therefore, $k([c, d]) = k(supp(t_1)) = supp(kt_1k^{-1}) \subseteq [c^{'}, d^{'}]$. Hence, call the conjugate $kt_1k^{-1} = t_2$ so that $supp(t_1) \cap supp(t_2) = \varnothing$. Therefore, $[t_1, t_2] = 1$ in I_3. Since $t_1 = [h_2, h_3]$ is already trivial in I_1 and I_2, we obtain that $[t_1, t_2]$ is trivial in I_1 and I_2. Thus, we can continue to form a basis of a free abelian group of K.

As a result, we have the following procedure for every integer k with $1 \leq k \leq n$ and for all $i, j \in \mathbb{Z}^+$:
If $[h_i, h_j] = 1$ in I_k, then continue to I_{k+1} with $[h_i, h_j]$.

If $[h_i, h_j] \neq 1$ in I_k, then say $t_1 = [h_i, h_j] \neq 1$ in I_k. Take $k \in K$ so that $k(supp(t_1)) = supp(kt_1k^{-1})$. Then, call $t_2 = kt_1k^{-1}$ with $supp(t_2) \cap supp(t_1) = \varnothing$ in I_k. Therefore, $[t_1, t_2] = 1$ in I_k. Thus, take $[h_i, h_j] = [t_1, t_2]$ and repeat from I_1. Since there are n-many intervals, eventually we obtain an infinite sequence of elements $h_1, h_2, h_3, \ldots \in K$ such that $\{h_1, h_2, \ldots\}$ generate a free abelian subgroup of infinite rank. $\square$

Corollary 2.2.13. *Every free subgroup of F is abelian.*

2.3 The Word Problem in F

In this section, we will describe the word problem, and explain an algorithm for the word problem in Thompson's group F. We will mainly follow [27]. For more on decision problems related to groups, one can see [20].

In his influential paper, Dehn [6] posed three significant decision problems.

Let G be a group given by a presentation. Then:

- **The Word Problem:** Decide given a word w in the generators of G, whether $w = 1$ or not.
- **The Conjugacy Problem:** Given two words $w_1, w_2 \in G$. Decide if there is $g \in G$ such as $w_1 = g^{-1}w_2g$.
- **The Isomorphism Problem:** Decide whether given two presentations define isomorphic groups.

It is clear that if G's conjugacy problem is decidable, then the word problem in G is also decidable by taking $w_2 = 1$.

It can be shown that the decidability of the above problems is independent of the particular presentation of a group. Also, there are finitely presented groups with an undecidable word problem ([23]).

Since we describe the word problem, we move on to the word problem in Thompson's

group F. Recall that F has the following infinite presentation:

$$F = \langle X_0, X_1, X_2, \ldots \mid X_k^{-1} X_n X_k = X_{n+1} \text{ for } k < n \rangle.$$

Also, recall that the unique normal form for an element of F is a word of the form

$$X_{q_1} X_{q_2} \cdots X_{q_m} X_{r_n}^{-1} \cdots X_{r_2}^{-1} X_{r_1}^{-1}$$

satisfying:
($\mathbf{N}_1$) $q_1 \leq q_2 \leq \cdots \leq q_m$ and $r_1 \leq r_2 \leq \cdots \leq r_n$, and
($\mathbf{N}_2$) if X_i and X_i^{-1} appear at the same time, then we have either X_{i+1} appears or X_{i+1}^{-1}.

Any word in this form satisfying only N_1 is called a word in a **seminormal form**. It is clear that a seminormal form is not unique:

Example 6. *The elements*

$$X_0 X_1 X_2 X_5 X_6 X_5^{-1} X_2^{-1} X_0^{-1} \textit{ and } X_0 X_1 X_3 X_5 X_6 X_5^{-1} X_3^{-1} X_0^{-1}$$

are in seminormal forms. However, their unique normal form is

$$X_0 X_1 X_4 X_5 X_4^{-1} X_0^{-1}.$$

In this section, we first find a seminormal form of any given word. Then, by erasing the elements, which do not satisfy the N_2, we reduce it to the normal form to see whether it satisfies the word problem. To get a seminormal form of a word systematically, we have the following rewriting system denoted by $\boldsymbol{S}$:
For all numbers k, n with $k < n$,

$$S_1 : X_k X_k^{-1} \leftrightarrow 1$$

$$S_2 : X_n X_k \leftrightarrow X_k X_{n+1}$$

$$S_3 : X_n^{-1} X_k \leftrightarrow X_k X_{n+1}^{-1}$$

$$S_4 : X_k^{-1} X_n \leftrightarrow X_{n+1} X_k^{-1}$$

$$S_5 : X_k^{-1} X_n^{-1} \leftrightarrow X_{n+1}^{-1} X_k^{-1}$$

These rewriting rules can be applied to any word as in the following example:

Example 7.

$$X_0X_1\underline{X_2X_5}X_6X_5^{-1}X_2^{-1}X_0^{-1} \xrightarrow{S_2} X_0X_1\underline{X_4X_2}X_6X_5^{-1}X_2^{-1}X_0^{-1}$$

$$\xrightarrow{S_2} X_0X_1X_4X_5X_2\underline{X_5^{-1}X_2^{-1}}X_0^{-1} \xrightarrow{S_5} X_0X_1X_4X_5\underline{X_2X_2^{-1}}X_4^{-1}X_0^{-1}$$

$$\xrightarrow{S_1} X_0X_1X_4X_5X_4^{-1}X_0^{-1}.$$

Note that we cannot apply any of $S_1, S_2, \cdots, S_5$ to w_5. Such a word is $\boldsymbol{S}$**-reduced**.

One can observe that this process terminates at an S-reduced word starting with any word, i.e., S is "terminating." Also, applying the rewriting rules in a different order will end in the same S-reduced word, i.e., S is "confluent." For more details, one can see [12]. Also, interested readers can see [8] and Proposition 3.1 in chapter 2 of [28]. The following lemma is evident.

Lemma 2.3.1 (Lemma 1, [27])**.** *A word is S-reduced if and only if it is in a seminormal form.*

We see that S changes the indices to get a seminormal form. Now, let us define a parametric function Δ_i for $i \in \mathbb{Z}$ as, $\Delta_i(X_n^{\pm 1}) := X_{n+i}^{\pm 1}$. If we have an element contradicting to the relation $X_k^{-1}X_nX_k = X_{n+1}$ for $k < n$, then Δ_i is not defined.

Before we give an algorithm that outputs a seminormal form for any word w, we examine the case, where w is given as a product of two words w_1 and w_2 in seminormal forms. Let $w_1 = p_1n_1$ and $w_2 = p_2n_2$, where p_1, p_2 and n_1, n_2 are the positive and negative parts of w_1, w_2, respectively. The steps of the idea to find a seminormal form of $w = w_1w_2 = p_1n_1p_2n_2$ are fundamentally given in order as follows:

1. Find a seminormal form of n_1p_2,
2. From the obtained word, find separately seminormal forms of the positive and negative parts,
3. Concatenate the results.

We have several algorithms that compute these steps. We will now give the first algorithm that performs step 1 for a negative n and a positive p in seminormal forms.

Algorithm 1 [Algorithm 2, [27]] Seminormal form of a product of negative and positive words with seminormal forms.

SIGNATURE. $w = Join_{-,+}(n, p, i_1, i_2)$.

INPUT. *Integers* $i_1, i_2 \in \mathbb{Z}$, *and seminormal forms of* $p = X_{t_1} \cdots X_{t_{u-1}} X_{t_u}$ *and* $n = X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$.

OUTPUT. *Seminormal form* w *in* F *such that* $w =_F \Delta_{i_1}(n)\Delta_{i_2}(p)$.

COMPUTATIONS.

A) *If* n *or* p *is an empty word, then output a product* pn.

B) *Add the integer* i_1 *to* r_1, *index of the last letter in* n, *and also add* i_2 *to* t_1, *index of the first letter in* p.

C) *If* $r_1 + i_1 = t_1 + i_2$, *then delete* $X_{r_1}^{-1}$ *and* X_{t_1} *from the product, and hence, compute* $w = Join_{-,+}(X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_2}^{-1} X_{t_2} \cdots X_{t_{u-1}} X_{t_u}, i_1, i_2)$. *Then, output* w.

D) *If* $r_1 + i_1 < t_1 + i_2$, *then delete* $X_{r_1}^{-1}$ *from the product, and add* 1 *to all indices of* p. *Compute* $w = Join_{-,+}(X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_2}^{-1} X_{t_1} \cdots X_{t_{u-1}} X_{t_u}, i_1, i_2 + 1)$, *and output* $wX_{r_1+i_1}^{-1}$.

E) *If* $r_1 + i_1 > t_1 + i_2$, *then delete* X_{t_1} *from the product, and add* 1 *to all indices of* n. *Compute* $w = Join_{-,+}(X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1} X_{t_2} \cdots X_{t_{u-1}} X_{t_u}, i_1 + 1, i_2)$, *and output* $X_{t_1+i_2} w$.

Lemma 2.3.2 (Lemma 2, [27]). *For any* $i_1, i_2 \in \mathbb{Z}$, *seminormal forms of* $p = X_{t_1} \cdots X_{t_{u-1}} X_{t_u}$ *and* $n = X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$, *Algorithm 1 gives a seminormal form for the product* $\Delta_{i_1}(n)\Delta_{i_2}(p)$.

Proof. We will use induction on the sum of the lengths of n and p, i.e., $|n| + |p|$. Suppose that $|n| + |p| = 0$. So, both n and p are empty words, and hence, the product is an empty word. By step (A), we have output $w = pn$, clearly in a seminormal form.

Now, suppose that the statement holds for $|n| + |p| = K$ and any shorter word. Then, we have the following four cases:

Case 1: $|n| = 0$ or $|p| = 0$

In this case, one of the words is an empty word. Hence, the product is already in a seminormal form.

Case 2: $r_1 + i_1 = t_1 + i_2$

The elements $X_{r_1+i_1}^{-1}$ and $X_{t_1+i_2}$ delete each other in the product $\Delta_{i_1}(n)\Delta_{i_2}(p)$. Thus,

we have $|n|+|p| < K$. By inductive assumption, $w =_F \Delta_{i_1}(n)\Delta_{i_2}(p)$ is a seminormal form.

Case 3: $r_1 + i_1 < t_1 + i_2$

Since n and p are already given in seminormal forms, we obtain that $r_1 + i_1$ is the smallest index in the product. Then, by using S_4 in S, we can rewrite the product as follows:

$$\Delta_{i_1}(n)\Delta_{i_2}(p) = X^{-1}_{r_s+i_1} \cdots X^{-1}_{r_2+i_1} \underline{X^{-1}_{r_1+i_1} X_{t_1+i_2}} X_{t_2+i_2} \cdots X_{t_u+i_2} \xrightarrow{S_4}$$

$$\xrightarrow{S_4} X^{-1}_{r_s+i_1} \cdots X^{-1}_{r_2+i_1} \underline{X_{t_1+i_2+1}} X_{t_2+i_2+1} \cdots X_{t_u+i_2+1} \underline{X^{-1}_{r_1+i_1}}$$

Observe since $r_1 + i_1$ is the smallest index in the product $\Delta_{i_1}(n)\Delta_{i_2}(p)$, the smallest index in $w = Join_{-,+}(X^{-1}_{r_s} X^{-1}_{r_{s-1}} \cdots X^{-1}_{r_2} X_{t_1} \cdots X_{t_{u-1}} X_{t_u}, i_1, i_2 + 1)$ is not less than $r_1 + i_1$. Since we eliminate $X^{-1}_{r_1+i_1}$ from w, by the inductive assumption, we get that w is a seminormal form for

$$\Delta_{i_1}(X^{-1}_{r_s} X^{-1}_{r_{s-1}} \cdots X^{-1}_{r_2})\Delta_{i_2+1}(X_{t_1} \cdots X_{t_{u-1}} X_{t_u}).$$

Thus, $wX^{-1}_{r_1+i_1} =_F \Delta_{i_1}(n)\Delta_{i_2}(p)$, and it is a seminormal form.

Case 4: $r_1 + i_1 > t_1 + i_2$

Similar to the previous case, since n and p are given in seminormal forms, we get that $t_1 + i_2$ is the smallest index in the product. Then, by using S_3 in S, we can rewrite the product as follows:

$$\Delta_{i_1}(n)\Delta_{i_2}(p) = X^{-1}_{r_s+i_1} \cdots X^{-1}_{r_2+i_1} \underline{X^{-1}_{r_1+i_1} X_{t_1+i_2}} X_{t_2+i_2} \cdots X_{t_u+i_2} \xrightarrow{S_3}$$

$$\xrightarrow{S_3} \underline{X_{t_1+i_2}} X^{-1}_{r_s+i_1+1} \cdots X^{-1}_{r_2+i_1+1} \underline{X^{-1}_{r_1+i_1+1}} X_{t_2+i_2} \cdots X_{t_u+i_2}$$

Since $t_1 + i_2$ is the smallest index in the product $\Delta_{i_1}(n)\Delta_{i_2}(p)$, the smallest index in $w = Join_{-,+}(X^{-1}_{r_s} X^{-1}_{r_{s-1}} \cdots X^{-1}_{r_1} X_{t_2} \cdots X_{t_{u-1}} X_{t_u}, i_1 + 1, i_2)$ is not less than $t_1 + i_2$. Since we eliminate $X_{t_1+i_2}$ from w, by the inductive assumption, we get that w is a seminormal form for

$$\Delta_{i_1+1}(X^{-1}_{r_s} X^{-1}_{r_{s-1}} \cdots X^{-1}_{r_1})\Delta_{i_2}(X_{t_2} \cdots X_{t_{u-1}} X_{t_u}).$$

Thus, $X_{t_1+i_2} w =_F \Delta_{i_1}(n)\Delta_{i_2}(p)$, and it is a seminormal form. □

We see that Algorithm 1 computes the step (1). For step (2), one can easily design two algorithms with ideas similar to Algorithm 1. Also, one can easily show that lemmas similar to Lemma 2.3.2 are true for step (2). Let us denote these algorithms that compute seminormal forms of negative and positive words as $Join_{-,-}(n_1, n_2, i_1, i_2)$ and $Join_{+,+}(p_1, p_2, i_1, i_2)$, respectively. Therefore, if we combine these three algorithms, we can easily find a seminormal form of a product of two words given in seminormal forms. Hence, we have the following algorithm that computes this seminormal form.

Algorithm 2 [Algorithm 3, [27]] Seminormal form of a product of two seminormal forms.
SIGNATURE. $w = Join(w_1, w_2)$.
INPUT. *Seminormal forms* w_1 *and* w_2.
OUTPUT. *Seminormal form* w *such that* $w =_F w_1 w_2$
COMPUTATIONS.
A) *Find negative and positive parts of words* w_1 *and* w_2*, and then write them as* $w_1 = p_1 n_1$ *and* $w_2 = p_2 n_2$.
B) *Perform* $Join_{-,+}(n_1, p_2, 0, 0) = w^{'}$*, and write it as a product of a positive and a negative word* $w^{'} = p_2^{'} n_1^{'}$.
C) *Perform the algortihm* $Join_{+,+}(p_1, p_2^{'}, 0, 0) = w^{''}$.
D) *Perform the algortihm* $Join_{-,-}(n_1^{'}, n_2, 0, 0) = w^{'''}$.
E) *Concatenate and output* $w^{''} w^{'''}$.

We know from Lemma 2.3.2 that $Join_{-,+}(n, p, i_1, i_2)$, $Join_{-,-}(n_1, n_2, i_1, i_2)$ and $Join_{+,+}(p_1, p_2, i_1, i_2)$ give us seminormal forms. So, the following lemma is obvious.

Lemma 2.3.3 (Lemma 3, [27])**.** *For any words* w_1 *and* w_2*, which are given in seminormal forms, the word* $w = Join(w_1, w_2)$ *is a seminormal form of the product* $w_1 w_2$.

We have seen how to find a seminormal form of a product of two words given in seminormal forms. Now, we will give an algorithm that computes a seminormal form of a word given in generators of F.

Algorithm 3 [Algorithm 4, [27]] Seminormal form of any word in F
SIGNATURE. $u = SeminormalForm(w)$.

INPUT. *A word w in generators of F.*

OUTPUT. *A seminormal form u such that $u = w$ in F.*

COMPUTATIONS.

A) *If the word w is empty or is a word with one letter, then directly output w itself.*

B) *Divide w into almost two halves, i.e., write it as product $w = w_1 w_2$ such that $|w_1| - |w_2| \leq 1$.*

C) *Recursively apply*

$u_1 = SeminormalForm(w_1)$, *and*

$u_2 = SeminormalForm(w_2)$.

D) *Perform the algorithm, and let $u = Join(u_1, u_2)$.*

E) *Output the result u.*

Lemma 2.3.4 (Lemma 4, [27])**.** *For any word w, which is given in generators of F, the output word $u = SeminormalForm(w)$ is a seminormal form of w.*

Proof. We will use induction on the length of the given word w. If $|w| = 1$, then it is already in a seminormal form. So, the output is correct. Hence, the base case is done. Suppose that the statement is true for any word of length $|w| = N$ and any shorter words. Let $|w| = N + 1$. Then, by step (B), we have w_1 and w_2 such that $|w_1| < N$ and $|w_2| < N$. By the inductive assumption, we get that in step (C) u_1 and u_2 are in seminormal forms. Since we already know from Lemma 2.3.3 that $Join(u_1, u_2)$ gives a seminormal form of a product of two seminormal forms u_1 and u_2, we obtain that the output u is a seminormal form of w. □

So, we know how to find a seminormal form of any word. Now, we will show how to find and cancel out the terms contradicting to (N_2), i.e., if X_i and X_i^{-1} appear at the same time, then we have neither X_{i+1} appears nor X_{i+1}^{-1}.

Lemma 2.3.5 (Lemma 5, [27])**.** *Given $w = X_{t_1} \cdots X_{t_{u-1}} X_{t_u} X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$ in a seminormal form. Suppose $(X_{t_m}, X_{r_n}^{-1})$ is the pair of generators in w contradicting to (N_2) where m and n are maximal with this property. In other words, we have $t_m = r_n$, and hence, X_{t_m} and $X_{r_n}^{-1}$ appear but X_{t_m+1} and $X_{r_n+1}^{-1}$ do not appear. Also, there are no other elements X_{t_i} or $X_{r_j}^{-1}$ contradicting to (N_2) such that $i > m$ and*

$j > n$. Let

$$w^{'} = X_{t_1} \cdots X_{t_{m-1}} \Delta_{-1}(X_{t_{m+1}} \cdots X_{t_u} X_{r_s}^{-1} \cdots X_{r_{n+1}}^{-1}) X_{r_{n-1}}^{-1} \cdots X_{r_1}^{-1}.$$

Then $w =_F w^{'}$. Furthermore, if there is any pair $(X_{t_c}, X_{r_d}^{-1})$ in $w^{'}$ contradicting to N_2, then $c < m$ and $d < n$.

Proof. We know w is in a seminormal form. By definition of seminormal form, all the indices have the ordering as $t_1 \leq \cdots \leq t_{u-1} \leq t_u$ and $r_1 \leq \cdots \leq r_{s-1} \leq r_s$. By definition of (N_2), all indices in $X_{t_{m+1}} \cdots X_{t_{u-1}} X_{t_u} X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_{n+1}}^{-1}$ are greater than $t_m + 1 = r_n + 1$. Hence, the opposites of S_2 and S_5 are applicable. Therefore, when we apply the opposites of S_2 and S_5 until the cancelation of X_{t_m} and $X_{r_n}^{-1}$ with S_1, we get the word $w^{'}$ since Δ_{-1} decreases all the indices by 1. Hence, $w =_F w^{'}$, and clearly, $w^{'}$ is in a seminormal form.

Since we choose m and n as maximals with that property, any numbers c, d such that $c > m$, $d > n$ contradict with this choice. Hence, for any pair $(X_{t_c}, X_{r_d}^{-1})$ in $w^{'}$ contradicting to N_2, we have $c < m$ and $d < n$. □

By starting from the middle of a word, we can detect and cancel out the terms contradicting to N_2 by the previous lemma. This idea is used in the following algorithm. So, the algorithm detects all "contradictory" pairs and deletes these "contradictory" pairs using Δ_i. A vital characteristic of this algorithm is that it keeps the data about how indices should be changed later, instead of changing them instantly. To store this data, we will use two stacks, namely S_1 and S_2. Stack S_1 is for the positive subword of w, while S_2 is for the negative subword of w.

Algorithm 4 [Algorithm 5, [27]] Deleting contradictory pairs from a seminormal form

SIGNATURE. $u = DeleteContradictoryPairs(w)$.

INPUT. *A seminormal form* $w = X_{t_1} \cdots X_{t_{k-1}} X_{t_k} X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$.

OUTPUT. *A word u (the normal form of w).*

INITIALIZATION. *Let $\epsilon = 0, \epsilon_1 = 0, \epsilon_2 = 0, u_1 = 1$ and $u_2 = 1$. Let $w_1 = X_{t_1} \cdots X_{t_{k-1}} X_{t_k}$ and $w_2 = X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$ be the positive and negative parts of w. Let $S_1 = \varnothing, S_2 = \varnothing$ be two stacks.*

COMPUTATIONS.

A) *Let the current* $w_1 = X_{t_1} \cdots X_{t_{k-1}} X_{t_k}$ *and* $w_2 = X_{r_s}^{-1} X_{r_{s-1}}^{-1} \cdots X_{r_1}^{-1}$.

B) *Let* X_a *be the leftmost letter of* u_1*,* X_b *be the rightmost letter of* u_2*, and* δ_i *(*$i = 1, 2$*) be the top element of* S_i*, i.e., the last element that was put there. If* S_i *is empty, then the corresponding variable is not defined.*

1) *If* $k > 0$ *and (*$s = 0$ *or* $t_k > r_s$*), then:*

a) *multiply* u_1 *on the left by* X_{t_k};

b) *delete* X_{t_k} *from* w_1;

c) *push* 0 *into* S_1;

d) *go to (5).*

2) *If* $s > 0$ *and (*$k = 0$ *or* $r_s > t_k$*), then:*

a) *multiply* u_2 *on the right by* $X_{r_s}^{-1}$*;*

b) *delete* $X_{r_s}^{-1}$ *from* w_2*;*

c) *push* 0 *into* S_2*;*

d) *go to (5).*

3) (Contradictory pair detection) *If* $t_k = r_s$ *and (the numbers* $a - \delta_1$ *and* $b - \delta_2$ *(those that are defined) are not equal to* t_k *or* $t_k + 1$*), then:*

a) *delete* X_{t_k} *from* w_1*;*

b) *delete* $X_{r_s}^{-1}$ *from* w_2*;*

c) *if* S_1 *is not empty, increase the top element of* S_1 *by* 1*;*

d) *if* S_2 *is not empty, increase the top element of* S_2 *by* 1*;*

e) *go to (5).*

4) (Neither contradictory pair nor erasable) *If (1) - (3) are not applicable (when* $t_k = r_s$ *and (one of the numbers* $a - \delta_1, b - \delta_2$ *is defined and is equal to either* t_k *or* $t_k + 1$*)), then:*

a) *multiply* u_1 *on the left by* X_{t_k};

b) *multiply* u_2 *on the right by* $X_{r_s}^{-1}$*;*

c) *delete* X_{t_k} *from* w_1;

d) *delete* $X_{r_s}^{-1}$ *from* w_2*;*

e) *push* 0 *into* S_1*;*

f) *push* 0 *into* S_2*;*

g) *go to (5).*

5) *If w_1 or w_2 is not empty, then go to (1).*

C) *While u_1 is not empty:*

1) *Let X_{t_1} be the first letter of u_1 (i.e., $u_1 = X_{t_1} \cdot u_1^{'}$);*

2) *Take c from the top of S_1 (i.e., take the last element that was put in S_1), and add to ϵ_1 (i.e., $\epsilon_1 + c \to \epsilon_1$);*

3) *Multiply w_1 on the right by $X_{t_1-\epsilon_1}$ (i.e., $w_1 \cdot X_{t_1-\epsilon_1} \to w_1$) ;*

4) *Delete X_{t_1} from u_1.*

D) *While u_2 is not empty:*

1) *Let $X_{r_1}^{-1}$ be the last letter of u_2 (i.e., $u_2 = u_2^{'} \cdot X_{r_1}^{-1}$);*

2) *Take c from the top of S_2 (i.e., take the last element that was put in S_2), and add to ϵ_2 (i.e., $\epsilon_2 + c \to \epsilon_2$);*

3) *Multiply w_2 on the left by $X_{r_1-\epsilon_2}^{-1}$ (i.e., $X_{r_1-\epsilon_2}^{-1} \cdot w_2 \to w_2$) ;*

4) *Delete $X_{r_1}^{-1}$ from u_2.*

E) *Return to $w_1 w_2$.*

Proposition 2.3.6 (Proposition 3, [27])**.** *The Algorithm 4 terminates at the normal form u of a seminormal form w. The number of operations needed for this algorithm to stop is bounded by $C \cdot |w|$, where C is a constant independent from w.*

Proof. It is clear from Lemma 2.3.5 that the algorithm outputs the normal form u of a seminormal form w. Since the algorithm proceeds letter-by-letter, and no letter is processed more than once, the time estimate is evident. □

Theorem 2.3.7 (Theorem 1, [27])**.** *In Thompson's group F, the normal form of a word w can be computed in time $O(|w| \cdot log|w|)$.*

2.4 Amenability

Motivated by the Banach-Tarski paradox, von Neumann defined:

Definition 13. *(von Neumann [22]) A group G is* ***amenable*** *if there is a function $\mu : 2^G \to [0, 1]$ such that*

1. $\mu(G) = 1$,

2. *If* $A \cap B = \varnothing$*, then* $\mu(A \cup B) = \mu(A) + \mu(B)$*,*

3. $\mu(gA) = \mu(A)$ *for all* $g \in G$*,* $A \subseteq G$*.*

Such a function is called a **finitely additive invariant probability measure**.

Example 8. *Every finite group is amenable. Define* $\mu(A) = \frac{|A|}{|G|}$*, where* $|A|$ *(respectively,* $|G|$*) is the number of elements in* A *(respectively, in* G*). (In fact, this is the unique such measure in this case.)*

Theorem 2.4.1. *(von Neumann [22])*

1. *The free group with two generators,* F_2*, is not amenable.*

2. *Abelian groups are amenable.*

3. *Amenable groups are closed under*

   (a) *taking subgroups,*

   (b) *taking quotients,*

   (c) *direct limits.*

So, it follows that a group, which contains F_2 as a subgroup is not amenable.

The *von Neumann Problem* asks whether a group that is not amenable must contain F_2. This was answered negatively by Olshanskii in [24].

Motivated by the von Neumann Problem, Geoghegan discovered the interest in knowing whether or not the Thompson's group F is amenable. In 1979 (see p. 549 of [10]), he conjectured that F does not contain a nonabelian free group and that F is not amenable. By Corollary 2.2.13, F does not contain any nonabelian free groups. However, it remains unknown whether F is amenable or not. This problem seems to be an important problem in infinite group theory, and motivates the study of F from various viewpoints.

CHAPTER 3

THOMPSON'S GROUP F AND GROUP BASED CRYPTOGRAPHY

In this chapter, we will give general notions of cryptography. Interested readers can consult the books [3], [7], [29] and chapter 3 of [15].

Cryptography uses mathematical tools to encrypt and decrypt data so that third parties cannot read, understand, or use data. Historically, cryptography was used to protect the secrecy of military and diplomatic communication. Today, with expansion of technology and information economy, cryptography plays a significant role not only for governments or organizations but also for individuals. For example, cryptography assures the confidentiality of data between two people. Also, it examines the authentication of the sender and receiver of a message.

Cryptography can be grouped into two main categories as symmetric key cryptography and asymmetric key cryptography. The difference depends on the key that is used for encryption and decryption. Symmetric key cryptography, also known as secret key cryptography, uses the same key for both encryption and decryption. Two communicating parties share the same password/key to encrypt and to decrypt the message. However, asymmetric key cryptography, also known as public key cryptography, uses different keys to encrypt and decrypt data. Each party has a pair of keys, one public key, and one secret private key. In this chapter, we will be interested in public key cryptography and mainly in key exchange protocols.

3.1 Public Key Cryptography

In this section, we will first explain the rudiments of public key cryptography, and then, we will focus on key exchange protocols. We will mainly follow the book [21].

Ideas, techniques of public key cryptography were officially invented in 1976 by Whitfield Diffie and Martin Hellman [9] to handle the key distribution between two parties. One of the primary importance of public key cryptography is that two communicating parties do not have to meet or share any secret password before encryption.

In public key cryptography, there are two keys used for each of the communicating parties. Each party has his/her private key and one common public key. Usually, the encryption key becomes public, while the decryption key is private. Therefore, anyone can encrypt using the public key, but only the private key owner can decrypt the secret message.

The basic idea behind public key cryptography is that encryption should be efficiently easy, but decryption should be hard to compute without the private key. For example, the RSA cryptosystem [25] uses the fact that the computation of the product of two large prime numbers is easy but factorizing a huge number is very hard. This kind of mathematical process is usually referred to as a trapdoor or a one-way function. For more formal definitions, one can see references. The main point is that there should be efficient ways (this generally means polynomial-time concerning an input's complexity) to compute this function. However, there should not be any apparent polynomial-time algorithm to compute the inverse of this function. Hence, one-way functions play a crucial role in public key cryptography.

A significant part of public key cryptography is the key establishment protocols. We will explain how two parties (say, Alice and Bob) share a secret key over an unprotected communication service.

We will present a key establishment process with an important example, which is considered as the origin of public key establishment protocols.

Example 9. *(The Diffie-Hellman key establishment protocol [9])*

The most straightforward and original implementation of the protocol uses the multiplicative group of integers modulo p, where p is a prime number, and g is primitive mod p, (i.e., g is a primitive root modulo n, that is, for every integer x that is relatively prime to n, there is an integer k such that $g^k \equiv x \pmod{n}$). A more general version of the protocol uses an arbitrary finite cyclic group. We will explain this general version as follows:

1. Alice and Bob fix a finite cyclic group G (written multiplicatively), and a generating element $g \in G$. The group G and the element g are public.
2. Alice chooses a random $a \in \mathbb{N}$, which she keeps as a secret, and she sends g^a to Bob.
3. Bob chooses a random $b \in \mathbb{N}$, which he keeps as a secret, and he sends g^b to Alice.
4. Since Alice knows a and g^b, she computes $K_A = (g^b)^a = g^{ba}$.
5. Since Bob knows b and g^a, he computes $K_B = (g^a)^b = g^{ab}$.

Since $ab = ba$, both Alice and Bob have the shared secret key $K = K_A = K_B$.

If the public information G and g are selected appropriately, then the protocol is considered secure against eavesdroppers. If an eavesdropper, say Eve, wants to obtain the key, she must solve the *Diffie-Hellman problem*. **The Diffie-Hellman problem** is the problem of obtaining g^{ab} from the public information g^a and g^b. This problem is considered hard to decipher if parameters are chosen appropriately.

Also, there is a famous problem called *the discrete logarithm problem*, which is to obtain a from g and g^a. It is clear that if there is an efficient way to solve the discrete logarithm problem, then it is easy to solve the Diffie-Hellman problem. However, there is no proof that the Diffie-Hellman problem and the discrete logarithm problem are equivalent. Note that, since g and g^a are publicly known, and a is a natural number in the discrete logarithm problem, Eve can go over all natural numbers x one by one to see if g^x matches with g^a. This type of attack is called a "brute force attack", or a "length-based attack". This method will take $O(|g|)$ multiplications, where $|g|$ is the order of the element g. In practical applications, order $|g|$ is about 10^{300}. Hence, this method is computationally unfeasible.

With increasing computation power, there is a need for increasing security of this

type of systems. This gives rise to a rather new research area in both cryptography and group theory. There are noncommutative groups that are well-studied in combinatorial group theory, which also can be used in public key cryptography. Some examples are braid groups [14], matrix groups, Thompson's groups [27], and Grigorchuk groups [11].

The idea of using infinite nonabelian groups in cryptography goes back to Neal R. Wagner and Marianne R. Magyarik [30], who constructed a protocol depending on the unsolvability of the word problem for finitely presented groups in 1985. Today, their protocol may seem simple, but it was a guiding light. Recently, the usage of noncommutative group theory in cryptography has been increased. Most recommended protocols in this area depend on *search problems*, which are variants of decision problems. Protocols depending on search problems are similar to protocols depending on trapdoor functions. In the next section, we will explain search problems and some protocols, which use noncommutative groups.

3.2 Group Based Cryptography and Some Cryptographic Schemes

In this section, we will explain how noncommutative groups can be used in key exchange protocols from a public key cryptography point of view, and we will investigate some protocols. We will also explain how we can choose a group to be a platform group in a cryptographic system.

3.2.1 Protocols Utilizing The Conjugacy Search Problem

In Section 2.3, we learned the conjugacy problem (or conjugacy decision problem), that is, decide algorithmically if there is an element $g \in G$ such that $w_1 = g^{-1}w_2g$ for given two words $w_1, w_2 \in G$ where G is a group with a given presentation. The *conjugacy search problem* is the following:

The Conjugacy Search Problem: Let G be a group with solvable word problem. Given $w_1, w_2 \in G$ and that $g^{-1}w_1g = w_2$ for some $g \in G$, find one such g.

Note that, in the conjugacy decision problem, one is interested in the existence of

the conjugating element, whereas in the conjugacy search problem, the task is to find such a conjugating element, given it exists.

While group theory is mostly interested in the conjugacy decision problem, complexity theory is interested in the conjugacy search problem. Note that if we know such an element exists, we can go over all conjugate words to see the equivalence. However, this process takes at least exponential-time in the length of the conjugate. Hence, this is considered secure against brute force attacks. Thus, if there are no known attacks to solve the conjugacy search problem in G, then we can think of $g \to g^{-1}w_1g$ as a one-way function. So, we can form a public-key cryptographic protocol based on it.

Now, we introduce a protocol, which is a basic form of the protocol suggested by Ko, Lee, et. al. in [14].

1. An element $g \in G$ is published
2. Alice chooses secret $\alpha \in G$, and sends $\alpha^{-1}g\alpha$ to Bob.
3. Bob chooses secret $\beta \in G$, and sends $\beta^{-1}g\beta$ to Alice.
4. Alice computes $\alpha^{-1}(\beta^{-1}g\beta)\alpha$, and Bob computes $\beta^{-1}(\alpha^{-1}g\alpha)\beta$.

This protocol depends on the commutativity of α and β. So, if we choose α and β as commuting elements of the group G, then Alice and Bob agree on a shared secret key $\alpha^{-1}\beta^{-1}g\beta\alpha = \beta^{-1}\alpha^{-1}g\alpha\beta$. In the paper [14], the authors use braid group B_n as group G since it has a good normal form for its elements, and choose α and β from two subgroups, which naturally commute.

Hence, the general version of Ko-Lee protocol is the following:

Given publicly a group G and two subsets $A, B \subseteq G$ such that $ab = ba$ for any $a \in A, b \in B$.

1. An element $g \in G$ is published.
2. Alice chooses secretly $\alpha \in A$, and sends the element $\alpha^{-1}g\alpha$ to Bob.
3. Bob chooses secretly $\beta \in B$, and sends the element $\beta^{-1}g\beta$ to Alice.
4. While Alice computes $K_A = \alpha^{-1}(\beta^{-1}g\beta)\alpha$, Bob computes his secret key $K_B = \beta^{-1}(\alpha^{-1}g\alpha)\beta$.

Since $\alpha\beta = \beta\alpha$ (and hence, $\alpha^{-1}\beta^{-1} = \beta^{-1}\alpha^{-1}$) in G, Alice and Bob agree on the

shared secret key $K = K_A = K_B$.

In fact, solving the conjugacy search problem is not necessary to find shared, secret key K in the above protocol. If an adversary finds two elements $\alpha_1, \alpha_2 \in C_G(\beta)$, where $C_G(\beta)$ is the centralizer of β in G, such that $\alpha_1 g \alpha_2 = \alpha^{-1} g \alpha$, then $\alpha_1(\beta^{-1} g \beta)\alpha_2 = \beta^{-1}(\alpha_1 g \alpha_2)\beta = \beta^{-1}(\alpha^{-1} g \alpha)\beta = K$. Similarly, if an adversary finds two elements $\beta_1, \beta_2 \in C_G(\alpha)$ satisfying $\beta_1 g \beta_2 = \beta^{-1} g \beta$, then $\beta_1(\alpha^{-1} g \alpha)\beta_2 = \alpha^{-1}(\beta_1 g \beta_2)\alpha = \alpha^{-1}(\beta^{-1} g \beta)\alpha = K$.

However, it is not easy to find a proper platform group for the protocols above. There are some natural conditions for choosing such a group put forward in [26]:

(C_0) The group has to be well-studied in literature. Specifically, the conjugacy search problem has to be well-studied in the group, or reducible to a familiar problem.

(C_1) The word problem should be solvable easily (in linear-time or quadratic-time) in the group G by a deterministic algorithm. The elements of G must have an easily computable normal form.

This condition is essential because when Alice and Bob exchange keys, they need to solve the word problem for decryption. Also, (C1) is vital to hide messages in the cryptosystem.

(C_2) There should be no fast algorithm to solve the conjugacy search problem.

Note that showing that a group satisfies (C2) is very hard. Therefore, the condition can be counted as satisfied if the conjugacy search problem has been studied by many people over a long enough time.

(C_3) The group G must have a property that disables to guess g from $g^{-1} w g$.

If the group has a normal form for its elements, then the group satisfies this condition.

(C_4) The group G should grow exponentially or at least super-polynomially, that is, the number of elements of length n should grow faster than any polynomial in n.

Observe that this condition is needed to prevent length-based attacks.

Groups are having (C1), (C4), generally (C2), and sometimes nearly (C3). While the word problem is efficiently solvable, the conjugacy search problem is computationally hard.

3.2.2 Protocols Utilizing The Decomposition Problem

The decomposition search problem is a generalization of the conjugacy search problem. It is defined as follows:

Let G be a group, $H \subseteq G$ and $w, w^{'} \in G$. Find two elements $\alpha, \beta \in H$ such that $\alpha \cdot w \cdot \beta = w^{'}$, provided at least one such pair exists.

Also, if we take β as α^{-1}, then the decomposition search problem becomes the conjugacy search problem.

The critical point in the decomposition search problem is that the two elements α and β are elements from H. Without this restriction, choosing $\alpha = 1$ and $\beta = w^{-1}w^{'}$ in G, we can have a straightforward solution.

Notice that solving the decomposition decision problem, i.e., figuring out algorithmically if there exist such $\alpha, \beta \in H$, might be hard for some subsets H. In the paper [14], the authors believe that it can be solvable by brute force attacks on H. Intuitively, it is easier to solve the general version of an equation with two unknowns instead of solving a unique version of the same equation with one unknown. Therefore, we generally believe that solving the decomposition search problem should be more comfortable than solving the conjugacy search problem.

A standard protocol depending on the decomposition search problem is as follows:

Given publicly a group G and two subgroups $A, B \subseteq G$ such that $ab = ba$ for any $a \in A$, $b \in B$.

1. An element $w \in G$ is published.
2. Alice chooses secretly $a_1, a_2 \in A$, and sends the element $a_1 w a_2$ to Bob.
3. Bob chooses secretly $b_1, b_2 \in B$, and sends the element $b_1 w b_2$ to Alice.
4. While Alice computes $K_A = a_1(b_1 w b_2)a_2$, Bob computes his secret key $K_B =$

$b_1(a_1wa_2)b_2$.

Since $a_1b_1 = b_1a_1$ and $a_2b_2 = b_2a_2$ in G, Alice and Bob agree on the shared secret key $K = K_A = K_B$.

3.2.3 Protocols Utilizing The Factorization Search Problem

The factorization search problem is as follows:

Let G be a group, $A, B \leq G$ be two subgroups, and $w \in G$. Find two elements $\alpha \in A$ and $\beta \in B$ such that $\alpha \cdot \beta = w$, provided at least one such pair exists.

A standard protocol based on the factorization search problem is the following:

Given publicly a group G and two subgroups $A, B \leq G$ such that $ab = ba$ for any $a \in A$, $b \in B$.

1. Alice secretly chooses $a_1 \in A$ and $b_1 \in B$. Then, she sends to Bob the element a_1b_1.
2. Bob secretly chooses $a_2 \in A$ and $b_2 \in B$. Then, he sends to Alice the element a_2b_2.
3. While Alice computes $K_A = b_1(a_2b_2)a_1 = a_2b_1a_1b_2 = a_2a_1b_1b_2$, Bob computes $K_B = a_2(a_1b_1)b_2 = a_2a_1b_1b_2$.

Since $ab = ba$ for any $a \in A$, $b \in B$, Alice and Bob agree on the shared secret key $K = K_A = K_B$.

Notice that if $a_1a_2 \neq a_2a_1$ and $b_1b_2 \neq b_2b_1$, then the eavesdropper, Eve, cannot obtain the key from the products $(a_1b_1)(a_2b_2)$ and $(a_2b_2)(a_1b_1)$.

Motivated by this, the factorization decision problem is defined as follows:

Let G be a group, $A, B \leq G$ be two subgroups, and $w \in G$. Decide if there exist two elements $\alpha \in A$ and $\beta \in B$ such that $\alpha \cdot \beta = w$.

As we saw, group theory has uses in cryptography. With some applications, we see that cryptography can affect group theory, as well. This decision problem looks like a new and nontrivial question in group theory and has not been studied so far. So, this

is an example of a group-theoretic question which comes from cryptography.

3.2.4 The Anshel-Anshel-Goldfeld Protocol

We will explain the Anshel-Anshel-Goldfeld protocol given in [1], which seems more secure than the other protocols we described. The reason is that this protocol does not require any commuting group or subgroups. This protocol can use any noncommutative group satisfying (C1) as a platform group. This condition is required for the feasibility of encryption/decryption.

(In this case, a^b stands for $b^{-1}ab$.)

1. Given publicly a group G and elements $a_1, a_2, \ldots, a_h, b_1, b_2, \ldots, b_k \in G$.
2. Alice selects a secret $x \in G$ as a word in $a_1, \ldots, a_h$ (i.e., $x = x(a_1, \ldots, a_h)$) and sends $b_1^x, b_2^x, \cdots, b_k^x$ to Bob.
3. Bob chooses a secret $y \in G$ as a word in $b_1, \ldots, b_k$ (i.e., $y = y(b_1, \ldots, b_k)$) and sends $a_1^y, a_2^y, \ldots, a_h^y$ to Alice.
4. While Alice computes $x(a_1^y, a_2^y, \ldots, a_h^y) = x^y = y^{-1}xy$, Bob computes $y(b_1^x, b_2^x, \ldots, b_k^x) = y^x = x^{-1}yx$.
5. Alice multiplies $y^{-1}xy$ by x^{-1} from left to obtain $K_A = x^{-1}y^{-1}xy = [x^{-1}, y^{-1}]$. Bob firstly multiplies $x^{-1}yx$ by y^{-1} from left, and then takes the inverse of $y^{-1}x^{-1}yx$ to get $K_B = (y^{-1}x^{-1}yx)^{-1} = x^{-1}y^{-1}xy = [x^{-1}, y^{-1}]$.

So, Alice and Bob agree on the shared secret key $K = K_A = K_B$.

In this protocol, we note that solving "the simultaneous conjugacy search problem" for $b_1^x, b_2^x, \cdots, b_k^x; a_1, a_2, \cdots, a_h$ is not enough. The eavesdropper, Eve, should also be able to solve the "membership" search problem (see below) in order to obtain x^y (or y^x) out of $a_1^y, a_2^y, \cdots, a_h^y$ (respectively, or $b_1^x, b_2^x, \cdots, b_k^x$). The reason for that is elements x or y are not just words in the generators of G, but are words in $a_1, a_2, \cdots, a_h$ (respectively, in $b_1, b_2, \cdots, b_k$).

The membership search problem is the following:

Let G be a group and $x, a_1, a_2, \cdots, a_h \in G$. Write (if possible) x as a word in $a_1, a_2, \cdots, a_h$.

Also, the membership decision problem is the following:

Let G be a group and $a_1, a_2, \cdots, a_h \in G$. Decide if given $x \in G$ belongs to the subgroup generated by $a_1, a_2, \cdots, a_h$.

Note that the membership decision problem is unsolvable for many groups. For example, Mihailova in [19] showed that $F_2 \times F_2$ has an algorithmically unsolvable membership decision problem, where F_2 is the free group of rank 2. For example, since a Braid group, B_n, includes subgroups isomorphic to $F_2 \times F_2$, the membership decision problem is not algorithmically solvable.

Also, note that if the eavesdropper, Eve, finds out some $y^{'} \in G$ such that $a_1^y = a_1^{y^{'}}, a_2^y = a_2^{y^{'}}, \cdots, a_h^y = a_h^{y^{'}}$, she cannot be sure that $y = y^{'}$ in G. However, if there is an element, say c_a, satisfying $a_i^{c_a} = c_a^{-1} a_i c_a = a_i$ for all i, i.e., c_a commutes with every element a_i, such that $y^{'} = c_a y$, then Eve can obtain $a_i^y = a_i^{y^{'}}$ for all i. Hence, she can get $a^y = a^{y^{'}}$ for any $a \in \langle a_1, a_2, \cdots, a_h \rangle = A$, in fact, $x^y = x^{y^{'}}$. In this case, the problem becomes whether or not $y^{'}$ (similarly, $x^{'}$) belongs to the subgroup generated by elements $b_1, b_2, \cdots, b_k$, $B = \langle b_1, b_2, \cdots, b_k \rangle$ (respectively, A). If they do not belong to the corresponding groups, then Eve may not get the shared secret key K. However, if they belong to the corresponding groups, then Eve can obtain the key K. In the latter case, since $x^{'}, x \in A$ (similarly, $y^{'}, y \in B$), $c_b \in A$ (respectively, $c_a \in B$). Hence, if $x^{'} = c_b x, y^{'} = c_a y$, where c_a (similarly, c_b) commutes with every elements of A (respectively, B), then

$$(x^{'})^{-1}(y^{'})^{-1}x^{'}y^{'} = (c_b x)^{-1}(c_a y)^{-1} c_b x c_a y = x^{-1} c_b^{-1} y^{-1} c_a^{-1} c_b x c_a y =$$

$$= x^{-1} y^{-1} x y = K$$

So, it follows that Eve can get the key K if and only if c_a and c_b commute. It appears that this is possible only if $x^{'} \in A$ and $y^{'} \in B$. Otherwise, Eve may not get the real shared key.

Therefore, in this protocol, the eavesdropper Eve has two possible choices to obtain a shared secret key. She can solve the conjugacy search problem in the group G to get x and y. Then, she can try to solve either the membership search problem or the membership decision problem. We know that the latter may be impossible. Hence,

Eve has to solve a more complicated variant of the conjugacy search problem, which is given as follows:

Let G be a group, $A \leq G$ be a subgroup, and $w_1, w_2 \in G$. Find an element $x \in A$ such that $x^{-1}w_1x = w_2$, provided that at least one such element exists.

Consequently, there might be some successful attacks on the Anshel-Anshel-Goldfeld protocol. However, we see that even if there is a fast algorithm to solve the conjugacy search problem in braid groups, the Anshel-Anshel-Goldfeld protocol is secure against this attack.

3.3 A Protocol Based on the Thompson's Group F

In the previous section, we saw that different types of search problems lead to various cryptographic schemes. The security of the cryptosystem depends on both the platform group and the problem that it uses. In this section, we will investigate a protocol that uses Thompson's group F as the platform group and is based on the decomposition search problem. This protocol was introduced in [27]. In the next section, we will see that this protocol is highly insecure.

We know from Subsection 3.2.1 that there are some primary conditions for a group to be suitable as a platform group. As explained in Section 2.4, Thompson's group F is well studied in the literature, mainly regarding the amenability problem. So, F satisfies the condition (C0). F grows exponentially by Corollary 2.2.11, also we know that the word problem can be solved easily since reducing a word to the normal form is fast by Theorem 2.3.7. Hence, we see that F satisfies (C1),(C3), and (C4), as well. Thus, Thompson's group F is an excellent candidate to be a platform group. We can generate different words by using one symbol at a time in F, which makes it more suitable compared to groups of numbers. For instance, we should choose big prime numbers to create a key in RSA. Since we cannot make a number prime just by adding one digit, we have to compute before using it. This disables us from having a large key space. Thus, using F as a platform group has many advantages.

Before giving the formal protocol, we will explain and prove some propositions.

Recall from Corollary 2.1.4 that the unique normal form for an element of F is a word of the form

$$X_{q_1}X_{q_2}\cdots X_{q_m}X_{r_n}^{-1}\cdots X_{r_2}^{-1}X_{r_1}^{-1}$$

if the followings hold:
($\mathbf{N}_1$) $q_1 \leq q_2 \leq \cdots \leq q_m$ and $r_1 \leq r_2 \leq \cdots \leq r_n$ and,
($\mathbf{N}_2$) if X_i and X_i^{-1} appear at the same time, then either X_{i+1} or X_{i+1}^{-1} appears.

Define the set A_p, for some fixed $p \in \mathbb{Z}^+$, as the set of elements with lengths of positive and negative parts are equal to each other in their normal forms. In other words, $A_p = \{a \in F \mid a = X_{i_1}\cdots X_{i_k}X_{j_k}^{-1}\cdots X_{j_1}^{-1},$ with $i_l < p+l$ and $j_l < p+l$ for every $l = 1, 2, \ldots, k\}$. Also, define the set B_p, for the same p in A_p, as the subgroup generated by $X_{p+1}, X_{p+2}, \ldots, X_{2p}$.

We will prove several facts about A_p and B_p. These facts will be used in the protocol at the end of this section and also in the next section.

Recall from Section 2.3 that parametric function $\Delta_i(X_n^{\pm 1}) := X_{n+i}^{\pm 1}$ for $i \in \mathbb{Z}$.

Proposition 3.3.1 (Proposition 1, [27]). *Given $\alpha \in A_p$ and $\beta \in B_p$, $\alpha\beta = \beta\alpha$.*

Proof. Let $\alpha = X_{i_1}\cdots X_{i_k}X_{j_k}^{-1}\cdots X_{j_1}^{-1}$ where $i_l < p+l$ and $j_l < p+l$ for every $l = 1, 2, \ldots, k$, and $\beta = X_{m_1}^{r_1}\cdots X_{m_n}^{r_n}$ where $m_s > p$ for every $s = 1, \ldots, n$. We will use induction on k and n to show that the statement $f(k,n) :\ \alpha\beta = \beta\alpha = X_{i_1}\cdots X_{i_k}\cdot(\Delta_k(X_{m_1}^{r_1}\cdots X_{m_n}^{r_n}))\cdot X_{j_k}^{-1}\cdots X_{j_1}^{-1}$ holds.

For the base case, suppose $k = 1$ and $n = 1$. Then, we have $\alpha = X_{i_1}X_{j_1}^{-1}$ and $\beta = X_{m_1}^{r_1}$. By hypothesis, we have $i_1 < p+1$ (similarly, $j_1 < p+1$) and $p < m_1$. So, $p+1 \leq m_1$. Hence, we obtain $i_1 < p+1 \leq m_1$ (similarly, $j_1 < p+1 \leq m_1$). Therefore, we have

$$\alpha\beta = X_{i_1}\underline{X_{j_1}^{-1}X_{m_1}^{r_1}} = X_{i_1}\underline{X_{(m_1+1)}^{r_1}X_{j_1}^{-1}} = X_{i_1}(\Delta_1(\beta))X_{j_1}^{-1},$$

and

$$\beta\alpha = \underline{X_{m_1}^{r_1}X_{i_1}}X_{j_1}^{-1} = \underline{X_{i_1}X_{(m_1+1)}^{r_1}}X_{j_1}^{-1} = X_{i_1}(\Delta_1(\beta))X_{j_1}^{-1}.$$

Suppose that $f(t,1)$ holds for some $t \in \mathbb{N}$. Since we have $i_1, j_1 < p+1 \leq m_1$, we

obtain

$$\alpha\beta = X_{i_1}\cdots X_{i_t}X_{j_t}^{-1}\cdots X_{j_2}^{-1}\underline{X_{j_1}^{-1}X_{m_1}^{r_1}} =$$
$$= X_{i_1}\cdots X_{i_{t+1}}X_{j_{t+1}}^{-1}\cdots X_{j_2}^{-1}\underline{X_{(m_1+1)}^{r_1}X_{j_1}^{-1}},$$

and

$$\beta\alpha = \underline{X_{m_1}^{r_1}X_{i_1}}X_{i_2}\cdots X_{i_t}X_{j_t}^{-1}\cdots X_{j_1}^{-1} =$$
$$= \underline{X_{i_1}X_{(m_1+1)}^{r_1}}X_{i_2}\cdots X_{i_{t+1}}X_{j_{t+1}}^{-1}\cdots X_{j_1}^{-1}.$$

Similarly, since $i_2, j_2 < p+2 \leq (m_1+1)$, again we can follow the same procedure. By continuing like this t-many times for each case, we obtain $i_{(t+1)}, j_{(t+1)} < p+(t+1) \leq (m_1+t)$, and hence we get

$$\alpha\beta = X_{i_1}\cdots X_{i_t}X_{i_{(t+1)}}X_{j_{(t+1)}}^{-1}X_{(m_1+t)}^{r_1}X_{j_t}^{-1}\cdots X_{j_1}^{-1},$$

and

$$\beta\alpha = X_{i_1}\cdots X_{i_t}X_{(m_1+t)}^{r_1}X_{i_{(t+1)}}X_{j_{(t+1)}}^{-1}X_{j_t}^{-1}\cdots X_{j_1}^{-1}.$$

Then, for $k = t+1$, since $i_{(t+1)}, j_{(t+1)} < p+(t+1) \leq (m_1+t)$, we obtain

$$\alpha\beta = X_{i_1}\cdots X_{i_t}X_{i_{(t+1)}}\underline{X_{j_{(t+1)}}^{-1}X_{(m_1+t)}^{r_1}}X_{j_t}^{-1}\cdots X_{j_1}^{-1} =$$
$$= X_{i_1}\cdots X_{i_{t+1}}\underline{X_{(m_1+(t+1))}^{r_1}X_{j_{t+1}}^{-1}}X_{j_t}^{-1}\cdots X_{j_1}^{-1} =$$
$$= X_{i_1}\cdots X_{i_{t+1}}\cdot(\Delta_{t+1}(\beta))\cdot X_{j_{t+1}}^{-1}\cdots X_{j_1}^{-1},$$

and

$$\beta\alpha = X_{i_1}\cdots X_{i_t}\underline{X_{(m_1+t)}^{r_1}X_{i_{(t+1)}}}X_{j_{(t+1)}}^{-1}X_{j_t}^{-1}\cdots X_{j_1}^{-1} =$$
$$= X_{i_1}\cdots X_{i_t}\underline{X_{i_{(t+1)}}X_{(m_1+(t+1))}^{r_1}}X_{j_{(t+1)}}^{-1}X_{j_t}^{-1}\cdots X_{j_1}^{-1} =$$
$$= X_{i_1}\cdots X_{i_{t+1}}\cdot(\Delta_{t+1}(\beta))\cdot X_{j_{t+1}}^{-1}\cdots X_{j_1}^{-1}.$$

Thus, $f(t+1,1)$ holds, as well.

Suppose that the statement $f(g,t)$ is valid for some positive integers g and t. We will show that $f(g,t+1)$ is also true. In this case, we have $\alpha = X_{i_1}\cdots X_{i_g}X_{j_g}^{-1}\cdots X_{j_1}^{-1}$ where $i_l < p+l$ and $j_l < p+l$ for every $l = 1,2,\ldots,g$; and $\beta = X_{m_1}^{r_1}\cdots X_{m_t}^{r_t}X_{m_{(t+1)}}^{r_{(t+1)}}$ where $m_s > p$ for every $s = 1,2,\ldots,t+1$. Then,

$$\alpha\beta = X_{i_1}\cdots X_{i_g}X_{j_g}^{-1}\cdots X_{j_1}^{-1}X_{m_1}^{r_1}\cdots X_{m_t}^{r_t}X_{m_{(t+1)}}^{r_{(t+1)}},$$

and

$$\beta\alpha = X_{m_1}^{r_1} \cdots X_{m_t}^{r_t} X_{m_{(t+1)}}^{r_{(t+1)}} X_{i_1} \cdots X_{i_g} X_{j_g}^{-1} \cdots X_{j_1}^{-1}.$$

Since $f(g,t)$ holds, we already get

$$\alpha\beta = X_{i_1} \cdots X_{i_g} X_{m_1+g}^{r_1} \cdots X_{m_t+g}^{r_t} X_{j_g}^{-1} \cdots X_{j_1}^{-1} X_{m_{(t+1)}}^{r_{(t+1)}}.$$

Since $p < m_{t+1}$ and $j_1 < p+1$, we have $j_1 < p+1 \leq m_{t+1}$. Hence, by continuing in this fashion g-many times to achieve

$$\alpha\beta = X_{i_1} \cdots X_{i_g} X_{m_1+g}^{r_1} \cdots X_{m_{(t+1)}+g}^{r_{(t+1)}} X_{j_g}^{-1} \cdots X_{j_1}^{-1} =$$

$$= X_{i_1} \cdots X_{i_g} \Delta_g(\beta) X_{j_g}^{-1} \cdots X_{j_1}^{-1}.$$

Similarly, for $\beta\alpha$, since each $m_s > p$ and each $i_l < p+l$, we can apply the similar process g-many times to every X_{m_s} in β. Hence, we obtain

$$\beta\alpha = X_{i_1} \cdots X_{i_g} X_{m_1+g}^{r_1} \cdots X_{m_{(t+1)}+g}^{r_{(t+1)}} X_{j_g}^{-1} \cdots X_{j_1}^{-1} =$$

$$= X_{i_1} \cdots X_{i_g} \Delta_g(\beta) X_{j_g}^{-1} \cdots X_{j_1}^{-1}.$$

Thus, we conclude that $\alpha\beta = \beta\alpha$ in group F for any $\alpha \in A_p$ and $\beta \in B_p$. □

Proposition 3.3.2 (Proposition 2, [27]). *Let $p \geq 2$ be a natural number. The set A_p is a subgroup of F, and $A_p = \langle X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1} \rangle$.*

Proof. Since $A_p = \{a \in F \mid a = X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1},$ with $i_l < p+l,$ and $j_l < p+l$ for every $l = 1, 2, \ldots, k\}$, clearly we have $1 \in A_p$ and

$$a^{-1} = (X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1})^{-1} = X_{j_1} \cdots X_{j_k} X_{i_k}^{-1} \cdots X_{i_1}^{-1} \in A_p.$$

Now, we will show that A_p is closed under multiplication. To show this, we will use the algorithms given in Section 2.3. Let $u = X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1}$ and $v = X_{m_1} \cdots X_{m_l} X_{n_l}^{-1} \cdots X_{n_1}^{-1}$ with $i_r < p+r, j_r < p+r, m_s < p+s$, and $n_s < p+s$ for every $r = 1, \ldots, k$, and $s = 1, \ldots, l$ be two arbitrary normal forms from A_p. We want to show that the normal form of uv is in A_p, too.

Notice that the lengths of positive and negative parts in uv are equal to each other. When we apply the relation of F to uv, the number of negative and positive letters

that are removed is equal to each other. Therefore, the lengths of negative and positive parts in the normal form of uv are equal to each other. So, we only need to show that indices in the normal form of uv satisfy the relation that such as $i_r < p+r, j_r < p+r$ for every $r = 1, \ldots, k$ in u. We see that negative and positive parts are next to each other in the product uv as

$$uv = X_{i_1} \cdots X_{i_k}(X_{j_k}^{-1} \cdots X_{j_1}^{-1} X_{m_1} \cdots X_{m_l}) X_{n_l}^{-1} \cdots X_{n_1}^{-1}.$$

By using Algorithm 1 given in Section 2.3, we can convert this part into a seminormal form, say w. So, w is the product $w = ab$ where a is a positive word, and b is a negative word. Clearly, one can prove that a and b satisfy A_p's relation by using induction on $k+l$. Then, by rearranging the terms in a and b to the order of the indices satisfying $(\mathbf{N}_1)$ from Section 2.3, we get the normal forms of a and b, say $a^{'}$ and $b^{'}$, respectively. With the induction on the number of operations used in rearrangements, it is clear that $a^{'}$ and $b^{'}$ satisfy the relation for A_p. Hence, the word $w^{'} = a^{'}b^{'}$ satisfies A_p's relation, and it is a seminormal form of uv. Now, by using Algorithm 4 from Section 2.3, we can convert a seminormal form of $w^{'}$ into the normal form of $w^{'}$. By the induction on the number of pairs contradicting to $(\mathbf{N}_2)$ from Section 2.3, one can observe that the normal form of $w^{'}$ satisfies A_p's relation. Thus, $uv \in A_p$, that is, A_p is closed under multiplication.

Consequently, the set A_p is a subgroup of F.

Now, we will prove that the subgroup A_p is generated by the elements

$$\{X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\}.$$

Denote $H = \langle X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\rangle$.

Claim: $A_p = H$

It is clear that the number of negative and positive parts are equal for any word in H. Also, we have $0 < p$ and $1 < p+1, 2 < p+2, \ldots, p < p+p$. Hence, A_p contains elements $\{X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\}$. To show that $u \in H$ for any element $u \in A_p$, we need the following lemma.

Lemma 3.3.3. *For every element $h \in H$, we obtain $X_0hX_0^{-1} \in H$.*

Proof. We will use $h = X_0X_i^{-1}$ where $i = 1, \ldots, p$ to show $X_0hX_0^{-1} \in H$. **Case** 1**:** $i \geq 2$

Since $i \geq 2$ and $0 < 2$, we have,

$$X_0hX_0^{-1} = X_0\underline{X_0X_i^{-1}}X_0^{-1} = X_0\underline{X_{i-1}^{-1}X_0}X_0^{-1} = X_0X_{i-1}^{-1} \in H$$

Case 2**:** $i = 1$, i.e., $h = X_0X_1^{-1}$

In this case, multiply some parts of $X_0hX_0^{-1}$ by $X_2^{-1}X_2$ to obtain

$$y = X_0hX_0^{-1} = X_0X_0X_1^{-1}X_0^{-1} = \underline{X_0(X_2^{-1}}X_2)X_0X_1^{-1}(X_2^{-1}\underline{X_2)X_0^{-1}}$$

Clearly, $X_0X_2^{-1}$ and $(X_0X_2^{-1})^{-1} = X_2X_0^{-1}$ belong to H. Hence, $y \in H$ if and only if $X_2X_0X_1^{-1}X_2^{-1} \in H$. Since $1 < 2$, we obtain $X_2X_0\underline{X_1^{-1}X_2^{-1}} = X_2X_0\underline{X_3^{-1}X_1^{-1}}$. Also, since we have $0 < 2$, we obtain $\underline{X_2X_0}X_3^{-1}X_1^{-1} = \underline{X_0X_3}X_3^{-1}X_1^{-1} = X_0X_1^{-1}$. It is clear that $X_0X_1^{-1} \in H$. Hence, $y \in H$. Thus, for any $h \in H$, we have $X_0hX_0^{-1} \in H$. □

Proof of the Claim:

Let $u = X_{i_1} \cdots X_{i_k}X_{j_k}^{-1} \cdots X_{j_1}^{-1} \in A_p$ where $i_l < p + l$ and $j_l < p + l$ for every $l = 1, 2, \ldots, k$.

We will prove the claim using induction on k.

Case 1**:** $k = 1$, i.e., $u = X_{i_1}X_{j_1}^{-1}$

When we put $X_0^{-1}X_0$ in the product $X_{i_1}X_{j_1}^{-1}$, we obtain

$$u = X_{i_1}X_0^{-1}X_0X_{j_1}^{-1} = (X_0X_{i_1}^{-1})^{-1}(X_0X_{j_1}^{-1}).$$

Since both $(X_0X_{i_1}^{-1})^{-1}$ and $X_0X_{j_1}^{-1}$ belongs to H as $i_1 < p + 1$ and $j_1 < p + 1$, we have $u \in H$ for $k = 1$.

Case 2**:**

Suppose that the claim holds for some positive integer $k > 1$. In other words, we have $u = X_{i_1} \cdots X_{i_k}X_{j_k}^{-1} \cdots X_{j_1}^{-1} \in H$ where $i_l < p + l$ and $j_l < p + l$ for every $l = 1, 2, \ldots, k$. If we again multiply the product by $X_0^{-1}X_0$, then we obtain $u =$

$X_{i_1}(X_0^{-1}\underline{X_0)X_{i_2}\cdots X_{i_k}X_{j_k}^{-1}\cdots X_{j_2}^{-1}(X_0^{-1}}X_0)X_{j_1}^{-1}$. However, we already know that $X_{i_1}X_0^{-1} = (X_0X_{i_1}^{-1})^{-1}$ and $X_0X_{j_1}^{-1}$ belong to H. Also, by the induction hypothesis, we have that $X_{i_2}\cdots X_{i_k}X_{j_k}^{-1}\cdots X_{j_2}^{-1} \in H$. Therefore, by Lemma 3.3.3, we obtain that $X_0X_{i_2}\cdots X_{i_k}X_{j_k}^{-1}\cdots X_{j_2}^{-1}X_0^{-1} \in H$. Since all parts in the product belong to H, we achieve that $u \in H$ for any $u \in A_p$.

Thus, we conclude that $A_p = \langle X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\rangle$. □

We know from Section 2.1 that the element $X_n \in F$ can be obtained by taking the identity function for the interval $[0, 1-\frac{1}{2^n}]$, and then, by "compressing" the function $A = X_0$ to the interval $[1-\frac{1}{2^n}, 1]$. (One can see Figure 2.10.) Define $\gamma_n = 1-\frac{1}{2^{n+1}}$ for every $n \in \mathbb{Z}^+$. Therefore, we have $X_n^{-1}([\gamma_n, 1]) = [\gamma_{n+1}, 1] \subseteq [\frac{3}{4}, 1]$. Hence, for every $t \in [\gamma_n, 1]$, we obtain that $(X_0X_n^{-1})(t) = t$ since the derivative $\frac{d}{dt}(X_0(X_n^{-1}(t))) = X_0'(X_n^{-1}(t))(X_n^{-1})'(t) = 2\cdot\frac{1}{2} = 1$. Hence, the support of $X_0X_n^{-1}$ lies in $[0, \gamma_n]$.

While $A_p = \langle X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\rangle$, $B_p = \langle X_{p+1}, X_{p+2}, \ldots, X_{2p}\rangle$ for any fixed $p \in \mathbb{Z}^+$. Therefore, we clearly see that A_p and B_p have disjoint supports as A_p is identity only on the interval $[\gamma_p, 1]$ and B_p is identity only on $[0, \gamma_p]$. Thus, A_p and B_p commute.

Notice that if $p = 1$, we have $A_1 = \langle X_0X_1^{-1}\rangle$, which is a cyclic group. When p gets bigger, A_p becomes the full piecewise linear homeomorphism group on the interval $[0, \gamma_p]$.

We will denote by $PL_2([a, b])$ the subgroup of F consisting of elements with support in $[a, b]$, $0 \le a < b \le 1$.

Lemma 3.3.4 (Lemma 6.1, [17]). $A_2 = \langle X_0X_1^{-1}, X_0X_2^{-1}\rangle = PL_2([0, \frac{7}{8}]) = PL_2([0, \gamma_2])$.

Proof. It is clear that $A_2 \subseteq PL_2([0, \frac{7}{8}])$ since $X_0X_1^{-1} \in PL_2([0, \frac{7}{8}])$ and $X_0X_2^{-1} \in PL_2([0, \frac{7}{8}])$.

Let $\alpha = X_0^2X_1^{-1}X_0^{-1}$ and $\beta = X_0X_1^2X_2^{-1}X_1^{-1}X_0^{-1}$ be the two generators of $PL_2([0, \frac{1}{2}])$.

Conjugation of $PL_2([0, \frac{1}{2}])$ by X_0^2 gives us

$$PL_2([0, \frac{7}{8}]) = \langle X_0^{-2}\alpha X_0^2, X_0^{-2}\beta X_0^2\rangle.$$

Since $X_0^{-2}\alpha X_0^2 = X_1^{-1}X_0 = X_0X_2^{-1}$ and

$$X_0^{-2}\beta X_0^2 = X_0^{-1}X_1^2X_2^{-1}X_1^{-1}X_0 = X_2^2X_3^{-1}X_2^{-1}$$

satisfy the conditions of A_p for $p = 2$, we obtain that both $X_0^{-2}\alpha X_0^2$ and $X_0^{-2}\beta X_0^2$ belong to A_2. Hence, $PL_2([0, \frac{7}{8}]) \subseteq A_2$. □

Theorem 3.3.5 (Theorem 6.2, [17]). *For all $p \geq 2$, $A_p = PL_2([0, \gamma_p])$, where $\gamma_p = 1 - \frac{1}{2^{(p+1)}}$.*

Proof. It is clear that $X_0^{-1}(PL_2([0, \gamma_p]))X_0 = PL_2([0, \gamma_{p+1}])$ for all $p \geq 0$. Hence, since $A_2 = PL_2([0, \gamma_2])$, and by the definition of A_p, we have that $PL_2([0, \gamma_p]) = X_0^{2-p}A_2X_0^{p-2} \subseteq A_p \subseteq PL_2([0, \gamma_p])$. Thus, we obtain $A_p = PL_2([0, \gamma_p])$. □

Corollary 3.3.6 (Corollary 6.3, [17]). *$A_p \cong B_p \cong F$ for all $p \geq 2$.*

The next proposition is a summary of facts about A_p and B_p.

Proposition 3.3.7.

1. *All the elements of A_p have normal forms as follows:*

$$X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1},$$

   where $i_l < p + l$ and $j_l < p + l$ for all $l = 1, 2, \ldots, k$.

2. *$A_p = \langle X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1} \rangle \leq F$, for any fixed $p \in \mathbb{Z}^+$.*

3. *$B_p = \langle X_{p+1}, X_{p+2}, \ldots, X_{2p} \rangle \leq F$, for any fixed $p \in \mathbb{Z}^+$.*

4. *$A_p = PL_2([0, \gamma_p])$ for all $p \geq 2$.*

5. *$B_p = PL_2([\gamma_p, 1])$.*

6. *A_p and B_p commute.*

7. *Given $\alpha \in A_p$ and $\beta \in B_p$ in the normal forms as*

$$\alpha = X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1},$$

   and

$$\beta = X_{m_1} \cdots X_{m_n} X_{r_s}^{-1} \cdots X_{r_1}^{-1}.$$

   Then the normal form of $\alpha\beta$ in A_pB_p is

$$\alpha\beta = X_{i_1} \cdots X_{i_k} \cdot (X_{m_1+k} \cdots X_{m_n+k} X_{r_s+k}^{-1} \cdots X_{r_1+k}^{-1}) \cdot X_{j_k}^{-1} \cdots X_{j_1}^{-1}.$$

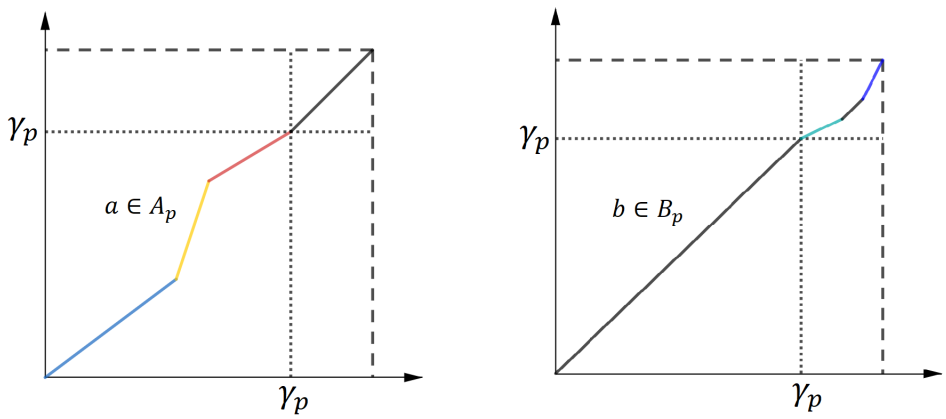


Figure 3.1: An element from A_p and B_p where $\gamma_p = 1 - \frac{1}{2^{p+1}}$

8. *$A_p \cong B_p \cong F$ for all $p \geq 2$.*

Theorem 3.3.5 and Lemma 2.2.4 imply the following:

Corollary 3.3.8. *There exists an element $\alpha \in A_p$ with $\alpha(t_1) = t_2$ for any $t_1, t_2 \in \mathbb{Z}[\frac{1}{2}] \cap [0, \gamma_p]$, where $\mathbb{Z}[\frac{1}{2}]$ is the ring of dyadic rational numbers.*

Corollary 3.3.9. *Given $t_0 \in \mathbb{Z}[\frac{1}{2}] \cap [0, \gamma_p]$ and $\overline{\alpha}(t) = \alpha|_{[0,t_0]}$ for an $\alpha \in A_p$. Suppose we know $\overline{\alpha}$, but we do not know α. Then, there exists an $\alpha_\sigma \in A_p$ such that $\alpha_\sigma(t) = \overline{\alpha}(t)$ for every $t \in [0, \gamma_p]$.*

(Similarly, the last two corollaries hold for the interval $[0, \gamma_p]$ and B_p.)

Now, we will describe the formal protocol using F as a platform group with the decomposition search problem as follows:

1. Set two numbers $p, L \in \mathbb{Z}^+$ and a word w in $X_0, X_1, \ldots$, that is, $w = w(X_0, X_1, \ldots)$.
2. Alice randomly chooses secret $a_1 \in A_p$ and $b_1 \in B_p$. Then, she finds the normal form of the element $a_1 w b_1$, and sends the normal form to Bob.
3. Bob randomly chooses secret $a_2 \in A_p$ and $b_2 \in B_p$. Then, he finds the normal form of the element $b_2 w a_2$, and sends the normal form to Alice.
4. While Alice computes $K_A = a_1 b_2 w a_2 b_1 = b_2 a_1 w b_1 a_2$, Bob computes $K_B = b_2 a_1 w b_1 a_2$. Since $a_i b_i = b_i a_i$ in F, Alice and Bob has a common secret key $K = K_A = K_B$.

We know that computational effort is significant in key agreement protocols as much as the security of the protocol. To provide both safety and efficiency, there are some suggestions about parameters given in [27] as follows:

1. Choose (randomly and uniformly) the number p from interval $[3, 8]$, and choose the positive integer L from set $\{256, 258, 260, \ldots, 318, 320\}$.

2. Choose publicly known word w as a product of generators from $S_W^{\pm 1}$ where $S_W = \{X_0, X_1, \ldots, X_{p+2}\}$. To choose w in this fashion, we start with the empty word w_0. Then, we multiply it from the right by a generator from $S_W^{\pm 1}$ to get w_1. Similarly, multiply w_1 from the right by a generator of $S_W^{\pm 1}$, and then, find the normal form to obtain w_2. Therefore, if we have a word, say w_i, we simply multiply it from the right by a generator of $S_W^{\pm 1}$, and then we find the normal form of the product to obtain w_{i+1}. We keep following this process unless we reach out that the length of the obtained word w_{i+1} is L.

3. Choose private elements a_1, a_2 as products of words from $S_A^{\pm 1}$ where the set $S_A = \{X_0X_1^{-1}, X_0X_2^{-1}, \ldots, X_0X_p^{-1}\}$. To choose a_1 or a_2 in this fashion, we start with the empty word u_0. Then, we multiply it from the right by a random word from $S_A^{\pm 1}$ to get u_1. Similarly, multiply u_1 from the right by a word of $S_A^{\pm 1}$, and then, find the normal form to obtain u_2. Therefore, if we have a word, say u_i, we simply multiply it from the right by a randomly chosen word from $S_A^{\pm 1}$, and then we find the normal form of the product to obtain u_{i+1}. We keep following this process unless we reach out that the length of the obtained word u_{i+1} is L.

4. Choose private elements b_1, b_2 as products of generators from $S_B^{\pm 1}$ where the set $S_B = \{X_{p+1}, X_{p+2}, \ldots, X_{2p}\}$. To choose b_1 or b_2 in this fashion, we start with the empty word v_0. Then, we multiply it from the right by a generator from $S_B^{\pm 1}$ to get v_1. Similarly, multiply v_1 from the right by a generator of $S_B^{\pm 1}$, and then, find the normal form to obtain v_2. Therefore, if we have a word, say v_i, we simply multiply it from the right by a generator of $S_B^{\pm 1}$, and then we find the normal form of the product to obtain v_{i+1}. We keep following this process unless we reach out that the length of the obtained word v_{i+1} is L.

Recall that by Theorem 2.2.10, the submonoid generated by $t = X_0X_1^{-1} = AB^{-1} \in A_p$ and $z = X_0X_2^{-1} = B^{-1}A \in A_p$ is free. Thus, the number of elements in A_p of length at most L, grows exponentially with L. This follows from the fact that for a word of length L, there are at least $2^{L/2} = \sqrt{2}^L$-many possibilities. Hence, we obtain $|A_p(L)| \geq \sqrt{2}^L$. We know that the key space contains $A_p(L)$, which consists of the words of length at least L. Thus, the key space grows exponentially with L.

The suggestions given above are against brute force attacks, also known as length based attacks. We will now describe one particular such attack suggested by Shpilrain and Ushakov in [27].

Define a directed labelled graph $\Omega = (V(\Omega), E(\Omega))$ where $V(\Omega) = F$ is the set of vertices and $E(\Omega)$ is the set of edges. $E(\Omega)$ consists of edges $v_1 \xrightarrow{(e_1,e_2)} v_2$ in F, where $e_1 \in S_A^{\pm 1}$, and $e_2 \in S_B^{\pm 1}$. Therefore, if there exist two elements $w, w^{'} \in F$ such that $a_1wb_1 = w^{'}$, where $a_1 \in S_A^{\pm 1}$ and $b_1 \in S_B^{\pm 1}$, then this means that there is a path in Ω connecting the vertices w and $w^{'}$. So, w and $w^{'}$ are in the same connected component of Ω. Denote this connected component by $\Omega_{w^{'}} = \Omega_w$.

In the protocol above, Alice sends a_1wb_1 to Bob. Hence, there exists a path between w and a_1wb_1, i.e. $\Omega_w = \Omega_{a_1wb_1}$. Since Eve knows a_1wb_1 and w, it is enough to find a path between w and a_1wb_1 in Ω_w to obtain Alice's key.

In [27], V. Shpilrain and A. Ushakov experimented tests on the protocol to break it. They used graphs as above to generate keys with an algorithm given below, Algorithm 5. The algorithm stops if it finds a path between w and $w^{'}$. Otherwise, it continues to build Ω_w and $\Omega_{w^{'}}$. The algorithm collects the parts that are built in the sets denoted by P_w and $P_{w^{'}}$. Also, Algorithm 5 collects the successful vertices of the path in the sets $R_w \subset P_w$ and $R_{w^{'}} \subset P_{w^{'}}$.

Algorithm 5 [Algorithm 1, [27]] Brute force attack
INPUT. *The publicly known words of Alice,* w *and* $w^{'}$
OUTPUT. *Words* $x_1 \in \langle S_A \rangle$ *and* $x_2 \in \langle S_B \rangle$ *satisfying* $w^{'} = x_1wx_2$.
INITIALIZATION. *Set* $P_w = \{w\}$, $P_{w^{'}} = w^{'}$, $R_w = \varnothing$, *and* $R_{w^{'}} = \varnothing$.
COMPUTATIONS.
1) *Find the shortest word* $u \in P_w \backslash R_w$.

2) *Multiply u on the right (respectively, left) by elements $S_B^{\pm 1}$ (respectively, $S_A^{\pm 1}$), and put every result in P_w with edges labeled in order.*
3) *Add u into R_w.*
4) *Repeat the same processes 1-3 with $P_{w'}$ and $R_{w'}$ accordingly.*
5) *If $P_w \cap P_{w'} = \varnothing$, then go to step 1.*
6) *If there exists $\overline{w} \in P_w \cap P_{w'}$, then detect a path in P_w from w to $\overline{w}$ and detect a path in $P_{w'}$ from $\overline{w}$ to w'. Connect them according to order and output the label of the result.*

Shpilrain and Ushakov [27] tested the protocol with this length-based attack, i.e. Algorithm 5, many times. They observed that the success rate of this algorithm was 0. In other words, the protocol is secure against this brute force attack. However, we will explain in the next section that how Francesco Matucci [17] showed that this protocol is not secure at all.

3.4 Cryptanalysis of a Protocol for Thompson's group F

In this section, we will follow Matucci [17] to show that the protocol of the previous section is highly insecure.

The eavesdropper, Eve, can acquire any private key of the two communicating parties, Alice and Bob rather easily. Eve knows public data w, u_1, u_2 where $u_1 = a_1 w b_1$ sent by Alice and $u_2 = b_2 w a_2$ sent by Bob in the protocol. She chooses whose key to obtain based on the graph of w. If $w(\gamma_p) \leq \gamma_p$, she can obtain Bob's key. Otherwise, she can obtain Alice's key. We first explain how to get Bob's key in the case $w(\gamma_p) \leq \gamma_p$:

By Proposition 3.3.7, we have $b_2(t) = t$ and $a_2(t) \leq \gamma_p$ for all $t \in [0, \gamma_p]$. Also, $w(\gamma_p) \leq \gamma_p$ implies $w(t) \leq \gamma_p$ for all $t \in [0, \gamma_p]$. Therefore, we obtain that $u_2(t) = (b_2 w a_2)(t) = (w a_2)(t)$ for all $t \in [0, \gamma_p]$. Hence, Eve can acquire a_2 by multiplying u_2 on the left with w^{-1}. Thus, by Proposition 3.3.7, $w^{-1} u_2 \in A_p B_p$ and

$$a_2(t) = \begin{cases} w^{-1} u_2(t), & t \in [0, \gamma_p] \\ t, & t \in [\gamma_p, 1] \end{cases}$$

So, Eve has the elements a_2, w, and u_2. Finally, she computes $u_2 a_2^{-1} w^{-1}$ to obtain b_2. Hence, she gets a_2, b_2, which is the private information of Bob and $b_2 a_1 w b_1 a_2$, which is the shared secret key K.

Secondly, we explain how to get Alice's key in the case $w(\gamma_p) > \gamma_p$:

By Proposition 3.3.7, we have $b_1^{-1}(t) = t$ and $a_1^{-1}(t) < \gamma_p$ for all $t \in [0, \gamma_p]$. Also, $w^{-1}(\gamma_p) < \gamma_p$ implies $w^{-1}(t) < \gamma_p$ for all $t \in [0, \gamma_p]$. Therefore, we obtain that $u_1^{-1}(t) = (b_1^{-1} w^{-1} a_1^{-1})(t) = (w^{-1} a_1^{-1})(t)$ for all $t \in [0, \gamma_p]$. Hence, Eve can acquire $a_1^{-1}(t)$ by multiplying $u_1^{-1}(t)$ on the left with w for all $t \in [0, \gamma_p]$, and gets $(w u_1^{-1})^{-1} = u_1 w^{-1} \in A_p B_p$. Thus, Eve can get a_1 and b_1, which are the private keys of Alice, and $a_1 b_2 w a_2 b_1$, which is the shared secret key K. Also, Eve observes that since $a_1(t) = t$ for all $t \in [\gamma_p, 1]$, she gets $w^{-1} u_1(t) = w^{-1} a_1 w b_1(t) = b_1(t)$ for all $t \in [\gamma_p, 1]$. Hence, she gets

$$b_1(t) = \begin{cases} t, & t \in [0, \gamma_p] \\ w^{-1} u_1(t), & t \in [\gamma_p, 1] \end{cases}$$

Recall from Proposition 3.3.7 that if $\alpha \in A_p$ and $\beta \in B_p$ are given in the normal forms as

$$\alpha = X_{i_1} \cdots X_{i_k} X_{j_k}^{-1} \cdots X_{j_1}^{-1},$$

and

$$\beta = X_{m_1} \cdots X_{m_n} X_{r_s}^{-1} \cdots X_{r_1}^{-1},$$

then the normal form of $\alpha\beta$ in $A_p B_p$ is

$$\alpha\beta = X_{i_1} \cdots X_{i_k} \cdot (X_{m_1+k} \cdots X_{m_n+k} X_{r_s+k}^{-1} \cdots X_{r_1+k}^{-1}) \cdot X_{j_k}^{-1} \cdots X_{j_1}^{-1}.$$

Now, we explain a combinatorial attack built on the above observations. Eve has the public elements u_1, u_2, w, and also, the number p.

1. Eve finds the normal forms of $k_1 := u_1 w^{-1}$ and $k_2 := w^{-1} u_2$.

2. We know from the above observations that at least one of k_1 and k_2 belongs to $A_p B_p$. Eve can figure out which one belongs, by using the form of normal forms of elements in $A_p B_p$. Then, she chooses this k_i.

3. She computes the A_p-part a_{k_i} of k_i.

4. If she gets a_{k_1} then she computes $b_{k_1} := w^{-1}a_{k_1}^{-1}u_1$. If she gets a_{k_2} then she computes $b_{k_2} := u_2 a_{k_2}^{-1} w^{-1}$.

5. Eve acquires K from $u_1, u_2, a_{k_i}, b_{k_i}$.

The above procedure is all clear except (2). We know that at least one of the normal forms of k_1 and k_2 belongs to A_pB_p. Say the normal form is

$$k_i = X_{g_1} \cdots X_{g_e} X_{h_f}^{-1} \cdots X_{h_1}^{-1}.$$

We determine the smallest index s such either g_{s+1} or h_{s+1} does not satisfy the index relation of A_p. Then, cut out the first and last s letters of k_i, and decrease all the indices of the remaining part by s. In this way, we can look at the indices of the final word to see whether the final word belongs to B_p or not. If the indices belong to the set $\{p+1, p+2, \ldots, 2p\}$, then the word belongs to B_p. Hence, $k_i \in A_pB_p$. Otherwise, the word does not belong to B_p, and then, $k_i \notin A_pB_p$. Also, observe that if $k_i \in A_pB_p$, then A_p-part a_{k_i} will be the product of the first and the last s letters of k_i.

It is clear that this attack can be performed in time $O(L \cdot logL)$, where L is the same length used in the protocol of [27].

We know how to obtain one's private keys and the shared secret key by observing the graph of w. However, in that case, we have to choose one party to attack, depending on the graph. Now, we will show that we do not need to choose one party. In other words, we can attack Bob's key even if $w(\gamma_p) > \gamma_p$, and Alice's key even if $w(\gamma_p) \leq \gamma_p$. To do so, we will use the description of the subgorup A_p given in Theorem 3.3.5. We first explain how to get Alice's key in the case $w(\gamma_p) \leq \gamma_p$:

By Proposition 3.3.7, we have $b_1(t) = t$ for all $t \in [0, \gamma_p]$. Therefore, we get $u_1(t) = a_1 w b_1(t) = a_1 w(t)$ for all $t \in [0, \gamma_p]$. Hence, $a_1(t) = u_1 w^{-1}(t)$ for all $t \in [0, w(\gamma_p)]$. So, a_1 is uniquely determined by its values in $[0, w(\gamma_p)]$. Thus, by Corollary 3.3.9, there exists an $a_\sigma \in A_p$ such that $a_\sigma = a_1$ on the interval $[0, w(\gamma_p)]$. Define b_σ as $b_\sigma := w^{-1}{a_\sigma}^{-1}u_1$, so that we obtain $b_\sigma(t) = w^{-1}{a_\sigma}^{-1}a_1 w(t) = w^{-1}w(t) = t$ for

every $t \in [0, \gamma_p]$. Thus, $b_\sigma \in B_p$ and $a_\sigma w b_\sigma = u_1$, which enable Eve to acquire the shared secret key K.

Secondly, we explain how to get Bob's key in the case $w(\gamma_p) > \gamma_p$:

By Proposition 3.3.7, we have $b_2^{-1}(t) = t$ and $a_2^{-1}(t) < \gamma_p$ for all $t \in [0, \gamma_p]$. Therefore, we get $u_2^{-1}(t) = a_2^{-1} w^{-1} b_2^{-1}(t) = a_2^{-1} w^{-1}(t)$ for all $t \in [0, \gamma_p]$. Hence, $a_2^{-1}(t) = u_2^{-1} w(t)$ for all $t \in [0, w^{-1}(\gamma_p)]$. So, a_2^{-1} is uniquely obtained in $[0, w^{-1}(\gamma_p)]$. Hence, by Corollary 3.3.9, there exists an $a_\sigma \in A_p$ such that ${a_\sigma}^{-1} = a_2^{-1}$ on the interval $[0, w^{-1}(\gamma_p)]$. Define ${b_\sigma}^{-1}$ as ${b_\sigma}^{-1} := w a_\sigma u_2^{-1}$, so that we obtain $b_\sigma(t)^{-1} = w a_\sigma a_2^{-1} w^{-1} = w^{-1} w(t) = t$ for every $t \in [0, \gamma_p]$. Thus, ${b_\sigma}^{-1} \in B_p$ and $(a_\sigma^{-1} w^{-1} b_\sigma^{-1})^{-1} = b_\sigma w a_\sigma = u_2$, which enable Eve to acquire the shared secret key K.

In conclusion, we see that Matucci [17] shows that an eavesdropper can easily obtain the shared secret key K of the protocol suggested by Shpilrain and Ushakov in [27].

CHAPTER 4

CONCLUSION

In this thesis, we have studied the Thompson's group F and group-based cryptography within two chapters. In the first chapter, we have investigated Thompson's group F and some of its properties. In Section 2.1, we defined F and gave some basic properties of its elements such as tree representations, reducibility, and normal forms. In Section 2.2, we proved F's algebraic properties, its subgroups, and quotients. Also, we have proved the simplicity of $F^{'}$. In Section 2.3, we investigated the word problem in F and gave an algorithm to find normal forms in F. In Chapter 3, we explained general notions of cryptography and analyzed a cryptographic protocol based on the group F. In Section 3.1, we explored fundamentals of public key cryptography. In Section 3.2, we focused on how nonabelian groups can be used in public key cryptography, and we gave several protocols based on several search problems. In Section 3.3, we analyzed the protocol, suggested by V. Shpilrain and A. Ushakov [27], based on the group F. In Section 3.4, we explored the cryptanalysis of Matucci [17] against the protocol of Shpilrain and Ushakov [27], and we concluded that the protocol is highly insecure. Hence, there is a need for proposing safe, efficient cryptosystems based on Thompson's group F using new ideas.

REFERENCES

[1] I. Anshel, M. Anshel, and D. Goldfeld. An algebraic method for public-key cryptography. *Math. Res. Lett.*, 6(3-4):287–291, 1999.

[2] K. S. Brown. Finiteness properties of groups. In *Proceedings of the Northwestern conference on cohomology of groups (Evanston, Ill., 1985)*, volume 44, pages 45–75, 1987.

[3] J. Buchmann. *Introduction to cryptography*. Undergraduate Texts in Mathematics. Springer-Verlag, New York, second edition, 2004.

[4] J. Burillo. *Introduction to Thompson's Group F*. Available at `https://web.mat.upc.edu/pep.burillo/F%20book.pdf`.

[5] J. W. Cannon, W. J. Floyd, and W. R. Parry. Introductory notes on Richard Thompson's groups. *Enseign. Math. (2)*, 42(3-4):215–256, 1996.

[6] M. Dehn. *On Infinite Discontinuous Groups*, pages 133–178. Springer New York, New York, NY, 1987.

[7] H. Delfs and H. Knebl. *Introduction to cryptography*. Information Security and Cryptography. Springer, Heidelberg, third edition, 2015. Principles and applications.

[8] N. Dershowitz. A taste of rewrite systems. In *Functional Programming, Concurrency, Simulation and Automated Reasoning*, pages 199–228. Springer, 1993.

[9] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Inform. Theory*, IT-22(6):644–654, 1976.

[10] S. M. Gersten and J. R. Stallings. *Combinatorial group theory and topology*. Number 111. Princeton University Press, 1987.

[11] D. Grigoriev. Key-agreement based on automaton groups rostislav grigorchuk. 2019.

[12] V. Guba and M. Sapir. The dehn function and a regular set of normal forms for r. thompson's group f. *Journal of The Australian Mathematical Society - J AUST MATH SOC*, 62, 06 1997.

[13] G. Higman. On infinite simple permutation groups. *Publ. Math. Debrecen*, 3:221–226 (1955), 1954.

[14] K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J.-s. Kang, and C. Park. New public-key cryptosystem using braid groups. In M. Bellare, editor, *Advances in Cryptology — CRYPTO 2000*, pages 166–183, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[15] D. Liu. *Next Generation SSH2 Implementation: Securing Data in Motion*. Elsevier Science, 2011.

[16] A. Mann. *How groups grow*, volume 395 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2012.

[17] F. Matucci. Cryptanalysis of the Shpilrain-Ushakov protocol for Thompson's group. *J. Cryptology*, 21(3):458–468, 2008.

[18] R. McKenzie and R. J. Thompson. An elementary construction of unsolvable word problems in group theory. In *Word problems: decision problems and the Burnside problem in group theory (Conf., Univ. California, Irvine, Calif. 1969; dedicated to Hanna Neumann)*, volume 71 of *Studies in Logic and the Foundations of Math.*, pages 457–478. 1973.

[19] K. A. Mihaĭlova. The occurrence problem for direct products of groups. *Dokl. Akad. Nauk SSSR*, 119:1103–1105, 1958.

[20] C. F. Miller, III. Decision problems for groups—survey and reflections. In *Algorithms and classification in combinatorial group theory (Berkeley, CA, 1989)*, volume 23 of *Math. Sci. Res. Inst. Publ.*, pages 1–59. Springer, New York, 1992.

[21] A. Myasnikov, V. Shpilrain, and A. Ushakov. *Group-based cryptography*. Advanced Courses in Mathematics. CRM Barcelona. Birkhäuser Verlag, Basel, 2008.

[22] J. Neumann. Zur allgemeinen theorie des masses. *Fundamenta Mathematicae*, 13(1):73–116, 1929.

[23] P. S. Novikov. *Ob algoritmičeskoĭ nerazrešimosti problemy toždestva slov v teorii grupp*. Trudy Mat. Inst. im. Steklov. no. 44. Izdat. Akad. Nauk SSSR, Moscow, 1955.

[24] A. Y. Olshanskii. On a geometric method in the combinatorial group theory. In *Proceedings of the International Congress of Mathematicians, Vol. 1, 2 (Warsaw, 1983)*, pages 415–424. PWN, Warsaw, 1984.

[25] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, Feb. 1978.

[26] V. Shpilrain. Assessing security of some group based cryptosystems. In *Group theory, statistics, and cryptography*, volume 360 of *Contemp. Math.*, pages 167–177. Amer. Math. Soc., Providence, RI, 2004.

[27] V. Shpilrain and A. Ushakov. Thompson's group and public key cryptography. In *International Conference on Applied Cryptography and Network Security*, pages 151–163. Springer, 2005.

[28] C. C. Sims. *Computation with finitely presented groups*, volume 48 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1994.

[29] N. Smart. *Cryptography: An Introduction*. Mcgraw-hill education. McGraw-Hill, 2003.

[30] N. R. Wagner and M. R. Magyarik. A public-key cryptosystem based on the word problem. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology*, pages 19–36, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg.