A Class of PL-Homeomorphism Groups with Irrational Slopes

Jason Brown
Supervised by Dr Lawrence Reeves

Presented in partial fulfillment of the requirements of the degree
Master of Science

October 2018

School of Mathematics and Statistics
The University of Melbourne

Contents

1. Introduction

Groups of piece-wise linear and projective homeomorphisms of intervals in $\mathbb{R}$ have been sources of many rare examples and counter-examples in group theory. The prototype for these groups was Thompson's group F, and the related groups T and G. Thompson's group F can be described as:

1. The group of associative laws on binary operations [GG06].
2. The automorphism group of elements in the free monoidal category generated by a single object A and a morphism $A \otimes A \to A$ [FL10].
3. The group of piecewise linear homeomorphisms on the compact interval $[0, 1]$ with slopes that are powers of 2, and with finitely many breakpoints all lying in the dyadic rationals [CFP96].

The tendency for Thompson's groups to appear in such diverse settings has earned them the name *chameleon groups*. Definition 1 was historically the first [Bel07], given by Richard Thompson in 1965. Definition 3 has become the most widely used and, for the purposes of the generalisations we consider here, the most convenient. In this language, Thompson's group T is the group of homeomorphisms on $[0, 1]/0, 1 \cong S^1$ with slopes that are powers of 2, finitely many breakpoints all lying in the dyadic rationals and the additional requirement that dyadic rationals are mapped to dyadic rationals. G is the group of piece-wise linear right-continuous bijections on $[0, 1)$ with the same breakpoints and slopes. The group G was the first proven example of a finitely presented infinite simple group, and it was later shown that T shares this property. The group F is also thought to be a potential counter-example to the (now disproven) Von Neumann conjecture which states that a group is amenable if and only if it contains a non-abelian free group. While it has been shown that F does not contain a non-abelian free group, the amenability of F is still an open problem. It is, however, proven that F is not in the class EG of elementary amenable groups [CFP96]. As a finitely presented group not in EG which does not contain a non-abelian free group, it is either a counter-example to the Von Neumann conjecture for finitely presented groups (disproven in 2003 [OS03]) or the conjecture that finitely presented amenable groups are elementary amenable (disproven in 1998 [Gri98]).

The unusual properties of Thompson's groups have motivated the study of a number of generalisations. The groups $G_{n,r}$, sometimes called Higman-Thompson groups, are a generalisation of Thompson's group G introduced by Graham Higman in 1974 [Ste92]. They are the class of groups indexed by $n, r \in \mathbb{Z}$ of piecewise linear right-continuous bijections on the interval $[0, r)$ with slopes powers of n and breakpoints in $\mathbb{Z}[\frac{1}{n}]$. These groups were found to share with G the property of being a finitely presented infinite simple group for n even. Another generalisation studied, for example, by Bieri and Strebel [BS14], is groups of piecewise linear homeomorphisms on non-compact intervals of $\mathbb{R}$, for example $\mathbb{R}_{\geq 0}$ or $\mathbb{R}$.

A generalisation for which there have been very few published results is to allow the slopes to be powers of an irrational number τ, and require breakpoints in $\mathbb{Z}[\tau]$. One exception to this is the case where $\tau = \frac{-1+\sqrt{5}}{2}$, first discussed by Sean Cleary in [Cle00] where the group was proven to have the FP_∞ property, and thus to be finitely presented, using results from [Bro87]. A much more recent work by Burillo, Nucinkis and Reeves [BNR18] has adopted a more combinatorial approach to prove a number of results linking the group, referred to as F_τ, with Thompson's group F. For example, F_τ also has a simple commutator subgroup, is finitely presented, torsion free, and contains a copy of F.

The methods used in both [Cle00] and [BNR18] suggest a further generalisation to the case where $\tau = \frac{-k+\sqrt{k^2+4}}{2}$ for $k \in \mathbb{Z}_{\geq 1}$, which is the class of groups we consider here. We will refer to such a group corresponding to the integer k as $\mathcal{F}_{\tau_k}$. We will show that for each $k \in \mathbb{Z}_{\geq 1}$:

1. $\mathcal{F}_{\tau_k} \ncong \mathcal{F}_{\tau_h}$ for $h \neq k$
2. $\mathcal{F}_{\tau_k} \ncong F_h$ for $h \in \mathbb{Z}_{\geq 1}$
3. $\mathcal{F}_{\tau_k}$ is finitely presented
4. $\mathcal{F}_{\tau_k}$ contains a copy of F_k
5. $\mathcal{F}_{\tau_k}$ is not elementary amenable

Much of the important insights and proofs for these $\mathcal{F}_{\tau_k}$ groups come from studying combinatorial properties of a related class of trees and the correspondence between these trees and the groups of homeomorphisms. The structure of this document reflects this. We begin with a formal definition of a broad class of groups of piecewise linear homeomorphisms, called $F(l, A, P)$ groups, in Chapter 2. We then follow the path traced by the study of Thompson's group F and F_τ by looking in Chapter 3 at a class of subdivisions of the unit interval with the property that breakpoints of elements of $\mathcal{F}_{\tau_k}$ always lie in such a subdivision. We also introduce a notion of trees corresponding to such subdivisions, just as binary trees correspond to the binary subdivisions used to study the group F. The connection between pairs of such subdivisions is established in Chapter 4. We then establish a long list of results for the trees described in Chapter 3. This provides the tools to establish the infinite and finite presentations for $\mathcal{F}_{\tau_k}$ in Chapter 5. Chapter 6 then uses these presentations to derive a presentation for the abelianisation of $\mathcal{F}_{\tau_k}$ and a normal form closely mirroring the normal form given for F_τ in [BNR18]. Finally, using some known results for the group F_1 and general $F(l, A, P)$ groups we show that each $\mathcal{F}_{\tau_k}$ contains F, is not elementary amenable and does not contain a free non-abelian subgroup. This demonstrates a connection between the amenability of F and $\mathcal{F}_{\tau_k}$.

2. $F(l, A, P)$ Groups

The class of $F(l, A, P)$ groups is a very broad generalisation of Thompson's group F where the slope groups, breakpoint sets and the interval are allowed to vary.

Definition 1 ($F(l, A, P)$)**:** We construct a group of piecewise linear homeomorphism on a compact interval by choosing:

1. P, a multiplicative subgroup of $\mathbb{R}_{>0}$
2. A, a $\mathbb{Z}[P]$-submodule of $\mathbb{R}$ (where P acts on $\mathbb{R}$ by multiplication) satisfying $P \cdot A = A$
3. An element $l \in A \cap \mathbb{R}_{>0}$

$F(l, A, P)$ is then the group of piecewise linear homeomorphisms $f : [0, l] \to [0, l]$ with finitely many breakpoints, all in A, and with the slopes of all the linear segments in P. The group operation is composition. We will adopt the convention that multiplication on the left is precomposition, so $fg = g \circ f$.

We could similarly define the analogous generalisations $T(l, A, P)$ and $G(l, A, P)$ of T and G respectively, though they are not discussed in this paper.

Thompson's group F can be expressed as the $F(l, A, P)$ group $F\big([0, 1], \mathbb{Z}[\frac{1}{2}], \langle 2\rangle\big)$, where $\langle 2\rangle$ denotes the multiplicative subgroup of $\mathbb{R}_{>0}$ generated by 2. A related class of groups are the groups $F\big([0, 1], \mathbb{Z}[\frac{1}{k}], \langle k\rangle\big)$, for $k \in \mathbb{Z}_{>1}$ which we will henceforth denote by F_{k-1}. Our reason for not using F_k to denote $F\big([0, 1], \mathbb{Z}[\frac{1}{k}], \langle k\rangle\big)$ is to highlight the correspondence we will demonstrate between the groups $\mathcal{F}_{\tau_k}$ and F_k. Note that with this definition, Thompson's group is F_1.

A comprehensive reference for this general class of groups is [BS14]. We will state two results for general $F(l, A, P)$ groups which are both found in [BS14].

Let $B = F(l, A, P)$ for some choice of l, A and P.

Proposition 1: *B is torsion free*

Proof. Assume $f \in B$ is not the identity. Let $D_x(f)$ denote the derivative at $x \in [0, 1)$ from the right (i.e. $\lim_{h\to 0^+}\Big(\frac{f(a+h)-f(x)}{h}\Big)$) and let a be the infimum of the set $\{x \in [0, 1] \mid f(x) \neq x\}$. Then $f(a) = a$, but $D_a(f) \neq 1$. Then $D_a(f^n) = (D_a f)^n \neq 1$ for any $n \in \mathbb{Z}$, so f is not a torsion element. We conclude that B is torsion free. $\square$

In particular, B is either trivial or infinite.

The next theorem we state paraphrases an important result proven by Brin and Squier in [BS85].

Theorem (3.1 from [BS85])**:** *B does not contain a non-abelian free group.*

Every $F(l, A, P)$ group is therefore either amenable, or a counter-example to the Von Neumann conjecture.

We will investigate a class of $F(l, A, P)$ groups related to a class of algebraic integers we now define.

Definition 2 (τ_k)**:** For $k \in \mathbb{Z}_{>0}$ we let τ_k denote the positive solution to the quadratic equation: $x^2 + kx = 1$ which is given by:

$$\tau_k := -\frac{k}{2} + \sqrt{\left(\frac{k}{2}\right)^2 + 1}$$

These algebraic integers are sometimes referred to as *metallic means*. We note that $\tau_k \in (0, 1)$ for all $k \in \mathbb{Z}_{>0}$.

Definition 3 ($\mathcal{F}_{\tau_k}$)**:** For $k \in \mathbb{Z}_{\geq 1}$ we define $\mathcal{F}_{\tau_k}$ to be $F(1, \mathbb{Z}[\tau_k], \langle \tau_k \rangle)$

With this notation, the group discussed in [Cle00] and [BNR18] with slopes powers of $\frac{-1+\sqrt{5}}{2}$ is $\mathcal{F}_{\tau_1}$.

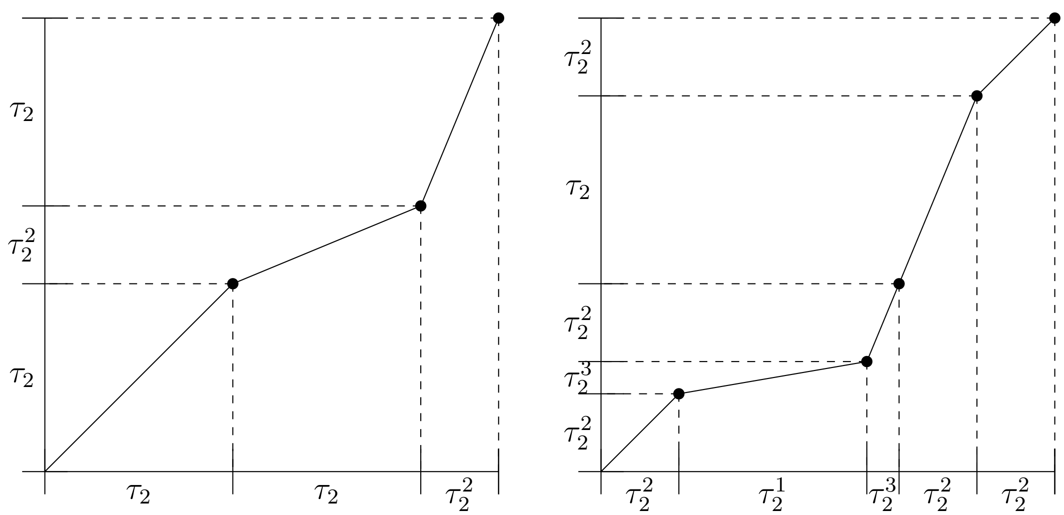


Figure 2.1: Two elements of $\mathcal{F}_{\tau_2}$

3. Sequences, Subdivisions and Trees

3.1 k-sequences and k-subdivisions

Assume $k \in \mathbb{Z}_{\geq 1}$.

Definition 4 (Simple k-sequence, ρ_n)**:** A simple k-sequence, ρ_n is a list $\rho_n : [k] \to \mathbb{Z}_{\geq 0}$ with:

$$\rho_n(i) = \begin{cases} 1 & i \neq n \\ 2 & i = n \end{cases}$$

where $[k]$ denotes the set $\{0, 1, \ldots, k\}$.

Example 1 (Simple 2-sequences)**:** For $k = 2$, there are exactly 3 simple k-sequences:

$$\rho_0 = 2, 1, 1 \quad \rho_1 = 1, 2, 1 \quad \rho_2 = 1, 1, 2$$

Definition 5 (k-sequence)**:** An element of the set kSeq of k-sequences is either:

1. 0
2. A sequence obtained from a k-sequence by replacing a term l with $(\rho_i)^{+l}$ for some simple k-sequence ρ_i (henceforth a "type i substitution of l"), where $(w)^{+s}$ denotes the sequence obtained by incrementing all terms in w by s.

For $w, w' \in k$Seq, we say that w' is a refinement of w if it is obtained from w by a sequence of substitutions.

Example 2 (k-sequence in 2Seq)**:** We can obtain $2, 3, 2, 1, 2 \in 2$Seq in the following way:

$$0 \longrightarrow (\rho_2)^{+0} = 1, 1, 2 \longrightarrow (\rho_1)^{+1}, 1, 2 \longrightarrow (1, 2, 1)^{+1}, 1, 2 \longrightarrow 2, 3, 2, 1, 2$$

Definition 6 (Generalised k-sequence)**:** An element of the set Gen$k$Seq of *generalised k-sequences* is a sequence $l_0, \ldots, l_n$ of non-negative integers such that $\sum_{i=0}^{n} \tau_k^{l_i} = 1$. A refinement of $w = l_0, \ldots, l_n$ is a sequence obtained by replacing some l_j in w with $(\rho_n)^{+l_j}$.

We will prove shortly that k-sequences and refinements of generalised k-sequences are indeed generalised k-sequences.

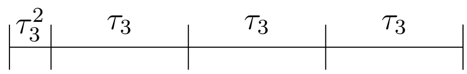
Definition 7 (Simple k-subdivision, κ_n)**:** A simple k-subdivision, κ_n, is a partition of $[0, 1]$ with breakpoints:

$$0, \tau_k, 2\tau_k, \ldots, (i-1)\tau_k, i\tau_k, i\tau_k + \tau_k^2, (i+1)\tau_k + \tau_k^2, \ldots, (k-1)\tau_k + \tau_k^2, 1$$

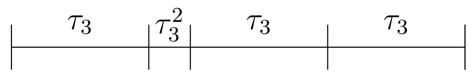
or equivalently:

$$b_0 = 0, b_1 = b_0 + \tau_k^{\rho_n(0)}, \ldots, b_{k+1} = b_k + \tau_k^{\rho_n(k)}$$

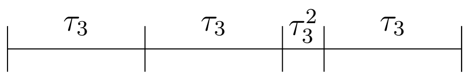
An example is more illuminating than the definition given above. Figure 3.1 shows all the possible simple 3-subdivisions of the unit interval.



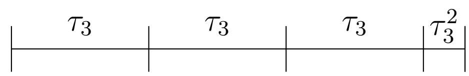
(a) Type 0



(b) Type 1



(c) Type 2



(d) Type 3

Figure 3.1: The four simple 3-subdivisions of the unit interval

Definition 8 (k-subdivision)**:** An element of the set kSub of k-subdivisions is either:

1. $[0, 1]$
2. A subdivision w' obtained from a subdivision $w \in k\text{Sub}$ by replacing an interval $[a, b]$ with the image of a simple k-subdivision κ_n under the map $\phi_{a,b} : [0, 1] \to [a, b]$, $\phi(x) = a + (b - a)x$ (henceforth a "type n partition of $[a, b]$")

We will say the *length* of a k-subdivision is the number of its intervals.

For example, (*) shows a 2-subdivision obtained by a sequence of two 2-partitions starting from the unit interval.

$$
\begin{aligned}
&[0, 1] \\
\longrightarrow &[0, \tau_k], [\tau_k, 2\tau_k], [2\tau_k, 1] \\
\longrightarrow &[0, \tau_k^2], [\tau_k^2, \tau_k(\tau_k + \tau_k^2)], [\tau_k(\tau_k + \tau_k^2), \tau_k], [\tau_k, 2\tau_k], [2\tau_k, 1]
\end{aligned}
\tag{*}
$$

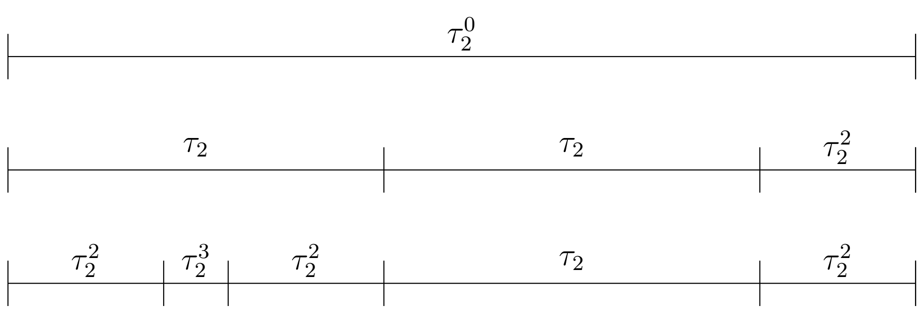


Figure 3.2: The 2-subdivision (*) obtained by two 2 partitions

Definition 9 (Gen$k$Sub)**:** An element of the set Gen$k$Sub of *generalised k-subdivisions* is a partition of the unit interval into finitely many intervals, each with length a power of τ_k. A refinement of a generalised k-subdivision, P, is any generalised k-subdivision obtained from P by a sequence of k-partitions (of any type).

Note that all k-subdivisions are also generalised k-subdivisions, however there exist generalised k-subdivisions which are not k-subdivisions, for example the generalised 1-subdivision given by:

$$P = \{ [\, 0 \,,\, \tau_1^4 \,], [\, \tau_1^4 \,,\, \tau_1^4 + \tau_1 \,], [\, \tau_1^4 + \tau_1 \,,\, 1 \,] \} \tag{$\dagger$}$$

We note that $(1 - (\tau_1^4 + \tau_1)) = \tau_1^3$. This couldn't be a 1-subdivision because the only way to obtain an interval of length τ_1 in a k-subdivision is by performing a 1-partition on the unit interval, and when we do this the τ_1 interval contains either 0 or 1. The generalised k-subdivision P contains an interval of length τ_k which contains neither 1 nor 0, so P cannot be a k-subdivision.

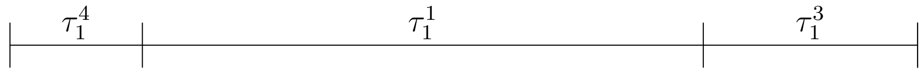


Figure 3.3: The generalised 1-subdivision ($\dagger$) which is not a 1-subdivision

Definition 10 (GentoSeq)**:** We define a map GentoSeq : Gen$k$Sub $\to$ Gen$k$Seq by sending a subdivision r to the sequence whose j^{th} term is $\log_{\tau_k}(b - a)$, where $[a, b]$ is the j^{th} interval of r.

Lemma 1: GentoSeq *is a bijection which preserves refinements and lengths*

Proof. It is clear from the definition that GentoSeq is a length preserving bijection. It remains to show that it preserves refinements.

Assume that r' is a k-subdivision obtained from another k-subdivision r by a n-partition of the j^{th} interval in r, and let $w = \text{GentoSeq}(r)$.

If the j^{th} term of w is l, then the j^{th} interval, I of r is of width τ_k^l. Performing a n-partition on I replaces it with $k+1$ intervals, whose lengths from left to right are $\rho_n(i) + l$. It follows that toSeq(r') is the k-sequence obtained from w by replacing l with $(\rho_n)^{+l}$, thus a refinement of w. $\square$

Corollary 1: *Refinements of generalised k-sequences are generalised k-sequences. In particular, k-sequences are generalised k-sequences (they are refinements of* $0 \in$ Gen$k$Seq*).*

Definition 11 (GentoSub)**:** We note that GentoSeq^{-1} is the map GentoSub : kSeq $\to$ kSub by letting toSub$(l_0, \ldots, l_n)$ be the k-subdivision defined by the break points $(b_0 = 0, b_1, \ldots, b_{n+1})$ where $b_{i+1} = b_i + \tau_k^{l_i}$.

Definition 12 (toSeq, toSub)**:** Note that GentoSeq$|_{k\text{Sub}}$ is a bijection from kSub to kSeq. This is because GentoSeq maps $[0, 1]$ to 0, and all k-subdivisions are refinements of $[0, 1]$ and all k-sequences are refinements of 0. We let toSub $=$ GentoSub$|_{k\text{Seq}}$.

Definition 13 (Uniform k-sequence)**:** A *uniform k-sequence of level l* is a generalised k-sequence with all terms l or $l + 1$.

An example of a uniform 2-sequence of level 3 is given by the leaf sequence $(3, 3, 3, 3, 3, 4, 4)$. We note that this is not a 2-sequence, whereas the uniform 2-sequence given by $(3, 3, 4, 3, 3, 4, 3)$ is.

Lemma 2: *Every generalised k-sequence has a uniform refinement. Moreover, if l is the maximum term in w, then w has a uniform refinement of level $l-1$*

Proof. Let w be a generalised k-sequence with maximum term l and minimum term h. We prove the result by induction on $l-h$.

If $l-h \leq 1$ then the generalised k-sequence is already uniform of level $l-1$. Now assume $l-h > 1$. We can perform a substitution (of any type) of each h term, replacing them with $h+1$ and $h+2$ terms. We will call this refinement w'. Note that $h+2 \leq l$, since by hypothesis $l-h > 1$. The minimum value in the leaf sequence of w' is now $h+1$ but the maximum value is still l. So w' has a uniform refinement by the induction hypothesis, and thus so does w. □

Lemma 3: *If w is a uniform k-sequence of level l and $h > l$ then w has a refinement which is a uniform k-subdivision of level h*

Proof. Assume w is a uniform k-sequence of level l. It suffices to show that w has a refinement that is a uniform k-sequence of level $l+1$.

All terms in w are either l or $l+1$. If there are no l terms then w is already a uniform k-sequence of level $l+1$. Otherwise, performing a substitution of one of the l terms gives a refinement w' of w with maximum term $l+2$. By Lemma 2 there exists a refinement of w' which is uniform at level $l+1$, which is therefore also a refinement of w. □

Lemma 4: *A uniform k-sequence at a given level is unique up to permutation.*

Proof. It suffices to show that for any given $l \in \mathbb{Z}_{\geq 0}$ there exists unique $n, m \in \mathbb{Z}$ such that $1 = n\tau_k^l + m\tau_k^{l+1}$. We prove this by contradiction. Assume there exists a distinct pair $n', m' \in \mathbb{Z}$ such that $n\tau_k^l + m\tau_k^{l+1} = n'\tau_k^l + m'\tau_k^{l+1}$, from which it follows that:

$$\tau_k = \frac{n-n'}{m'-m}$$

But τ_k is irrational for all $k > 0$, so we have a contradiction. □

3.2 k-trees

A natural way to represent a subdivision is to construct a tree, where nodes of the tree represent intervals of the subdivision and branching in the tree corresponds to a partition of the interval in the subdivision. Studying such tree representations of subdivisions has proven fruitful in the study of Thompson's groups [CFP96; Bur18] and the group $\mathcal{F}_{\tau_1}$ [BNR18]. With some small modifications, we can use trees in a similar way to study the general group $\mathcal{F}_{\tau_k}$.

Definition 14 (k-tree)**:** We define an element of the set of k-trees, kTree, recursively as either:

1. A leaf: $\varnothing$

2. A node $(n, f) \in [k] \times k\text{Tree}^{[k]}$

where $k\text{Tree}^{[k]}$ denotes the set of functions from $[k]$ to kTree. We say the *type* of a tree $T = (n, f)$ is n. $\varnothing$ has no type.

Example 3 (Simple tree, Λ_n)**:** After $\varnothing$, the most simple k-trees are those with a single non-leaf:

$$\Lambda_n := (n, \text{nc}) \quad \text{where} \quad \text{nc}(i) = \varnothing \; \forall i \in [k]$$

Definition 15 (Subtree, height, order, refinement)**:** Given a k-tree, T, a *subtree* of T as either:

1. T
2. $f(i)$ for some $i \in [k]$ where $T' = (n, f)$ is a subtree of T

Given a subtree T' of T, we define its *height* to be:

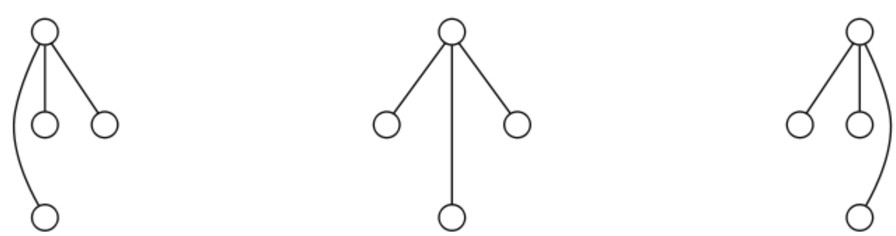
1. 0 if $T' = T$
2. $h + \rho_n(i)$ if $T' = f(i)$, for some subtree $T'' = (n, f)$ of height h

We say a subtree T' of T is *left of* a subtree T'' if there exists some other subtree $\hat{T} = (n, f)$ and $i < j \in [k]$ such that T' is a subtree of $f(i)$ and T'' is a subtree of $f(j)$.

For $T, T' \in k$Tree, T' is a refinement of T if either

1. $T = \varnothing$
2. $T = (n, f)$ and $T' = (n, f')$ with $f'(i)$ a refinement of $f(i)$ for all $i \in [k]$

We can gain some intuition about k-trees by constructing graphical representations called *tree diagrams*. We will represent the k-tree $\varnothing$ with a single node, ○, and we will represent a k-tree (n, f) by drawing a node, the root, with a line attached to each subtree $f(i)$ in left to right order. The line attaching the root to $f(n)$ will be twice as long. This ensures that the height of a given subtree corresponds to the distance from the root node in the tree diagram. Figure 3.4 shows the tree diagrams for each simple 2-tree.



(a) $\Lambda_0 : (2, 1, 1)$ (b) $\Lambda_1 : (1, 2, 1)$ (c) $\Lambda_2 : (1, 1, 2)$

Figure 3.4: The tree diagrams of the three simple 2-trees

We will introduce another graphical representation of k-trees called *compact tree diagrams* which are both more compact, and can be used to more easily represent a general class of k-trees, rather than specific k-trees. This representation is constructed by letting the compact

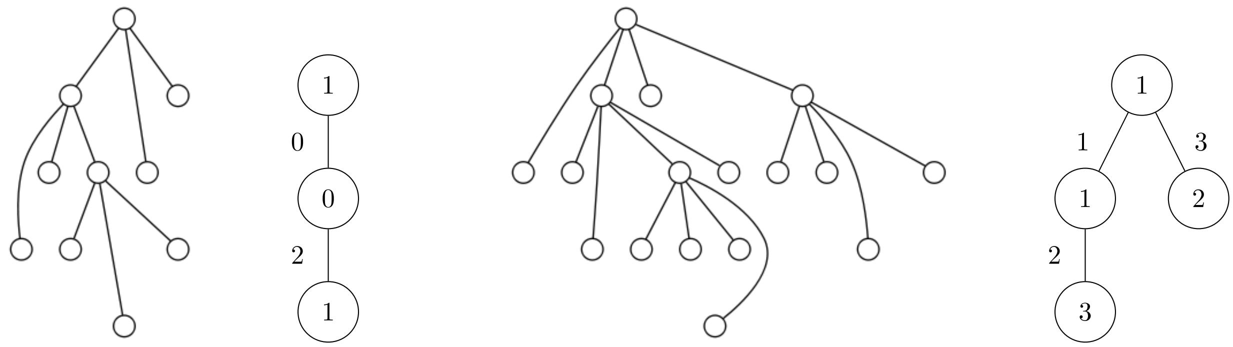


Figure 3.5: Two examples of k-trees, with $k = 2$ and 3, comparing the compact and standard notations

tree diagram for $\varnothing$ be empty, and letting the compact tree diagram for a k-tree (n, f) be a node with label n attached to each of its non-leaf children $f(i)$ by edges labeled i. Figure 3.5 shows both the tree diagram and compact tree digram for various k-trees.

Definition 16 (Leaf sequence)**:** From a k-tree T we construct a list $\mathrm{LSeq}(T)$, called the *leaf sequence* of T as follows:

1. $\mathrm{LSeq}(\varnothing) = 0$
2. $\mathrm{LSeq}(n, f) = \bigoplus_{i=0}^{k}(\mathrm{LSeq}(f(i)))^{+\rho_n(j)}$

where $\oplus$ indicates concatenation of lists.

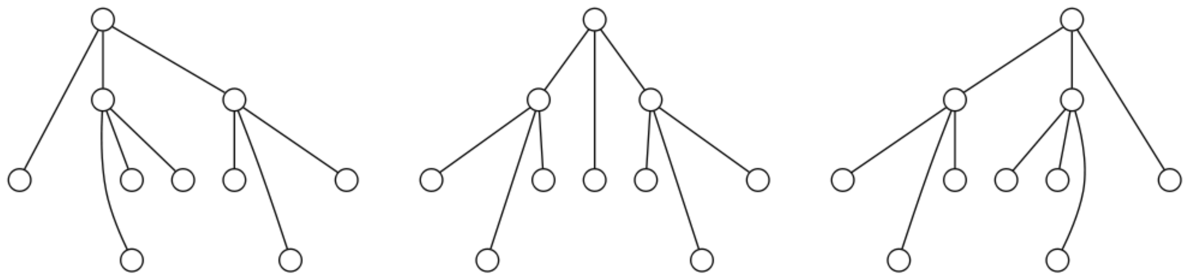
Remark. Note that the leaf sequence of a k-tree can equivalently be expressed as the heights of its subtrees which are leaves in left-to-right order. Occasionally this definition will prove more useful.

Example 4 (The leaf sequence of simple k-trees)**:** Note that $\mathrm{LSeq}(\Lambda_n) = \rho_n$:

$$\begin{aligned}
\mathrm{LSeq}(\Lambda_n) &= \bigoplus_{i=0}^{n-1}(\mathrm{LSeq}(\mathrm{nc}(i)))^{+1} \oplus (\mathrm{LSeq}(\mathrm{nc}(n)))^{+2} \oplus \bigoplus_{i=n+1}^{k}(\mathrm{LSeq}(\mathrm{nc}(i)))^{+1} \\
&= \bigoplus_{i=0}^{n-1}(\mathrm{LSeq}(\varnothing))^{+1} \oplus (\mathrm{LSeq}(\varnothing))^{+2} \oplus \bigoplus_{i=n+1}^{k}(\mathrm{LSeq}(\varnothing))^{+1} \\
&= \bigoplus_{i=0}^{n-1}(0)^{+1} \oplus (\mathrm{LSeq}(0))^{+2} \oplus \bigoplus_{i=n+1}^{k}(0)^{+1} \\
&= \underbrace{1, \ldots, 1}_{(n-1)\ 1\text{'s}}, 2, \underbrace{1, \ldots, 1}_{(k-n)\ 1\text{'s}} \\
&= \rho_n
\end{aligned}$$

This computation may be too much detail

This can be seen visually in the case $k = 2$ by considering the heights of the leaf nodes of each of the simple 2-trees shown in Figure 3.4.



(a) T_1 (b) T_2 (c) T_3

Figure 3.6: Three distinct 2-trees with the same leaf sequence $(2, 3, 2, 2, 2, 3, 2)$

Lemma 5: $\mathrm{LSeq} : k\mathrm{Tree} \to k\mathrm{Seq}$ *is a well-defined surjection*

Proof. We first show that the leaf sequence of any k-tree is in fact a k-sequence.

Note that $\mathrm{LSeq}(\varnothing) = 0 \in k\mathrm{Seq}$. If $T = (n, f)$ then let $w = \mathrm{LSeq}(T) = \bigoplus_{i=0}^{k} (\mathrm{LSeq}(f(i)))^{+\rho_n(i)}$. By induction we may assume that $w_i = \mathrm{LSeq}(f(i)) \in k\mathrm{Seq}$ for each i. From the definition of k-sequences we can see that for any two k-sequences $u, v \in k\mathrm{Seq}$, replacing a term l in u with $(v)^{+l}$ gives another k-sequence, since v itself is obtained by a sequence of k-substitutions. *is this clear enough?* Our result follows by noting that w is obtained from ρ_n by replacing each term $\rho_n(i)$ with $(w_i)^{+\rho_n(i)}$.

We now prove surjectivity. Clearly $0 \in k\mathrm{Seq}$ is $\mathrm{LSeq}(\varnothing)$, so is in the image of LSeq. Assume now that some $w \in k\mathrm{Seq}$ is equal to $\mathrm{LSeq}(T)$ and that w' is obtained from w by replacing some l in w by $(\rho_j)^{+l}$.

By the fact that $w = \mathrm{LSeq}(T)$, there exists a leaf subtree of T (hence forth "leaf of T") of height h. Since T is not a leaf, there must exist some other subtree $T' = (n, f)$ of T and some $i \in [k]$ such that this leaf is $f(i)$. If we let $f(i) = \Lambda_j$, then we get a new k-tree, $\hat{T}$, whose leaf sequence is obtained by replacing $l \in w$ with $(\mathrm{LSeq}(\Lambda_j))^{+l}$.

By Example 4 we have that $\mathrm{LSeq}(\Lambda_j) = \rho_j$ and so $\mathrm{LSeq}\left(\hat{T}\right) = w'$. $\square$

We can show that LSeq is not in general injective by constructing different k-trees with the same leaf sequence. Figure 3.6 shows 3 such 2-trees.

Definition 17 (Leaf equivalence)**:** We say $T_1, T_2 \in k\mathrm{Tree}$ are *leaf equivalent*, written $T_1 \sim T_2$, if $\mathrm{LSeq}(T_1) = \mathrm{LSeq}(T_2)$.

Lemma 6: *If* T' *is a refinement of* $T \in k\mathrm{Tree}$ *then* $\mathrm{LSeq}(T')$ *is a refinement of* $\mathrm{LSeq}(T)$.

Proof. If $T = \varnothing$, then $\mathrm{LSeq}(T')$ is a refinement of $\mathrm{LSeq}(T) = 0$, since every k-sequence is a refinement of 0.

If $T = (n, f)$, then $T' = (n, f')$ where $f'(i)$ is a refinement of $f(i)$ for all $i \in [k]$. Then:

$$\mathrm{LSeq}(T) = \bigoplus_{i=0}^{k} (\mathrm{LSeq}(f(i)))^{+\rho_n(i)} \qquad \mathrm{LSeq}(T') = \bigoplus_{i=0}^{k} (\mathrm{LSeq}(f'(i)))^{+\rho_n(i)}$$

By the induction hypothesis, each $\mathrm{LSeq}(f'(i))$ can be obtained from $\mathrm{LSeq}(f(i))$ by substitutions of simple k-sequences. It follows that $\mathrm{LSeq}(T')$ can be obtained from $\mathrm{LSeq}(T)$ by such substitutions, and thus is a refinement of $\mathrm{LSeq}(T)$. □

Corollary 2: $\mathrm{toSub} \circ \mathrm{LSeq} : k\mathrm{Tree} \to k\mathrm{Sub}$ *is a refinement-preserving surjection.*

We will use the phrase "T represents r" to mean $(\mathrm{toSub} \circ \mathrm{LSeq})(T) = r$.

Definition 18 ($\mathrm{Gen}k\mathrm{Tree}$)**:** An element of the set $\mathrm{Gen}k\mathrm{Tree}$ of *generalised k-trees* is either:

1. A leaf: $\varnothing$
2. A node $(w, f) \in \bigsqcup_{s \in \mathbb{Z}_{\geq 0}} \mathrm{Gen}k\mathrm{Seq}_s \times k\mathrm{Tree}^{[s]}$

where $\mathrm{Gen}k\mathrm{Seq}_s$ is the set of generalised k-sequences of length $s + 1$.

Definition 19 ($\mathrm{GenLSeq}$)**:** We define for each generalised k-tree a *generalised leaf sequence* with the map $\mathrm{GenLSeq} : \mathrm{Gen}k\mathrm{Tree} \to \mathrm{Gen}k\mathrm{Seq}$ defined by:

1. $\mathrm{GenLSeq}(\varnothing) = 0$
2. $\mathrm{GenLSeq}(w, f) = \bigoplus_{i=0}^{s} (\mathrm{LSeq}(f(i)))^{+w(i)}$

Note that we can think of $k\mathrm{Tree}$ as a subset of $\mathrm{Gen}k\mathrm{Tree}$ by the map:

$$\iota : k\mathrm{Tree} \to \mathrm{Gen}k\mathrm{Tree} \qquad \text{by} \qquad \varnothing \mapsto \varnothing \qquad (n, f) \mapsto (\rho_n, f)$$

which satisfies $\mathrm{GenLSeq} \circ \iota = \mathrm{LSeq}$.

Lemma 7: $\mathrm{GenLSeq}$ *is well-defined, surjective and preserves refinements*

Proof. We can see that if $T = (w, \mathrm{nc}) \in \mathrm{Gen}k\mathrm{Tree}$ then $\mathrm{GenLSeq}(T) = w$, and thus $\mathrm{GenLSeq}$ is surjective.

That $\mathrm{GenLSeq}$ preserves refinements follows from the fact that $\mathrm{LSeq} : k\mathrm{Tree} \to k\mathrm{Seq}$ preserves refinements. We conclude that $\mathrm{GenLSeq}$ is well-defined by the fact that its image is therefore the set of refinements of generalised k-sequences, which by Corollary 1 is $\mathrm{Gen}k\mathrm{Seq}$. □

3.2.1 Graft Operations

Definition 20 ($\mathrm{GenLGT}_n, \mathrm{GenRGT}_n$)**:** We define subsets $\mathrm{GenLGT}_n, \mathrm{GenRGT}_n \subseteq \mathrm{Gen}k\mathrm{Tree}$ as:

1. $\mathrm{GenRGT}_n = \{(w, f) \in \mathrm{Gen}k\mathrm{Tree} \mid w(n+1) = w(n) - 1,\ f(n+1) = (m, g),\ m \neq k\}$
2. $\mathrm{GenLGT}_n = \{(w, f) \in \mathrm{Gen}k\mathrm{Tree} \mid w(n-1) = w(n) - 1,\ f(n-1) = (m, g),\ m \neq 0\}$

Definition 21 (GenRGraft$_n$, GenLGraft$_n$)**:** We define a family of functions $\text{GenRGraft}_n : \text{GenRGT}_n \to \text{GenLGT}_{n+1}$ as follows. Assume $T = (w, f) \in \text{GenRGT}_n$, let $(m, g) = f(n+1)$. Note that $m \neq k$ and $w(i+1)$ is well-defined. Then we define $\text{GenRGT}_n(T) = (w', f')$ where:

$$(w'(i), f'(i)) = \begin{cases} (w(i), f(i)) & i \neq n, n+1 \\ (w(n+1), (m+1, g')) & i = n \\ (w(n), g(k)) & i = n+1 \end{cases} \quad \text{where} \quad g'(i) = \begin{cases} f(n) & i = 0 \\ g(i-1) & i \neq 0 \end{cases}$$

Note that $w'(n) = w(n+1) = w(n) - 1 = w'(n+1) - 1$, and $f'(n) = (m+1, g')$, with $m + 1 \neq 0$, so $(w', f') \in \text{GenLGT}_{n+1}$.

Similarly, we define functions $\text{GenLGraft}_n : \text{GenLGT}_n \to \text{GenRGT}_{n-1}$ by mapping $T = (w, f) \in \text{GenLGT}_n$, $(m, g) = f(n-1)$, to $\text{GenRGT}_n(T) = (w', f')$ where:

$$(w'(i), f'(i)) = \begin{cases} (w(i), f(i)) & i \neq n, n-1 \\ (w(n-1), (m-1, g')) & i = n \\ (w(n), g(0)) & i = n-1 \end{cases} \quad \text{where} \quad g'(i) = \begin{cases} f(n) & i = k \\ g(i+1) & i \neq k \end{cases}$$

Note that $w'(n) = w(n-1) = w(n) - 1 = w'(n-1) - 1$, and $f'(n) = (m-1, g')$, with $m - 1 \neq k$, so $(w', f') \in \text{GenRGT}_{n-1}$.

Lemma 8:

$$\text{GenLGraft}_{n+1} \circ \text{GenRGraft}_n = \text{id}_{\text{GenRGT}_n} \quad \text{and} \quad \text{GenRGraft}_{n-1} \circ \text{GenLGraft}_n = \text{id}_{\text{GenLGT}_n}$$

Proof. *By inspection? It's a straight-forward computation* □

Lemma 9: GenLGraft$_n$ *and* GenRGraft$_n$ *preserve leaf sequences.*

Proof. It suffices to show that GenRGraft$_n$ preserves leaf sequences by the fact that:

$$\text{GenLGraft}_n = (\text{GenRGraft}_{n-1})^{-1}$$

Assume $T = (w, f) \in \text{GenRGT}_n$ with $(m, g) = f(n+1)$. Then GenLSeq(T) is given by:

$$\bigoplus_{i=0}^{n-1} (\text{LSeq}(f(i)))^{+w(i)} \oplus (\text{LSeq}(f(n)))^{+w(n)} \oplus (\text{LSeq}(f(n+1)))^{+w(n+1)} \oplus \bigoplus_{i=n+2}^{s} (\text{LSeq}(f(i)))^{+w(i)}$$

Letting $T' = (n+1, f') = \text{GenRGraft}_n(T)$, GenLSeq$(T')$ is:

$$\bigoplus_{i=0}^{n-1} (\text{LSeq}(f'(i)))^{+w'(i)} \oplus (\text{LSeq}(f'(n)))^{+w'(n)} \oplus (\text{LSeq}(f'(n+1)))^{+w'(n+1)} \oplus \bigoplus_{i=n+2}^{s} (\text{LSeq}(f'(i)))^{+w'(i)}$$
$$\bigoplus_{i=0}^{n-1} (\text{LSeq}(f(i)))^{+w(i)} \oplus (\text{LSeq}(f'(n)))^{+w(n+1)} \oplus (\text{LSeq}(f'(n+1)))^{+w(n)} \oplus \bigoplus_{i=n+2}^{s} (\text{LSeq}(f(i)))^{+w(i)}$$

since $f'(i) = f(i)$ and $w(i) = w'(i)$ for $i \neq n, n+1$.

We can see that $\text{GenLSeq}(T) = \text{GenLSeq}(T')$ if and only if:

$$\text{LSeq}(f'(n)) \oplus (\text{LSeq}(f'(n+1)))^{+1} = (\text{LSeq}(f(n)))^{+1} \oplus \text{LSeq}(f(n+1))$$

using the fact that $w(n) = w(n+1) + 1$.

This follows from the definitions of f' and g':

$$\begin{aligned}
&\text{LSeq}(f'(n)) \oplus (\text{LSeq}(f'(n+1)))^{+1} \\
=&\,\text{LSeq}(m+1, g') \oplus (\text{LSeq}(g(k)))^{+1} \\
=&\,(\text{LSeq}(g'(0)))^{+1} \oplus \bigoplus_{i=1}^{m}(\text{LSeq}(g'(i)))^{+1} \oplus (\text{LSeq}(h(m+1)))^{+2} \oplus \bigoplus_{i=m+2}^{k}(\text{LSeq}(g'(i)))^{+1} \\
&\quad \oplus (\text{LSeq}(g(k)))^{+1} \\
=&\,(\text{LSeq}(f(n)))^{+1} \oplus \left(\bigoplus_{i=0}^{m-1}(\text{LSeq}(g(i-1)))^{+1} \oplus (\text{LSeq}(g(m)))^{+2} \oplus \bigoplus_{i=m+1}^{k}(\text{LSeq}(g(i)))^{+2}\right) \\
=&\,(\text{LSeq}(f(n)))^{+1} \oplus \text{LSeq}(m, g) \\
=&\,(\text{LSeq}(f(n)))^{+1} \oplus \text{LSeq}(f(n+1))
\end{aligned}$$

□

Definition 22 (LGT, RGT)**:** We define subsets $\text{LGT}, \text{RGT} \subseteq k\text{Tree}$ as:

1. $\text{RGT} = \{(n, f) \in k\text{Tree} \,|\, n \neq k,\ f(n+1) = (m, g),\ m \neq k\}$
2. $\text{LGT} = \{(n, f) \in k\text{Tree} \,|\, n \neq 0,\ f(n-1) = (m, g),\ m \neq 0\}$

Lemma 10: $\text{RGT} = k\text{Tree} \cap \left(\bigsqcup_{n=0}^{k-1} \text{GenRGT}_n\right)$, $\text{LGT} = k\text{Tree} \cap \left(\bigsqcup_{n=1}^{k} \text{GenLGT}_n\right)$

Proof. This follows from the fact that $\rho_j(n+1) = \rho_j(n) - 1$ and $\rho_j(n-1) = \rho_j(n) - 1$ only hold for $j = n$. □

Definition 23 (RGraft, LGraft)**:** We define functions $\text{RGraft} : \text{RGT} \to \text{LGT}$ and $\text{LGraft} : \text{LGT} \to \text{RGT}$ by:

$$\text{RGraft}(n, f) = \text{GenRGraft}_n(n, f) = (n+1, f')$$

$$\text{LGraft}(n, f) = \text{GenRGraft}_n(n, f) = (n-1, f')$$

Lemma 11: $\text{LGraft} \circ \text{RGraft} = \text{id}_{\text{RGT}}$ *and* $\text{RGraft} \circ \text{LGraft} = \text{id}_{\text{LGT}}$

Proof. This follows from Lemma 8. □

Lemma 12: LGraft *and* RGraft *preserve leaf sequences.*

Proof. This follows from Lemma 9. □

Our tree diagrams provide some intuition for this result, as the two graft operations visually resemble detaching and reattaching edges without moving any of the nodes. For example, Figure 3.7 demonstrates how T_2 from Figure 3.6 is obtained from T_1 by the function RGraft.

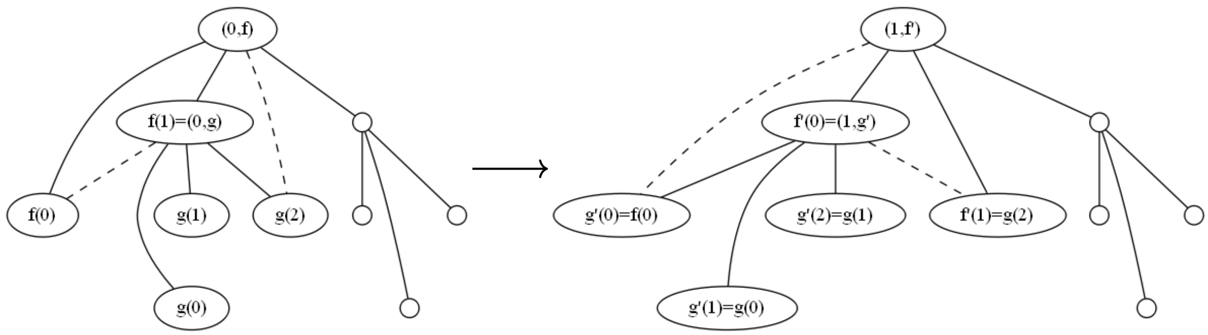


Figure 3.7: Using RGraft to transform T_1 from Figure 3.6 into T_2

Definition 24 (LT_k, RT_k)**:**

$$\mathrm{RT}_k := \{(n, f) \in k\mathrm{Tree} \mid n \neq k\} \qquad \mathrm{LT}_k := \{(n, f) \in k\mathrm{Tree} \mid n \neq 0\}$$

Definition 25 (A_R, A_L)**:** We recursively define two functions $\mathrm{A}_R : k\mathrm{Tree} \to \mathrm{RT}_k$ and $\mathrm{A}_L : k\mathrm{Tree} \to \mathrm{LT}_k$ as:

1. $\mathrm{A}_R(\varnothing) = \Lambda_0$
2. $\mathrm{A}_R(n, f) = (n, f) \quad n \neq k$
3. $\mathrm{A}_R(k, f) = \mathrm{LGraft}(k, f')$ where:

$$f'(i) = \begin{cases} f(i) & i \neq k-1 \\ \mathrm{A}_L(f(i)) & i = k-1 \end{cases}$$

1. $\mathrm{A}_L(\varnothing) = \Lambda_1$
2. $\mathrm{A}_L(n, f) = (n, f) \quad n \neq 0$
3. $\mathrm{A}_L(0, f) = \mathrm{RGraft}(0, f')$ where:

$$f'(i) = \begin{cases} f(i) & i \neq 1 \\ \mathrm{A}_R(f(i)) & i = 1 \end{cases}$$

To see that these functions are well-defined, we use induction. Clearly the functions are well-defined on $\varnothing$, and in the case where $n \neq k$ for A_R and $n \neq 0$ for A_L.

Note that if $T = (k, f)$, then by the induction hypothesis, $f'(k-1) = \mathrm{A}_L(f(k-1)) \in \mathrm{LT}_k$, and so $(k, f') \in \mathrm{LGT}$. It follows that $\mathrm{A}_R(k, f) = \mathrm{LGraft}(k, f') \in \mathrm{RGT} \subset \mathrm{RT}_k$. A similar argument shows that $\mathrm{A}_L(0, f) \in \mathrm{LGT} \subset \mathrm{LT}_k$. We conclude both functions are well-defined.

Lemma 13: *For $T \in k\mathrm{Tree}$, each of $\mathrm{LSeq}(\mathrm{A}_R(T))$ and $\mathrm{LSeq}(\mathrm{A}_L(T))$ are either equal to $w = \mathrm{LSeq}(T)$, or obtained from w by a single substitution.*

Proof. Note that if $T = \varnothing$, then:

$$\mathrm{LSeq}(\mathrm{A}_R(T)) = \mathrm{LSeq}(\Lambda_0) = \rho_0 \qquad \mathrm{LSeq}(\mathrm{A}_L(T)) = \mathrm{LSeq}(\Lambda_1) = \rho_1$$

and both ρ_0, ρ_1 are obtained from $0 = \mathrm{LSeq}(T)$ by a single substitution.

If $T = (n, f)$ where $n \neq 0, k$ then $\mathrm{A}_R(T) = \mathrm{A}_L(T) = T$, and so the result holds in this case.

Assume $T = (0, f)$. Then $T' = \mathrm{A}_L(T) = \mathrm{RGraft}(0, f')$ with f' defined as in Definition 25. Then:

$$\mathrm{LSeq}(T') = \mathrm{LSeq}(\mathrm{RGraft}(0, f')) = \mathrm{LSeq}(0, f')$$

by Lemma 12. From our definition of f', we have:

$$\begin{aligned}
\mathrm{LSeq}(0, f') &= (\mathrm{LSeq}(f'(0)))^{+2} \oplus (\mathrm{LSeq}(f'(1)))^{+1} \oplus \bigoplus_{i=2}^{k} (\mathrm{LSeq} f'(i))^{+1} \\
&= (\mathrm{LSeq}(f(0)))^{+2} \oplus (\mathrm{LSeq}(\mathrm{A}_R(f(1))))^{+1} \oplus \bigoplus_{i=2}^{k} (\mathrm{LSeq} f(i))^{+1}
\end{aligned}$$

By the induction hypothesis, we have that $\mathrm{LSeq}(\mathrm{A}_R(f(1)))$ is obtained from $\mathrm{LSeq}(f(1))$ by at most a single substitution, and so it follows that $\mathrm{LSeq}(T')$ is obtained by at most a single substitution from:

$$(\mathrm{LSeq}(f(0)))^{+2} \oplus (\mathrm{LSeq}(f(1)))^{+1} \oplus \bigoplus_{i=2}^{k} (\mathrm{LSeq} f(i))^{+1} = \mathrm{LSeq}(T)$$

A similar argument shows the same relationship between $\mathrm{LSeq}\mathrm{A}_R(T)$ and $\mathrm{LSeq}(T)$ in the case where $T = (k, f)$. □

Lemma 14: *If $T = (n, f)$ represents the k-subdivision r, and x is a breakpoint of r which lies in the i^{th} interval $[b_i, b_{i+1}]$ of κ_n, then $\frac{x-b_i}{b_{i+1}-b_i}$ is a breakpoint of the k-subdivision represented by $f(i)$.*

Proof. We note first that $T = (n, f)$ is a refinement of Λ_n, that $\mathrm{LSeq}(\Lambda_n) = \rho_n$, and so by Lemma 6, $\mathrm{LSeq}(T)$ is a refinement of ρ_n. From the definition of LSeq, this refinement w is obtained by replacing each $\rho_n(i)$ with $(w_i)^{+\rho_n(i)}$ where $w_i = \mathrm{LSeq}(f(i))$.

By our refinement-preserving bijection toSub, we then note that $r = \mathrm{toSub}(w)$ is obtained from κ_n by replacing the the i^{th} interval, $[b_i, b_{i+1}]$ with $\phi_{b_i, b_{i+1}}(r')$ where $r' = \mathrm{toSub}(w_i)$ is the k-subdivision represented by $f(i)$.

Thus, if x is a breakpoint of r in $[b_i, b_{i+1}]$, it is $\phi_{b_i, b_{i+1}}(x')$ for some breakpoint of r'. It follows that r' contains as a breakpoint:

$$\phi^{-1}_{b_i, b_{i+1}}(x) = \frac{x - b_i}{b_{i+1} - b_i}$$

□

Lemma 15: *For $T \in k\mathrm{Tree}$, if $\mathrm{A}_L(T) \not\sim T$, then the subdivision r represented by T does not contain τ_k as a break-point. Similarly, if $\mathrm{A}_R(T) \not\sim T$ then r does not contain $1 - \tau_k$.*

Proof. Note that if $T = \varnothing$, then $r = (\mathrm{toSub} \circ \mathrm{LSeq})(T) = [0, 1]$ contains neither τ_k nor $1 - \tau_k$. We now proceed by induction.

If $\mathrm{A}_L(n, f) \not\sim (n, f)$, then from our definition of A_L we see that:

1. $n = 0$
2. $\mathrm{A}_R(f(1)) \not\sim f(1)$

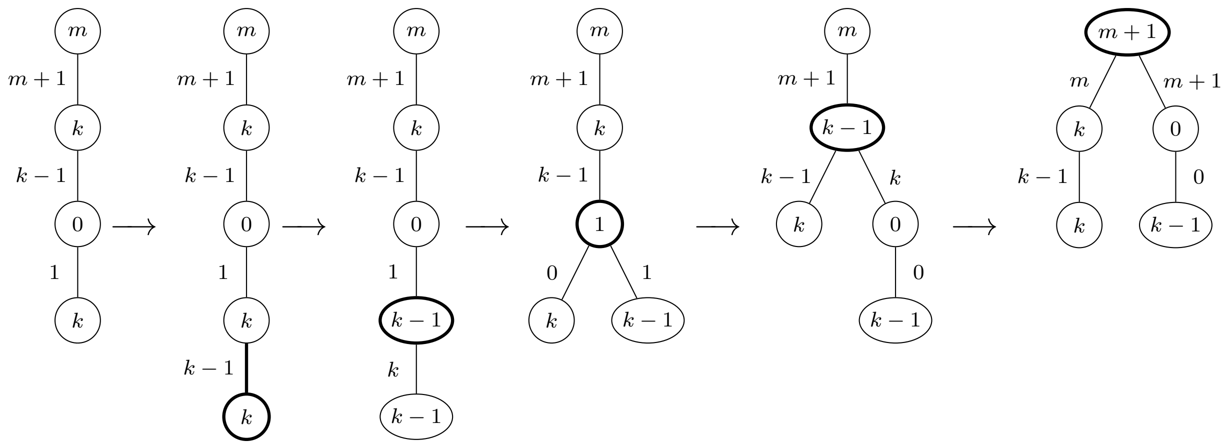


Figure 3.8: An example of the action of the function Incr, where each substitution of a leaf node and each graft operation is shown.

By induction, we conclude that $r' = (\text{toSub} \circ \text{LSeq})(f(1))$ doesn't contain $1 - \tau_k$ as a breakpoint.

We now note that $\tau_k \in [\tau_k^2, \tau_k^2 + \tau_k]$ which is the 1^{th} interval of κ_0 (note that we have been using zero indexing). From Lemma 14, $\tau_k \in r$ would imply $\frac{\tau_k - \tau_k^2}{\tau_k + \tau_k^2 - \tau_k^2} = 1 - \tau_k$ is a breakpoint of $f(1)$, but we have shown that this is not the case.

A similar argument gives the result for $\mathrm{A}_R(T)$. *is this ok?* $\square$

Definition 26 (Incr, Decr)**:** We define functions $\text{Incr} : \text{RT}_k \backslash \{\varnothing\} \to \text{LGT}$ and $\text{Decr} : \text{LT}_k \backslash \{\varnothing\} \to \text{RGT}$:

$\text{Incr}(n, f) = \text{RGraft}(n, f')$ where:

$$f'(i) = \begin{cases} f(i) & i \neq n+1 \\ \mathrm{A}_R(f(i)) & i = n+1 \end{cases}$$

$\text{Decr}(n, f) = \text{LGraft}(n, f')$ where:

$$f'(i) = \begin{cases} f(i) & i \neq n-1 \\ \mathrm{A}_L(f(i)) & i = n-1 \end{cases}$$

Remark. Note that $\text{Incr}|_{\text{RGT}} = \text{RGraft}$ and $\text{Decr}|_{\text{LGT}} = \text{LGraft}$.

We can think of the functions Incr and Decr as algorithms acting on k-trees, transforming them by a sequence of graft operations and possibly the substitution of a leaf node with a simple tree. Figure 3.8 demonstrates this process for a sample input to the function Incr.

Lemma 16: *For $T = (n, f)$, if $\text{Incr}(T) \not\sim T$ then $r = (\text{toSub} \circ \text{LSeq})(T)$ does not contain $(n+1)\tau_k$ as a breakpoint. If $\text{Decr}(T) \not\sim T$ then $(\text{toSub} \circ \text{LSeq})(T)$ does not contain $(n-1)\tau_k$ as a breakpoint.*

Proof. As in the proof for Lemma 15, we see that if $\text{Incr}(T) \not\sim T$, then:

$$\text{LSeq}(\mathrm{A}_R(f(n+1))) \neq \text{LSeq}(f(n+1))$$

from the definition of Incr. By Lemma 15 we conclude that $(\text{toSub} \circ \text{LSeq})(f(n+1))$ doesn't contain $(1-\tau_k)$ as a breakpoint.

Note that $(n+1)\tau_k \in [n\tau_k + \tau_k^2, (n+1)\tau_k + \tau_k^2]$ which is the 1^{th} interval of κ_0. From Lemma 14, $(n+1)\tau_k \in r$ would therefore imply that $\frac{(n+1)\tau_k - n\tau_k - \tau_k^2}{\tau_k} = 1 - \tau_k$ is a breakpoint of $f(1)$, but we have shown that this is not the case.

A similar argument gives the result for $\text{Decr}(T)$. □

Lemma 17: *If* $T = (n, f), T' = (m, g) \in k\text{Tree}$ *with* $n \leq m$ *satisfy* $T \sim T'$*, then* $\text{Incr}^{n-m}(T) \sim T$ *and* $\text{Decr}^{n-m}(T') \sim T'$

Maybe do an argument using weights of trees

Proof. The case where $m = n$ is trivial. For the case where $m > n$, it suffices to prove this in the case where $m = n + 1$. From this result we obtain our more general result by iteration.

We note that the subdivision r' represented by $T' = (n+1, g)$ contains the breakpoint $(n+1)\tau_k$, and thus so does the subdivision r represented by T. From the contrapositive of Lemma 16 we conclude that $\text{Incr}(T) \sim T$.

A similar argument gives the result for $\text{Decr}(T)$. □

Lemma 18: *If* $T \sim T'$ *then* T *can be transformed into* T' *by a sequence of graft operations.*

Proof. This is true for the case where $T = \varnothing$, since if $T' \sim \varnothing \iff T' = \varnothing$.

If $T = (n, f)$, and $T' = (m, g)$, then either Incr^{m-n} or Decr^{n-m} will transform T into a tree $\hat{T} = (m, f')$ with $T \sim \hat{T}$ by Lemma 17. Because $T \sim \hat{T}$, it follows from the definitions of Incr and Decr that $\hat{T}$ is obtainable from T by a sequence of graft operations.

By the fact that $(m, f') \sim (m, g)$, we see that $f'(i) \sim g(i)$ for all $i \in [k]$. By the induction hypothesis, it follows that $f'(i)$ can be transformed into $g(i)$ by a sequence of graft operations for each i.

Thus, the sequence of graft operations transforming T into $\hat{T}$ followed by the sequence of graft operations transforming each of the subtrees transforms T into T'. □

Note that this proof establishes an algorithm for transforming one tree into another. There is a general algorithm which transforms a tree T *into a refinement of another,* T'*, in a minimal way, which transforms* T *to* T' *if* $T \sim T'$*. I may explicitly describe this algorithm, as it gives a constructive way to describe tree products later.*

Lemma 19: *If* T_1 *and* T_2 *are* k*-trees such that every proper non-leaf subtree is of type* 0 *or* 1*, and* $T_1 \sim T_2$*, then there exists a sequence of graft operations transforming* T_1 *into* T_2 *such that the result of each graft operation is another* k*-tree with with proper subtrees of types* 0 *and* 1.

Proof. In the scope of this proof we will use the phrase "good tree" to mean a k-tree without proper subtrees of types greater than 1, and "good graft operation" to mean a graft operation which transforms a good tree into another good tree.

Note that the result holds trivially in the case $k = 1$, so we will henceforth assume $k > 1$.

We will prove the result by induction on T_1, noting that the result holds in the base case $T_1 = \varnothing$.

Assume that $T_1 = (n, f) \sim T_2 = (m, g)$ are good trees. If $n = m$, then the result holds by the induction hypothesis. We therefore assume that $n < m$ without loss of generality, as the relation "obtainable by a sequence of good graft operations" is symmetric. As with the general case in Lemma 18 it suffices to consider the case where $m = n + 1$.

We know that $\mathrm{Incr}(T) = \hat{T} = (n + 1, f')$ is a k-tree of type $n + 1$. Moreover, from our definition of Incr, we see that $\mathrm{Incr}(T) = \mathrm{RGraft}(T)$, as $\mathrm{A}_R(f(n + 1)) = f(n + 1)$ by the fact that $f(n + 1)$ is of type less than 2, and thus less than k (by the hypothesis $k > 1$).

If $\hat{T}$ is a good tree then by our induction hypothesis we can transform each $f'(i)$ into $g(i)$ by a sequence of good graft operations (noting that $f'(i) \sim g(i)$) and thus the result holds.

It remains only to consider the case where $\hat{T} = \mathrm{RGraft}(T)$ contains a proper subtree of type 2. From our definition of RGraft this can only occur when $f(n + 1) = (1, h)$ for some h, resulting in $f'(n) = (2, h')$ being the unique non-good proper subtree of $\hat{T}$. By our previous argument it suffices to demonstrate a sequence of good graft operations which produce a k-tree of type $(n + 1)$ from T in this case.

By the induction hypothesis, there exists a sequence of good graft operations transforming $f'(n) = (2, h')$ into $g(0)$. Because the type of g is less than 1, it follows that one of these graft operations is a left graft at the root, since otherwise the type of $(2, h')$ cannot be decreased. Thus, there exists a (possibly empty) sequence S of good graft operations which changes $h'(1)$ into a good k-tree of type 1 (since otherwise a left graft cannot be performed on $(2, h')$).

Having established this fact, we can describe a sequence of good graft operations transforming T_1 into a good k-tree of type $n + 1$, from which the result follows:

1. Perform the sequence S of graft operations on $h(0) = h'(1)$
2. Perform a left graft at $f(n + 1) = (1, h)$ to replace it with a subtree of the form $(0, h'')$.
3. Perform a right graft at the root

□

The idea behind this result is that a sequence of graft operations between two good k-trees is not necessarily good but can always be rearranged into a sequence where every graft operation is good. Figure 3.9 shows how alternating steps 2 and 3 from the above proof gives two different ways to obtain the same k-tree, only one of which is good.

3.2.2 Normal Trees

We will define a property on k-trees which will prove useful in our later construction of the normal form on $\mathcal{F}_{\tau_k}$.

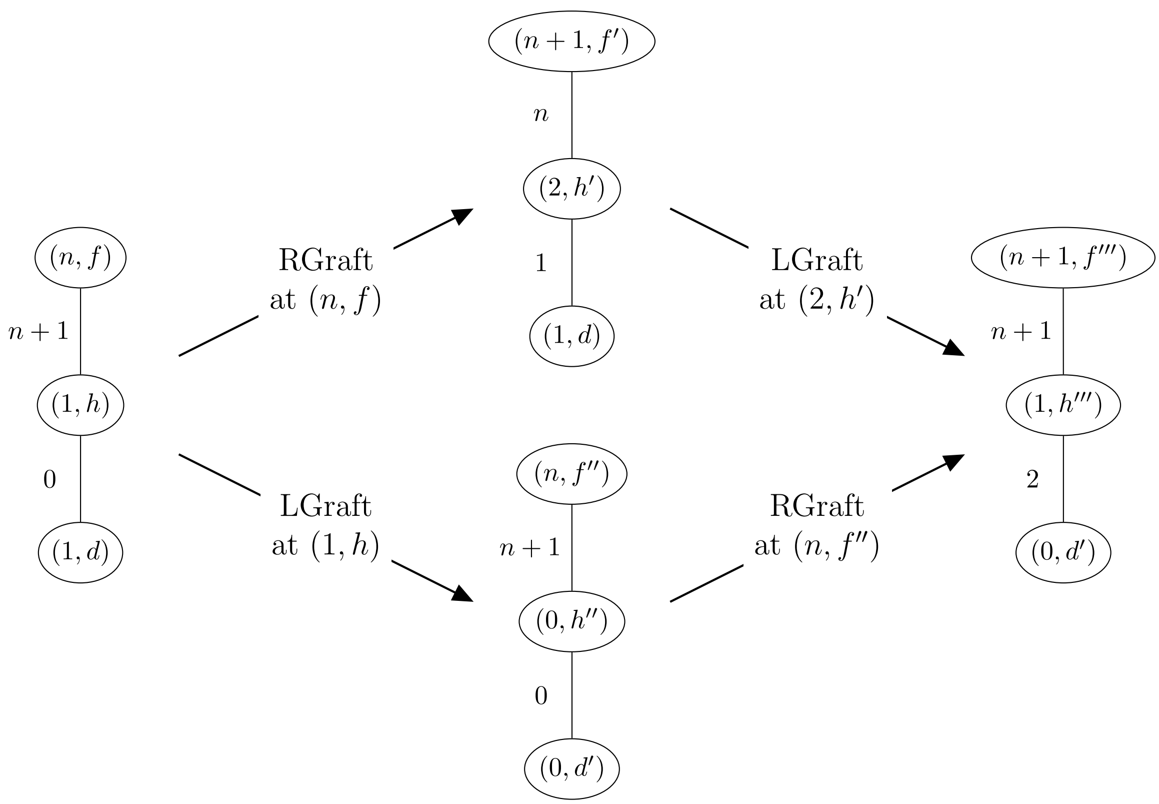


Figure 3.9: This figure illustrates the idea that if the right-most tree can be obtained by graft operations then it can be obtained by good graft operations. Note that this is achieved here by taking the lower path from left to right.

Definition 27 (normal k-tree)**:** We will say a k-tree is *normal* if all of its non-leaf subtrees are of type 0 or 1, and for every subtree $(1, f)$ of type 1, $f(0) = \varnothing$. We will say a k-tree is semi-normal if it is of the form (n, f) with $f(i)$ normal for all $i \in [k]$.

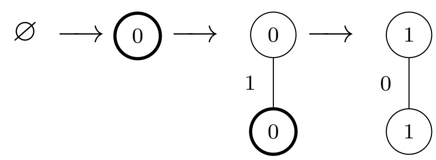
Lemma 20: *If T is a semi-normal, then by replacing leaves with Λ_0 and performing graft operations we can produce another semi-normal k-tree T with type one lower than T, for $T \in \mathrm{LT}_k$, or one greater for $T \in \mathrm{RT}_k$.*

Proof. Note that this result naturally only applies to non-leaf k-trees. For the sake of this proof, we will extend this result to the tree $\varnothing$ by declaring "type one lower than $\varnothing$" to be 0, and "type one greater than $\varnothing$" to be 1. This allows us to use a simpler inductive proof.

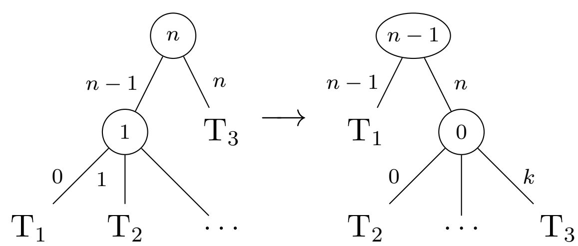
First we show that this is true for the base case $T = \varnothing$. Clearly we can substitute T for Λ_0 and thus decrease the type of $\varnothing$. We can also increase the type of $\varnothing$ as follows:

1. Substitute $\Lambda_0 = (0, \mathrm{nc})$ for $\varnothing$
2. Substitute Λ_0 for $\mathrm{nc}(1) = \varnothing$ to obtain T'
3. Let $\hat{T} = \mathrm{RGraft}(T')$

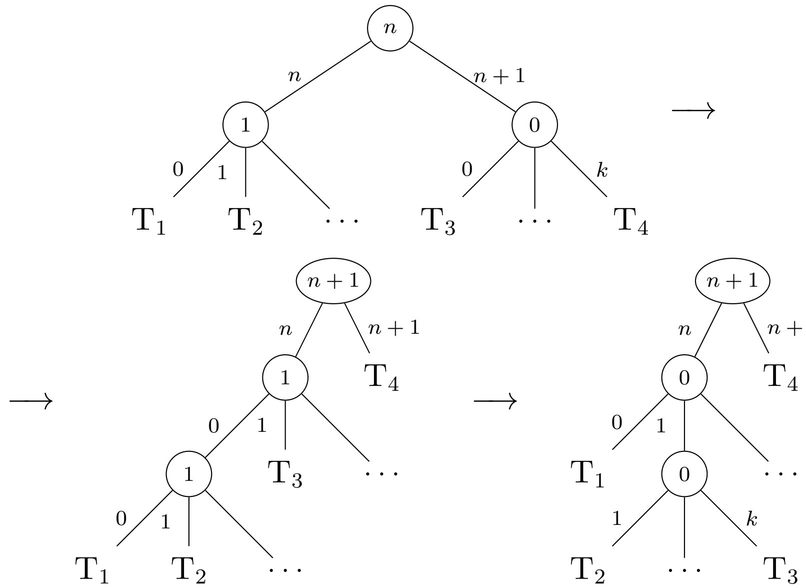
The effect of these steps is shown below. Note the desired output.



Now assume $T = (n, f)$ for $n > 0$. By the induction hypothesis we can transform the subtree $f(n-1)$ to be of type 1 and semi-normal by the addition of Λ_0's and graft operations. It therefore suffices to consider the case where $f(n-1)$ is of type 1 and merely semi-normal. By the diagram below we see that performing a left graft operation at the root produces a k-tree of the desired form. The fact that the resulting k-tree is semi-normal follows from the fact that each T_i is normal by the hypothesis that T and $f(n-1)$ were semi-normal.



A similar argument shows us that we can increase the type of T when $n < k$. By the induction hypothesis we can suitably transform $f(n+1)$ to be a semi-normal tree of type 0 and we can suitably transform $f(n)$ to be a semi-normal tree of type 1. We then perform a right graft at (n, f) to obtain $(n+1, f')$ followed by a left graft at the type 1 subtree $f'(n)$ as shown below.



□

Corollary 3: *Every semi-normal tree can be converted to a normal tree by graft operations and Λ_0 substitutions.*

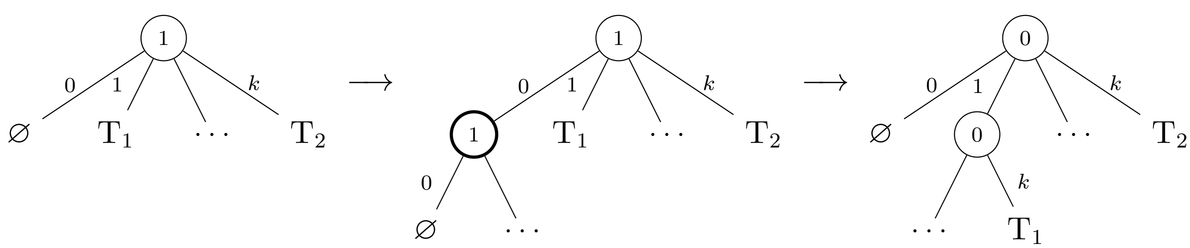
Proof. Let T be a semi-normal tree. By Lemma 20 we can continue to decrease the type of the root, with the T remaining semi-normal throughout the process, until the type of the root node of T is 0. At this point T will be a normal tree, since any semi-normal tree with root node of type 0 is normal. □

Lemma 21: *Every k-tree can be converted to a normal k-tree by graft operations and Λ_0 substitutions.*

Proof. We prove this by induction. Clearly $\varnothing$ is normal. If $T = (n, f)$, then by induction each of $f(i)$ can be converted to normal k-trees by a suitable process, thus transforming T into a semi-normal tree. From Corollary 3, T can then be converted into a normal tree. □

Lemma 22: *If T is a normal k-tree, then it can be transformed by a sequence of Λ_1 substitutions and graft operations into a k-tree with all non-leaf subtrees of type* 0.

Proof. We can prove this by induction, noting that it is trivially true in the case $T = \varnothing$. If $T = (0, f)$, then the result follows from the induction hypothesis. It suffices to consider the case $T = (1, f)$. By the assumption that T is normal, $f(0) = \varnothing$. We can therefore perform a Λ_1 substitution of $f(0)$ and then perform a left graft operation at the root:



By the induction hypothesis the resulting k-tree can suitably be converted to a k-tree with all subtrees of type 0. □

3.3 k-subdivision and $\mathbb{Z}[\tau_k]$

Thompson's groups are studied using binary subdivisions and trees. Binary subdivisions are defined recursively as either the interval $[0,1]$ or the result of splitting an interval in a binary subdivision evenly in two. These subdivisions have the property that for any finite subset $S \subset \mathbb{Z}[\frac{1}{2}] \cap [0,1]$ there exists a binary subdivision whose breakpoints contain S. This is also true for k-subdivisions. We will prove this by way of two intermediary results:

1. Every generalised k-subdivision has a refinement which is a k-subdivision
2. For any finite subset $S \subset \mathbb{Z}[\tau_k] \cap [0,1]$ there exists a generalised k-subdivision whose breakpoints contain S

Lemma 23: *Let u and v be two uniform k-sequences of level l such that for some $j \in \mathbb{Z}_{\geq 0}$:*

$$u(j) = l + 1 = v(j+1) \qquad v(j) = l = v(j+1)$$

Let u' be a refinement of u. Then there exists a common refinement of u' and v.

Proof. We prove this using properties of generalised k-trees. Assume u, v, j and u' satisfy the hypothesis of the Lemma.

By Lemma 7, $v = \mathrm{GenLSeq}(v, \mathrm{nc})$, $u = \mathrm{GenLSeq}(u, \mathrm{nc})$ and there exists some generalised k-tree $T = (u, f)$ such that $\mathrm{GenLSeq}(T) = u'$.

Let $T' = (u, f')$ where:

$$f'(i) = \begin{cases} f(i) & i \neq j+1 \\ \mathrm{A}_R(f(i)) & i = j+1 \end{cases}$$

We note that T' is a refinement of T, and that $T' \in \mathrm{GenRGT}_j$. From our definition of $\mathrm{GenRGraft}_n$, $\hat{T} = \mathrm{GenRGraft}_n(T') = (v, g)$ for some $g : [s] \to k\mathrm{Tree}$, and thus is a refinement of (v, nc).

By Lemma 7, $w = \mathrm{GenLSeq}(T')$ is a refinement of u' and $w' = \mathrm{GenLSeq}\left(\hat{T}\right)$ is a refinement of v. By Lemma 9, we have $w = w'$, and so w is a common refinement for u' and v. $\square$

Lemma 24: *If v is a uniform k-sequences of level l, u is the uniform k-sequence at level l with all $l+1$'s to the left of l's, and u' is a refinement of u then there exists a common refinement of u' and v.*

Proof. Assume u, v and u' satisfy the hypothesis of the Lemma. By Lemma 4 v is a permutation of u. We can therefore get from u to v by performing a sequence $s_1, s_2, \ldots, s_n$ of switches of the form $w_l, (l+1), l, w_r \mapsto w_l, l, (l+1), w_r$. Let $v_0 = u$, $v_1 = s_1(v_0)$, $v_2 = s_2(v_1)$ and so on up to $v = v_n = s_n(v_{n-1})$.

We now obtain our result by induction. Clearly u' is a common refinement of u' and $v_0 = u$. Now assume we have a common refinement u_i for u' and v_i. Then v_i, v_{i+1} and u_i satisfy the conditions on u, v and u' from Lemma 23. There therefore exists a common refinement u_{i+1} of u_i and v_{i+1}, which is thus a common refinement of u' and v_{i+1}.

By induction we conclude that there exists a common refinement of u' and $v = v_n$. $\square$

Lemma 25: *Any two generalised k-sequences have a common refinement*

Proof. Let u and v be two generalised k-sequences. Let l_u and l_v be the maximum term of u and v respectively.

Then by Lemmas 2 and 3 there exist refinements u' and v' of u and v which are both uniform of level $l = \max(l_u, l_v) - 1$. Let w be the uniform k-sequence at level l with all $(l+1)$'s to the right of l's. Then by Lemma 24 there exists a common refinement w' of w and u' and a common refinement w'' of w' and v', which is therefore a common refinement of u' and v' and thus of u and v. $\square$

Lemma 26: *If u is a generalised k-sequence then it has a refinement which is a k-sequence.*

Proof. Let v be any k-sequence (for example 0). Then by Lemma 25 there exists a common refinement w of u and v. Because w is a refinement of the k-sequence v, w is also a k-sequence. $\square$

Corollary 4: *Every generalised k-subdivision has a refinement which is a k-subdivision.*

Proof. This follows from our isomorphism $\mathrm{GentoSeq} : \mathrm{Gen}k\mathrm{Sub} \to \mathrm{Gen}k\mathrm{Seq}$ which preserves refinements and maps k-subdivisions to k-sequences. $\square$

This was part one of our proof that finite subsets in $\mathbb{Z}[\tau_k] \cap [0,1]$ are subsets of breakpoints of some k-subdivision. It remains to show that every such subset is a subset of breakpoints of a generalised k-subdivision.

Lemma 27: *If $x \in \mathbb{Z}[\tau_k]$ then there exists $m, n, N \in \mathbb{Z}$ such that*

$$x = m\tau_k^N + n\tau_k^{N+1}$$

Proof. It suffices to show that for any $r, N \in \mathbb{Z}$ with $r \le N+1$, $\tau_k^r = m\tau_k^N + n\tau_k^{N+1}$ for some $m, n \in \mathbb{Z}$. Once we have proven this for $x = \tau_k^r$, then the result follows for any $x = \sum_{i=0}^{l} \alpha_i \tau_k^i$, since we can write each $\alpha_i \tau_k^i$ term in the form $m\tau_k^l + n\tau_k^{l+1}$.

We can prove this by induction on the difference between r and N. First note that the proposition holds trivially if $r = N+1$ or if $r = N$.

Assume that the result holds in the case where $N + 1 \ge r \ge N - i$ for some $i \ge 0$.

Then if $r = N - i - 1$ we can write:

$$\tau_k^r = (\tau_k^2 + k\tau_k)\tau_k^r = \tau_k^{r+2} + k\tau_k^{r+1} = \tau_k^{N-i+1} + k\tau_k^{N-i}$$

By the induction hypothesis, there exists $m, n, m', n' \in \mathbb{Z}$ satisfying:

$$\tau_k^{N-i+1} = m\tau_k^N + n\tau_k^{N+1}, \qquad \tau_k^{N-i} = m'\tau_k^N + n'\tau_k^{N+1}$$

So we have that:

$$\tau_k^r = (m + km')\tau_k^N + (n + kn')\tau_k^{N+1}$$

$\square$

Lemma 28: *If $x = m\tau_k^N + n\tau_k^{N+1}$ then there exist $m', n' \in \mathbb{Z}$ such that $x = m'\tau_k^{N+1} + m'\tau_k^{N+2}$ with $\tau_k m' - n' = -\tau_k(\tau_k m - n)$*

Proof. Take $m' = n + km$ and $n' = m$. Then:

$$\begin{aligned} m'\tau_k^{N+1} + n'\tau_k^{N+2} &= (n+km)\tau_k^{N+1} + m\tau_k^{N+2} \\ &= n\tau_k^{N+1} + m\tau_k^N\left(k\tau_k + \tau_k^2\right) \\ &= n\tau_k^{N+1} + m\tau_k^N \\ &= x \end{aligned}$$

We also have that:

$$\begin{aligned} \tau_k m' - n' &= \tau_k(km+n) - m \\ &= \tau_k(km+n) - (\tau_k^2 + k\tau_k)m && (\tau_k^2 + k\tau_k = 1) \\ &= \tau_k km + \tau_k n - \tau_k^2 m - \tau_k km \\ &= \tau_k n - \tau_k^2 m \\ &= -\tau_k(\tau_k m - n) \end{aligned}$$

□

Lemma 29: *If $x \in \mathbb{Z}[\tau_k]$ and $x \geq 0$ then there exists $m, n, N \in \mathbb{Z}$ with $m, n \geq 0$ such that $x = m\tau_k^N + n\tau_k^N$*

Proof. The result holds trivially for $x = 0$, so we will assume for the rest of the proof that $x > 0$.

By Lemma 27 we know that there exists some $m, n, N \in \mathbb{Z}$ with m and n not necessarily positive, such that $x = m\tau_k^N + n\tau_k^{N+1}$.

By repeated use of Lemma 28 we can then find for any $l \in \mathbb{Z}_{>0}$ another pair $m', n' \in \mathbb{Z}$ such that $x = m'\tau_k^{N+l} + n'\tau_k^{N+l+1}$ with $|\tau_k m' - n'| = \tau_k^l|\tau_k m - n|$. Now chose l such that $N + l \geq 2$ and $\tau_k^l|\tau_k m - n| < \frac{x}{2}$ (recall that $0 < \tau_k < 1$). Then we have $m', n' \in \mathbb{Z}$ with $x = m'\tau_k^{N+l} + n'\tau_k^{N+l+1}$ and $|\tau_k m' - n'| < \frac{x}{2}$.

Now we show that n' must be positive:

$$\begin{aligned} & m'\tau_k^{N+l} + n'\tau_k^{N+l+1} = x \\ \implies & \left(n' + \frac{x}{2}\right)\tau_k^{N+l-1} + n'\tau_k^{N+l+1} > x && (|\tau_k m' - n'| < \tfrac{x}{2} \implies \tau_k m' < n' + \tfrac{x}{2}) \\ \implies & n'\left(\tau_k^{N+l-1} + \tau_k^{N+l+1}\right) > x - \frac{x}{2}\tau_k^{N+l-1} \\ \implies & n'\left(\tau_k^{N+l-1} + \tau_k^{N+l+1}\right) > \frac{x}{2} && (x - \tfrac{x}{2}\tau_k^{N+l-1} \geq x - \tfrac{x}{2} \text{ since } N + l \geq 1) \\ \implies & n' > \frac{x}{2} && (\left(\tau_k^{N+l-1} + \tau_k^{N+l+1}\right) < (\tau_k + k\tau_k^2) = 1) \end{aligned}$$

But since $(\tau_k m' - n') > -\frac{x}{2}$ we have that:

$$\tau_k m' > n' - \frac{x}{2} > 0$$

and so $m' > 0$ as well. □

Lemma 30: *If $S \subset \mathbb{Z}[\tau_k] \cap [0,1]$ is finite then there exists a generalised k-subdivision whose breakpoints are a superset of S*

Proof. Assume S satisfies the hypothesis. Consider the subdivision R of the unit interval whose breakpoints are the elements of $S \cup \{0,1\}$. Each interval $[a,b]$ of this subdivision satisfies $b-a \in \mathbb{Z}[\tau_k] \cap [0,1]$, and so by Lemma 29 there exists some $m,n,N \in \mathbb{Z}_{\geq 0}$ with $b-a = m\tau_k^N + n\tau_k^{N+1}$. It follows that the interval $[a,b]$ could be replaced with m intervals of length τ_k^N and n intervals of length τ_k^{N+1}. Repeating this for each interval in the original subdivision R produces a generalised k-subdivision whose breakpoints contain the elements of S. $\square$

Corollary 5: *If $S \subseteq \mathbb{Z}[\tau_k] \cap [0,1]$ is finite then there exists a k-subdivision whose breakpoints are a superset of S.*

Proof. This follows from Lemma 30 and Corollary 4. $\square$

4. The Group $\mathcal{F}_{\tau_k}$

4.1 $\mathcal{F}_{\tau_k}$ as pairs of k-subdivisions

Thompson's group F has the property that any element of Thompson's group can be represented by a pair of subdivisions, and conversely that every pair of subdivisions with the same number of intervals represents an element of F. We are now in a position to be able to establish the same result for the groups $\mathcal{F}_{\tau_k}$ and k-subdivisions.

Definition 28 (kSubPair, Intpl)**:** Let kSubPair denote the set of pairs of k-subdivisions with equal length. We describe a function $\text{Intpl} : k\text{SubPair} \to \mathcal{F}_{\tau_k}$ by mapping P, Q to the function $f_{P,Q} : [0,1] \to [0,1]$ which maps the breakpoints of P to the breakpoints of Q and then interpolates linearly:

$$f_{P,Q}(x) := q_i + \frac{q_{i+1} - q_i}{p_{i+1} - p_i}(x - p_i) \qquad x \in [p_i, p_{i+1}]$$

Studying the graph of such a function $f_{P,Q}$ is the best way to understand this construction. An example of a pair of k-subdivisions and its corresponding function is given in Figure 4.1. That this function is well-defined with codomain $\mathcal{F}_{\tau_k}$ is shown by Lemma 31.

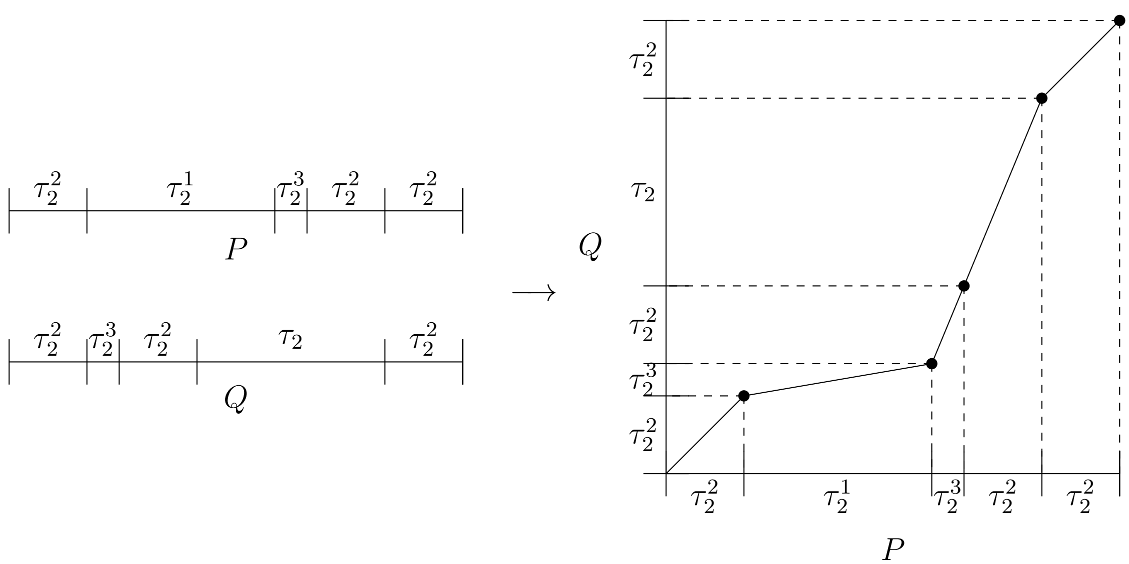


Figure 4.1: A visual depiction of the map $F : \mathcal{P}_3 \to \mathcal{F}_{\tau_3}$

Lemma 31: *If* $(P, Q) \in k\text{SubPair}$*, then* $f_{P,Q} \in \mathcal{F}_{\tau_k}$

Proof. We check each of the conditions on elements of the group $\mathcal{F}_{\tau_k}$:

1. $\boldsymbol{f_{P,Q}}$ **is piece-wise linear:** this follows directly from the definition

2. $\boldsymbol{f_{P,Q}}$ **is a homeomorphism:** as a map from a compact space to a Hausdorff space, it suffices to show that f is continuous and bijective. Continuity follows from the Pasting Lemma since the finitely many closed sets $[p_i, p_{i+1}]$ cover $[0, 1]$ and the restriction of $f_{P,Q}$ to each closed set is affine, and therefore continuous. Injectivity follows from the fact that f is strictly increasing. Surjectivity follows from the Intermediate Value Theorem and the fact that 0 and 1 are in the image of f

3. **The breakpoints of** $\boldsymbol{f_{P,Q}}$ **lie in** $\mathbb{Z}[\boldsymbol{\tau_k}]$**:** because the breakpoints of $f_{P,Q}$ are the breakpoints of P and the breakpoints of a k-subdivision lie in $\mathbb{Z}[\tau_k]$.

4. **The slopes of the affine segments of** $\boldsymbol{f_{P,Q}}$ **are powers of** $\boldsymbol{\tau_k}$**:** Each k-subdivision is in particular a generalised k-subdivision, so each $(p_{i+1} - p_i)$ and each $(q_{i+1} - q_i)$ is a power of τ_k. The slope of each affine segment given by $\frac{q_{i+1}-q_i}{p_{i+1}-p_i}$ is therefore also a power of τ_k.

□

Our proof that Intpl is surjective will rely on some technical results. To improve fluency we will use the phrase "r respects f" to mean the breakpoints of r, a subdivision, contain the breakpoints of $f \in \mathcal{F}_{\tau_k}$.

Lemma 32: *If* $r \in \mathrm{Gen}k\mathrm{Sub}$ *respects* f *and* r' *is obtained from* r *by an* n*-partition of the* j^{th} *interval of* r*. Then:*

1. $f(r) \in \mathrm{Gen}k\mathrm{Sub}$
2. $q' = f(r')$ *is obtained from* $q = f(r)$ *by the same partition*
3. $f = \mathrm{Intpl}(r, q)$

Proof. For 1: each interval I of r lies between two adjacent breakpoints of f, so the length of $f(I)$ is a multiple of the length of I by some power of τ_k, and thus a power of τ_k by the fact that $r \in \mathrm{Gen}k\mathrm{Sub}$.

For 2: assume the j^{th} interval is $[a, b]$. Then r' is obtained by replacing $[a, b]$ with $\phi_{a,b}(\kappa_n)$, so $f(r')$ is obtained by replacing $f([a, b])$ with $f(\phi_{a,b}(\kappa_n)) = \phi_{f(a),f(b)}(\kappa_n)$ by the fact that f is affine on $[a, b]$.

For 3: note that we implicitly extend the definition of Intpl to $\mathrm{Gen}k\mathrm{Sub}$ in the obvious way. The result holds because f is affine between its breakpoints, and the linear interpolation of the image of points under an affine map is that same affine map. *Could I explain this better?* □

Lemma 33: $\mathrm{Intpl} : k\mathrm{SubPair} \to \mathcal{F}_{\tau_k}$ *is surjective.*

Proof. We need to show that given any element $f \in \mathcal{F}_{\tau_k}$ we have a $(P, Q) \in k\mathrm{SubPair}$ such that $f_{P,Q} = f$. We can do so as follows:

1. Let $S = (s_0 = 0 < s_1 < \cdots < s_l = 1) \subseteq \mathbb{Z}[\tau_k]$ be the set of breakpoints of f.

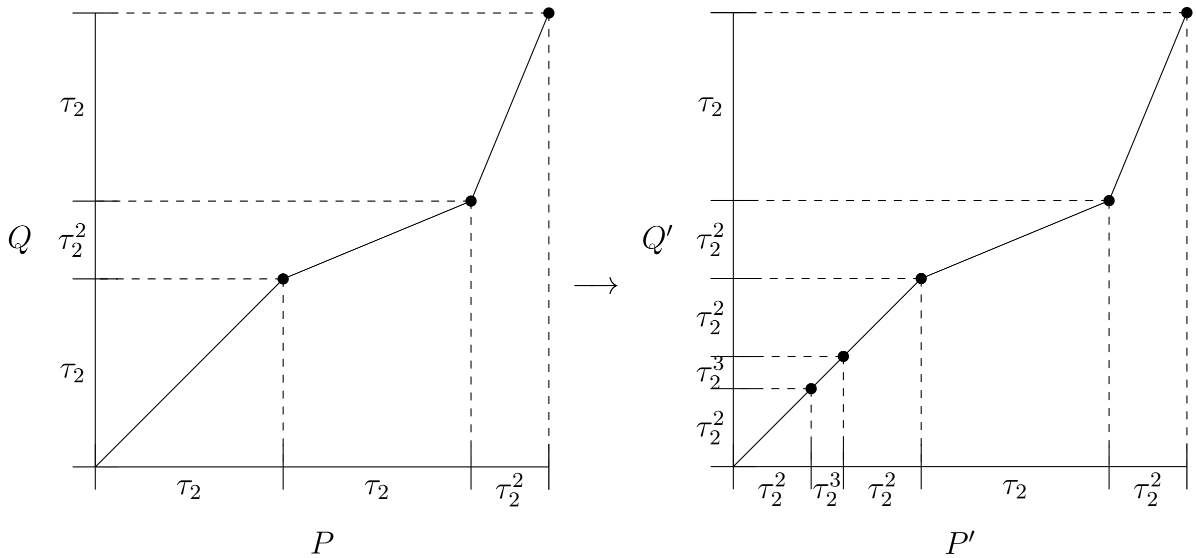


Figure 4.2: Performing the same k-partition on corresponding intervals leaves the element $f_{P,Q}$ unchanged.

2. By Lemma 5 there exists a k-subdivision, R, whose set of breakpoints contains S as a subset.
3. By Lemma 32 part 1 $K = f(R) \in \text{Gen}k\text{Sub}$.
4. By Corollary 4, there exists a refinement Q of K which is a k-subdivision.
5. By Lemma 32 part 2, $P = f^{-1}(K')$ is a refinement of R, and thus a k-subdivision.
6. By Lemma 32 part 3, $f = \text{Intpl}(P, Q)$.

□

Note that the map Intpl is not injective: for any two k-subdivisions P and Q, $\text{Intpl}(P, P) = \text{Intpl}(Q, Q) = \text{id}_{[0,1]}$. More generally, for $(P, Q) \in k\text{SubPair}$ if we perform a k-partition on an interval in P to obtain P' and a k-partition of the same type on the corresponding interval in Q to obtain Q', then $\text{Intpl}(P', Q') = \text{Intpl}(P, Q)$. This is illustrated in Figure 4.2.

Lemma 34: *For any k-subdivisions r, p and q with the same number of intervals:*

$$\text{Intpl}(r, p)\,\text{Intpl}(p, q) = \text{Intpl}(r, q)$$

$$\text{Intpl}(r, p) = \text{Intpl}(p, r)^{-1}$$

Proof. These follow immediately form the definition of Intpl, recalling our convention that left multiplication in $\mathcal{F}_{\tau_k}$ is precomposition. □

Lemma 35: *For any k-subdivisions r, p and q with the same number of intervals, the following are equivalent:*

1. $r = p$

2. $\text{Intpl}(r, p) = \text{id}_{[0,1]}$

3. $\mathrm{Intpl}(q,r) = \mathrm{Intpl}(q,p)$

4. $\mathrm{Intpl}(r,q) = \mathrm{Intpl}(q,p)^{-1}$

5. $\mathrm{Intpl}(r,q) = \mathrm{Intpl}(p,q)$

Proof. 1 $\iff$ 2 follows immediately from the definition of Intpl. The other implications follow from Lemma 34.

For 2 $\implies$ 3:

$$\mathrm{Intpl}(q,r) = \mathrm{Intpl}(q,r)\,\mathrm{id}_{[0,1]} = \mathrm{Intpl}(q,r)\,\mathrm{Intpl}(r,p) = \mathrm{Intpl}(q,p)$$

For 3 $\implies$ 4:

$$\mathrm{Intpl}(r,q) = \mathrm{Intpl}(q,r)^{-1} = \mathrm{Intpl}(q,p)^{-1}$$

For 4 $\implies$ 5:

$$\mathrm{Intpl}(r,q) = \mathrm{Intpl}(q,p)^{-1} = \mathrm{Intpl}(p,q)$$

For 5 $\implies$ 2:

$$\mathrm{Intpl}(r,p) = \mathrm{Intpl}(r,q)\,\mathrm{Intpl}(q,p) = \mathrm{Intpl}(p,q)\,\mathrm{Intpl}(q,p) = \mathrm{Intpl}(p,p) = \mathrm{id}_{[0,1]}$$

□

Not sure I need to state/prove the last two lemmas, they are used later, but perhaps I could just note that the result is true when needed.

4.2 $\mathcal{F}_{\tau_k}$ as pairs of k-trees

Given our surjection from kSubPair to $\mathcal{F}_{\tau_k}$, we can now define a surjection from pairs of k-trees to $\mathcal{F}_{\tau_k}$:

Definition 29 (kTreePair, $H : k\mathrm{TreePair} \to \mathcal{F}_{\tau_k}$)**:** Let kTreePair denote the set of pairs of k-trees with the same number of leaves. Then we define $H : k\mathrm{TreePair} \to \mathcal{F}_{\tau_k}$ by the mapping:

$$(T_1, T_2) \mapsto \mathrm{Intpl}((\mathrm{toSub} \circ \mathrm{LSeq})(T_1), (\mathrm{toSub} \circ \mathrm{LSeq})(T_2))$$

Using the map H, we can lift the group structure of $\mathcal{F}_{\tau_k}$ to the set of fibres of H, giving a group isomorphic to $\mathcal{F}_{\tau_k}$:

Definition 30 ($\mathcal{T}_k$, $\widetilde{H}$)**:** Let $\mathcal{T}_k$ denote the set of fibres of H. Equivalently, $\mathcal{T}_k$ is the set kTreePair modulo the equivalence relation $x \sim y$ if $H(x) = H(y)$. We define a group structure $\star : \mathcal{T}_k \times \mathcal{T}_k \to \mathcal{T}_k$ by:

$$[x] \star [y] = H^{-1}(H(x)H(y)) = H^{-1}(H(y) \circ H(x))$$

We let $\widetilde{H} : \mathcal{T}_k \to \mathcal{F}_{\tau_k}$ denote the map induced by H, and note that it is a group isomorphism.

Lemma 36: *For any k-trees T_1, T_2 and S with the same number of leaf nodes, the following are equivalent:*

1. $T_1 \sim T_2$
2. $[T_1, T_2] = \mathrm{id}_{\mathcal{T}_k}$
3. $[S, T_1] = [S, T_2]$
4. $[T_1, S] = [S, T_2]^{-1}$
5. $[T_1, S] = [T_2, S]$

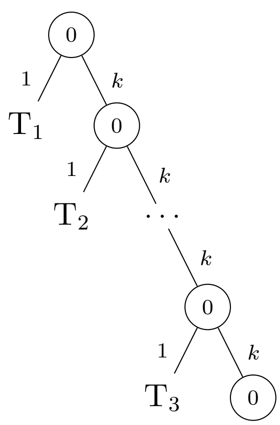
Proof. This follows immediately form Lemma 35. □

Definition 31 (Right-aligned k-tree)**:** A *right-aligned k-tree* a k-tree of the form $(0, f)$ where $f(k)$ is either $\varnothing$ or a right-aligned k-tree. Informally, it is a k-tree with all the nodes along its right edge are of type 0.

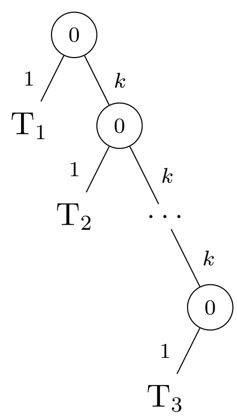
Definition 32 (Reduced right-aligned k-tree)**:** A *right-aligned k-tree* a k-tree of the form $(0, f)$ with $f \neq \mathrm{nc}$ where $f(k)$ is either $\varnothing$ or a right-aligned k-tree. Informally, it is a right-aligned k-tree where the right-most subtree is not simple.

Definition 33 (Spine)**:** A *spine* is either $\varnothing$ or a k-tree of the form $(0, f)$ with $f(i) = \varnothing$ for $i \neq k$, and $f(k)$ a spine.

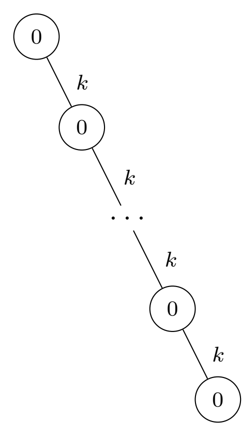
Compact tree diagrams representing each of the above types of trees are shown in Figure 4.3. Note that all spines and reduced right-aligned k-trees are right-aligned k-trees, but no spine is reduced.



(a) A right-aligned k-tree which is not reduced



(b) A reduced right-aligned k-tree



(c) A spine

Figure 4.3: Right-aligned k-trees. Note that T_i in the diagrams denotes a (possible empty) subtree

For a given k-tree T we will let $\mathrm{sp}(T)$ denote the unique spine with the same number of leaf nodes as T. This allows us to discuss a class of k-tree pairs of the form $(T, \mathrm{sp}(T))$. We will then use the notation $\widetilde{T}$ to denote the equivalence class $[T, \mathrm{sp}(T)] \in \mathcal{T}_k$.

Lemma 37: *Let T_1, T_2 be two k-trees with the same number of nodes. Let S be a spine. Then:*

1. $[\mathrm{sp}(T_1), T_1] = \widetilde{T_1}^{-1}$
2. $[T_1, T_2] = \widetilde{T_1} \star \widetilde{T_2}^{-1}$
3. $\widetilde{S} = \mathrm{id}_{\mathcal{T}_k}$
4. *If $T_1 \sim T_2$ then $\widetilde{T_1} = \widetilde{T_2}$*
5. *The set $\left\{\widetilde{T} \,|\, T \in k\mathrm{Tree}\right\}$ generates $\mathcal{T}_k$*

Proof. 1 follows directly from Lemma 36. 2 follows from 1 and the fact that $\mathrm{sp}(T_1) = \mathrm{sp}(T_2)$. 3 follows from Lemma 36 and the fact that $\mathrm{sp}(S) = S$. 4 follows from Lemma 36. 5 follows from 2 and the fact that every element of $\mathcal{T}_k$ can be expressed as $[T', T'']$ for some pair of k-trees T' and T''. □

Lemma 38: *If T_1 and T_2 are right-aligned k-trees with T_2 reduced and $T_1 \sim T_2$ then T_1 is also reduced.*

Proof. The right-most leaves of any such pair T_1, T_2 of k-trees must be of the same height h, given that T_1 and T_2 have the same leaf sequence. If T_1 were not reduced, then its lowest type 0 node along the right edge would have no children, and its leaf sequence would terminate with

$$\ldots h+1, \underbrace{h, \ldots, h}_{k\, h\text{'s}}$$

But this cannot be the last $k+1$ terms of the leaf sequence of T_2, because at least one of the children of the last type 0 node along the right edge of T_2 is not a leaf. We conclude that T_1 must be reduced. □

Lemma 39: *If T_1 and T_2 are right-aligned, reduced k-trees satisfying $\widetilde{T_1} \star \widetilde{T_2}^{-1} = \mathrm{id}_{\mathcal{T}_k}$, then $T_1 \sim T_2$.*

Proof. By Lemma 36 it suffices to show that T_1 and T_2 have the same number of nodes. Once we have established this, we then have that $\widetilde{T_1} \star \widetilde{T_2}^{-1} = [T_1, T_2]$ by Lemma 37 and then $T_1 \sim T_2$ by Lemma 36. We will show that T_1 and T_2 have the same number of nodes by contradiction.

Assume without loss of generality that T_2 has a number of nodes greater than that of T_1, and thus that the length of $\mathrm{sp}(T_2)$ is greater than that of $\mathrm{sp}(T_1)$. It follows that $\mathrm{sp}(T_2)$ is obtained from $\mathrm{sp}(T_1)$ by adding some number $d \geq 1$ of type 0 nodes to the right edge of $\mathrm{sp}(T_1)$. Let T_1' be the k-tree obtained from T_1 by adding that same number of type 0 nodes to its right edge. Note that T_1' is therefore right-aligned but no longer reduced. But then it follows that $(T_1', T_2) \in (T_1, \mathrm{sp}(T_1)) \star (\mathrm{sp}(T_2), T_2)$ and so $[T_1', T_2] = \mathrm{id}_{\mathcal{T}_k}$ (by our hypothesis) and thus $T_1' \sim T_2$ (by Lemma 36). By Lemma 38 we conclude that T_1' is reduced, which gives us a contradiction.

It follows that T_1 and T_2 have the same number of nodes, and therefore $T_1 \sim T_2$. □

4.3 A Generating Set for $\mathcal{F}_{\mathcal{T}_k}$

We now return our attention to the group $\mathcal{T}_k$, where we can use the results above to find representatives of a certain form in kTreePair for elements in $\mathcal{T}_k$.

Lemma 40: *For two k-trees, $T_1 \sim T_2$, if T_i' is obtained by a type n substitution of the j^{th} leaf of T_i, $i = 1, 2$, then $[T_1, T_2] = [T_1', T_2']$.*

Proof. From Lemma 5 we saw that a type n substitution of the j^{th} leaf in a k-tree corresponds to a type n substitution of the j^{th} term of the leaf sequence of that k-tree. Since $\mathrm{LSeq}(T_1) = \mathrm{LSeq}(T_2)$, it follows that $\mathrm{LSeq}T_1' = \mathrm{LSeq}T_2'$. $\square$

Not sure I need to state this lemma

Definition 34 (Normal Pair)**:** We say $(T_1, T_2) \in k\mathrm{TreePair}$ is *normal* if:

1. T_1 and T_2 are both right-aligned
2. T_1 is normal
3. T_2 contains only subtrees of type 0

Lemma 41: *Every element of $\mathcal{T}_k$ is represented by a normal pair.*

Proof. Assume $[T_1, T_2] \in \mathcal{T}_k$. By Lemmas 21 and 22 we can convert T_2 into a k-tree T_2' with all subtrees of type 0 by a sequence of graft operations and Λ_0 and Λ_1 substitutions of leaves. We note that if T' is obtained from T_2 by a graft operation, then $[T_1, T_2] = [T_1, T_2]$ by the fact that graft operations preserve leaf sequences (Lemma 12). If T' is obtained from T_2 by a substitution of a leaf, then we let T'' be the result of performing the same substitution on T_1, and we have $[T'', T'] = [T_1, T_2]$ by Lemma 40. Putting these results together, we note that by mirroring each substitution performed to obtain T_2' on the k-tree T_1, we obtain a new k-tree T_1' such that $[T_1', T_2'] = [T_1, T_2]$.

We can then transform T_1' into a normaltree T_1'' using only graft operations and Λ_0 substitutions. We can furthermore transform T_1'' into a right-aligned normal tree $\widehat{T_1}$ by decreasing if necessary the types of all subtrees along the right edge by Lemma 20. This argument relies on the fact that this transformation never requires a Λ_0 expansion of the right-most leaf, which is clear from the proof of Lemma 20.

By performing the corresponding Λ_0 substitutions on T_2', we obtain another k-tree $\widehat{T_2}$ also with only type 0 subtrees (and thus right-aligned) such that $[T_1, T_2] = \left[\widehat{T_1}, \widehat{T_2}\right]$. Note that $\left(\widehat{T_1}, \widehat{T_2}\right)$ is a normal pair. $\square$

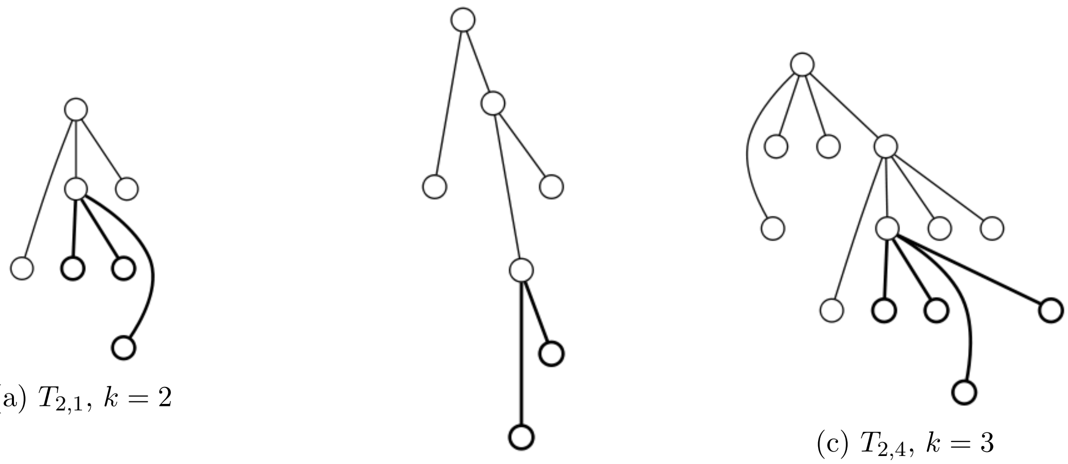
Corollary 6: *The group $\mathcal{T}_k$ is generated by the set $\left\{\widetilde{T} \,|\, T \text{ is normal}\right\}$*

Proof. From Lemma 41 we see that every element of $\mathcal{T}_k$ has a representation as $[T_1, T_2]$ where T_1 is normal and right-aligned, and T_2 is a tree with all nodes of type 0 (and therefore normal). From Lemma 37 we have:

$$f = \widetilde{T_1} \star \widetilde{T_2}^{-1}$$

It follows that every element of $\mathcal{T}_k$ can be expressed as the product of elements of the form $\widetilde{T}$ and their inverses where T is normal. $\square$

Definition 35 (Elementary pair)**:** We define the *elementary k-tree* of kind n and level i, written $T_{n,i}$, to be the k-tree constructed by starting with the smallest spine with at least $i+2$ leaves and performing a Λ_n substitution for the i^{th} leaf. We define the *elementary pair* of type n and level i, written $[n]_i$ to be $\widetilde{T_{n,i}}$.



(a) $T_{2,1}$, $k = 2$

(b) $T_{0,1}$, $k = 1$

(c) $T_{2,4}$, $k = 3$

Figure 4.4: Three examples of elementary trees

Lemma 42: *If T is a right-aligned tree, and T' is obtained from T by a Λ_n substitution of the i^{th} leaf (zero indexed) of T which is not the right-most leaf then $\widetilde{T'} = \widetilde{T} \star [n]_i$.*

Proof. We will show that there exists a k-tree S such that $\widetilde{T} = [T', S]$ and $[n]_i = [S, \operatorname{sp}(T')]$. Our result will then follow by Lemma 34 from the computation:

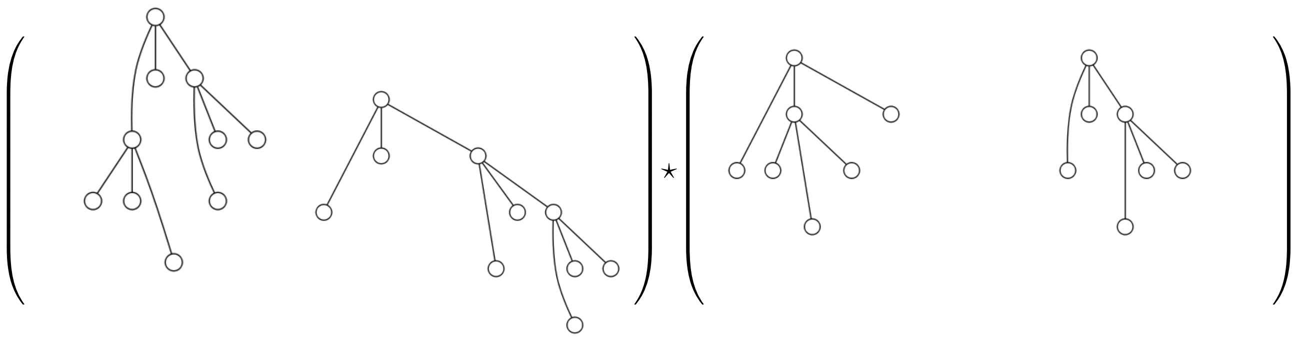
$$\widetilde{T} \star [n]_i = [T', S] \star [S, \operatorname{sp}(T')] = [T', \operatorname{sp}(T')] = \widetilde{T'}$$

We define the k-tree S to be the k-tree obtained by a Λ_n substitution of the i^{th} leaf of $\operatorname{sp}(T)$. We have $[T', S] = \widetilde{T}$ by Lemma **??**. Because the i^{th} leaf of T is not the right-most leaf, it must also be true that the i^{th} leaf of $\operatorname{sp}(T)$ is not the right-most leaf. We can therefore obtain S as a refinement of $T_{n,i}$ by repeated Λ_0 substitutions at the right-most leaf. Performing the corresponding substitutions on $\operatorname{sp}(T_{n,i})$ gives another spine, $\operatorname{sp}(S)$. Because S has the same number of leaves as T', $\operatorname{sp}(S) = \operatorname{sp}(T')$. We thus have $[n]_i = [S, \operatorname{sp}(T')]$. $\square$

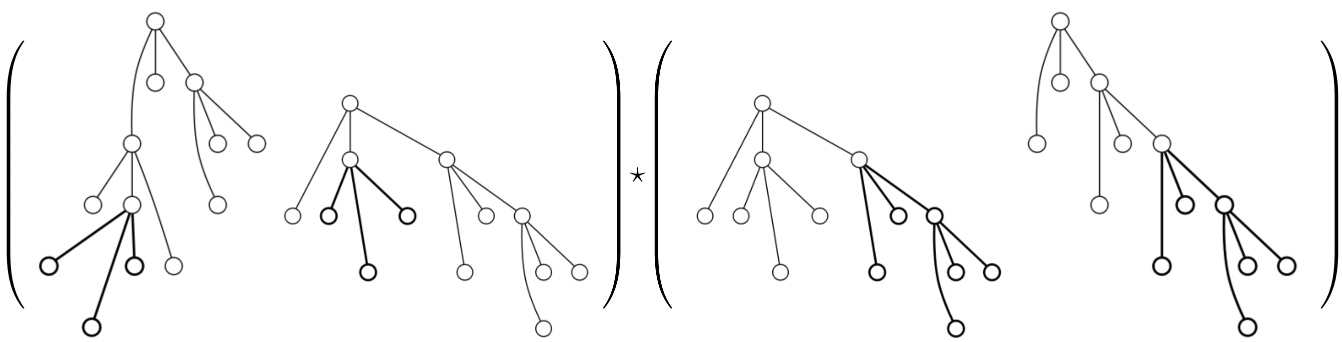
The construction of S as a refinement of $\operatorname{sp}(T)$ and $T_{n,i}$ is shown pictorially in Figure 4.5 for a simple choice of T in the case $k = 2$, $n, i = 1$.

Corollary 7: *For any right-aligned normal k-tree T there exists a collection of elementary pairs of type 0 or 1, $\{x_1, \ldots, x_n\}$, such that $\widetilde{T} = x_1 \star \ldots \star x_n$.*

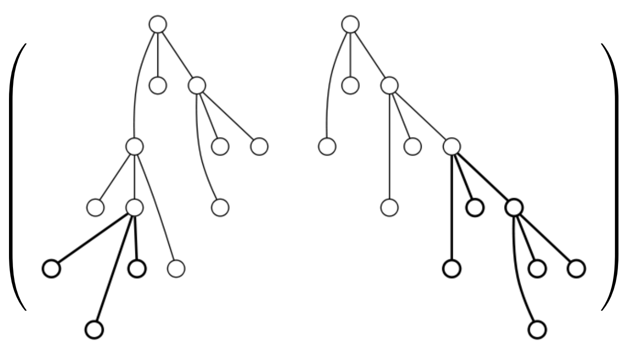
Proof. Any right-aligned normal tree T can be constructed by starting with the spine S that is its right edge, and then performing Λ_0 and Λ_1 substitutions of leaves other than the right-most leaf. From Lemma 42 it follows that there exists elementary pairs of types 0 and 1:



(a) $\widetilde{T} \star [1]_1$



(b) $(T', S) \star (S, \mathrm{sp}(T'))$



(c) $\widetilde{T'}$

Figure 4.5: The effect of right multiplication by an elementary pair

$x_1, \ldots, x_n$, such that:

$$\widetilde{T} = \widetilde{S} \star x_1 \star \ldots \star x_n = x_1 \star \ldots x_n$$

recalling that $\widetilde{S} = \mathrm{id}_{\mathcal{T}_k}$ by Lemma 37. $\square$

Corollary 8: $\Theta = \{[n]_i \,|\, 0 \leq n \leq 1,\ i \in \mathbb{Z}_{\geq 0}\}$ *is a generating set for* $\mathcal{T}_k$

Proof. From Corollary 6, $\mathcal{T}_k$ is generated by elements $\widetilde{T}$ where T is normal, and by Corollary 7 each such $\widetilde{T}$ is the product of elementary pairs of types 0 and 1. The set of such pairs is Θ. $\square$

Corollary 9: *Let* n_i *denote* $\widetilde{H}([n]_i)$. *Then* $G := \{n_i \,|\, 0 \leq n \leq 1,\ i \geq 0\}$ *generates* $\mathcal{F}_{\tau_k}$.

Proof. This follows from the fact that $\widetilde{H} : \mathcal{T}_k \to \mathcal{F}_{\tau_k}$ is a group isomorphism. $\square$

5. Presentations

5.1 An infinite presentation for the $\mathcal{F}_{\tau_k}$ groups

We now have a generating set for the group $\mathcal{F}_{\tau_k}$. This is not a minimal generating set; we will see later that we can can construct a generating set for each $\mathcal{F}_{\tau_k}$ with $2k+2$ elements. The generating set G is important because we can describe a fairly simple infinite presentation for $\mathcal{F}_{\tau_k}$ over G, analogously to the presentation given for F in, for example, [Bur18; CFP96; BS14], and for the group $\mathcal{F}_{\tau_1}$ in [BNR18]. When discussing words on the generators in G it will be useful to distinguish between equality of words and equality of the group elements represented by a word. We will use $u \equiv v$ to indicate that u and v are the same words, and $u = v$ will indicate merely that they represent the same element of $\mathcal{F}_{\tau_k}$.

5.1.1 Relations on the generators

There are two types of relations we need to consider. The first class of relations will be familiar from the study of the groups F_k and corresponds to the fact that we can construct the same k-tree in multiple ways, depending on the order in which we add the nodes. The second class of relations comes from the fact that different k-trees can have the same leaf sequence, and therefore represent the same subdivision. Each relation in the second class will correspond to a graft operation.

Let Π be the free group generated by the set $\mathcal{G} = \{\mathbf{n}_i \,|\, n \in \{0,1\},\ i \geq 0\}$. Let $\phi : \Pi \to \mathcal{T}_k$ be the surjective group homomorphism extending the map $\mathbf{n}_i \mapsto [n]_i$.

Relations of the first kind

Our first class of relations is of the form:

$$\mathbf{n}_j\mathbf{m}_i = \mathbf{m}_i\mathbf{n}_{j+k} \qquad \forall n, m \in \{0,1\}, i < j$$

Let $R_1 = \left\{\mathbf{n}_j\mathbf{m}_i\mathbf{n}_{j+k}^{-1}\mathbf{m}_i^{-1} \,|\, n, m \in \{0,1\},\ i < j\right\}$

Relations of the second kind

Our second class of relations is of the form:

$$\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2 \qquad \forall i \in \mathbb{Z}_{\geq 0}$$

Let $R_2 = \left\{\mathbf{0}_i\mathbf{0}_{i+1}\mathbf{1}_i^{-2} \,|\, i \geq 0\right\}$

Lemma 43: $R_1 \subseteq \ker(\phi)$

Proof. Assume $n, m \in \{0, \ldots, k\}$ and $i < j$. Then:

$$f := \phi\big(\mathbf{n}_j\mathbf{m}_i\mathbf{n}_{j+k}^{-1}\mathbf{m}_i^{-1}\big) = [n]_j \star [m]_i \star [n]_{j+k}^{-1} \star [m]_i^{-1}$$

Let S be a spine with more than $j+1$ leaf nodes. Then because $\widetilde{S} = 1_{\mathcal{T}_k}$ (Lemma 37) we have:

$$f = \widetilde{S} \star [n]_j \star [m]_i \star {[n]_{j+k}}^{-1} \star {[m]_i}^{-1} \star \widetilde{S}^{-1} = \left(\widetilde{S} \star [n]_j \star [m]_i\right) \star \left(\widetilde{S} \star [m]_i \star [n]_{j+k}\right)^{-1}$$

By Lemma 42, $\widetilde{S} \star [n]_j \star [m]_i = \widetilde{T_1}$ where T_1 is the tree obtained from S by first adding a node of type n to the j^{th} leaf, to obtain a new tree T_1' and then adding a node of type m to the i^{th} leaf node of T_1'.

Similarly, $\widetilde{S} \star [m]_i \star [n]_{j+k} = \widetilde{T_2}$ where T_2 is the tree obtained from adding a node of type m to the i^{th} node, obtaining T_2', and then adding a node of type n to the $j+k^{\text{th}}$ node of T_2'.

Upon inspection, $T_1 = T_2$, because the $j+k^{\text{th}}$ node of T_2' was the j^{th} node of S. We have simply changed the order in which we added the nodes. A visual depiction of this is given in Figure 5.1. It follows that:

$$f = \widetilde{T_1} \star {\widetilde{T_1}}^{-1} = \mathrm{id}_{\mathcal{T}_k}$$

Thus each element of R_1 is in the kernel of ϕ. □

Lemma 44: $R_2 \subseteq \ker(\phi)$

Proof. This proof is similar to the proof for Lemma 43, but relies on the fact that performing a graft operation does not change the leaf sequence of a k-tree.

Assume $i \in \mathbb{Z}_{\geq 0}$. Then:

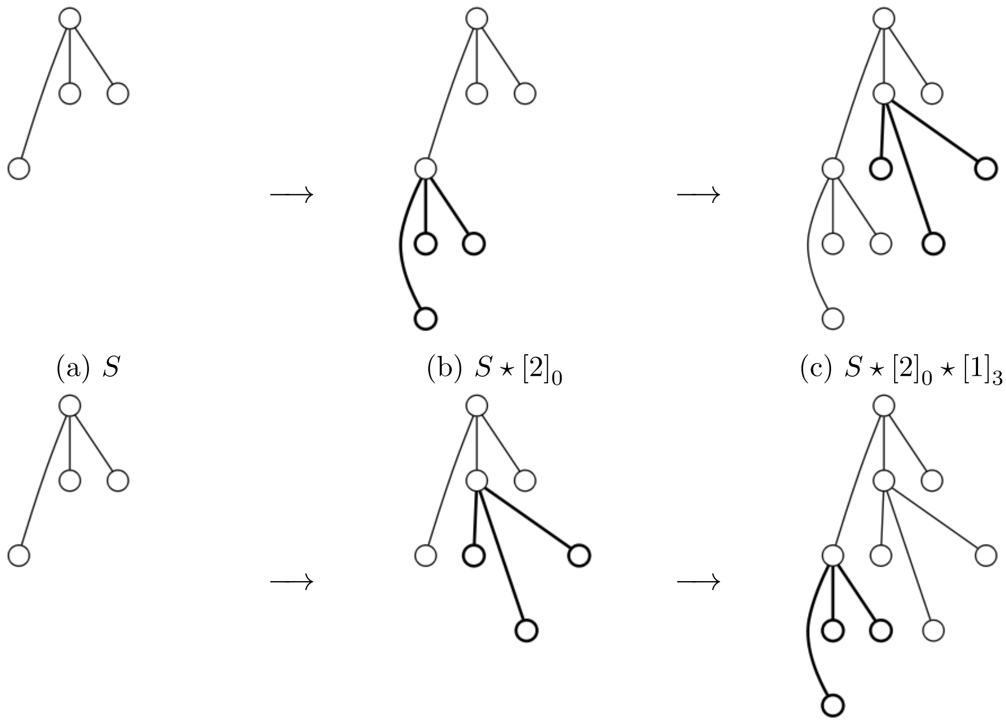
$$f := \phi\big(\mathbf{0}_i\mathbf{0}_{i+1}\mathbf{1}_i^{-2}\big) = \; [0]_i \star [0]_{i+1} \star {[1]_i}^{-2}$$

Let S be a spine with more than $j+1$ leaf nodes. Then:

$$\begin{aligned} f =& \widetilde{S} \star [0]_i \star [0]_{i+1} \star {[1]_i}^{-2} \star \widetilde{S}^{-1} \\ =& \left(\widetilde{S} \star [0]_i \star [0]_{i+1}\right)\left(\widetilde{S} \star {[1]_i}^2\right)^{-1} \end{aligned}$$

By Lemma 42, $\widetilde{S} \star [0]_i \star [0]_{i+1} = \widetilde{T_1}$ where T_1 is the tree obtained from S by first adding a node of type 0 to the i^{th} leaf, N, to obtain a new tree T_1' and then adding a node of type 0 to the $(i+1)^{\text{th}}$ leaf node of T_1'. Note that the $(i+1)^{\text{th}}$ leaf node of T_1' is $N(1)$, which is the leaf node to the right of the long child of N.

Similarly, $\widetilde{S} \star {[1]_i}^2 = \widetilde{T_2}$ where T_2 is the tree obtained from adding a node of type (1) to the i^{th} node, N, obtaining T_2', and then again adding a node of type (1) to the $(i)^{\text{th}}$ leaf node of



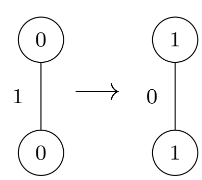
(a) S (b) $S \star [2]_0$ (c) $S \star [2]_0 \star [1]_3$

(d) S (e) $S \star [1]_1$ (f) $S \star [1]_1 \star [2]_0$

Figure 5.1: The relations in R_1 hold in $\mathcal{T}_k$. In this example, we have $n = 1$, $k = 2$, $m = 0$, $i = 0$ and $j = 1$. Note that the spines that make up the tree pairs are not displayed.

T_2'. Note that the i^{th} leaf node of T_2' is $N(0)$ which is the leaf node to the left of the long child of N. This is illustrated in Figure 5.2.

In this case, T_1 and T_2 will not be the same k-tree, but we can obtain T_2 from T_1 by performing a right graft at the node N, and so $T_1 \sim T_2$:



It follows that:

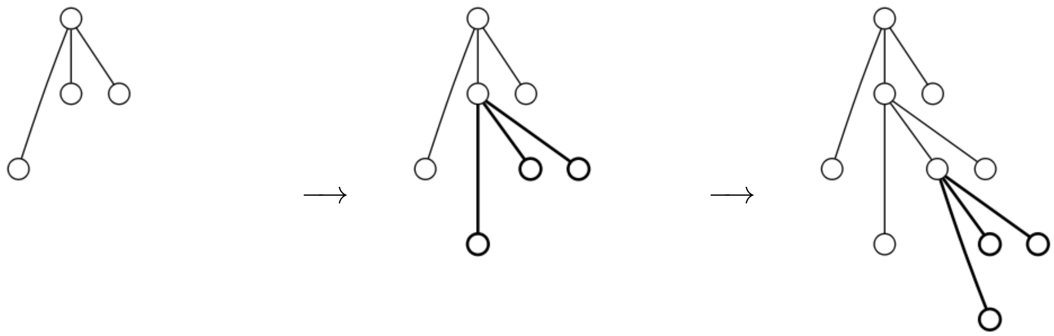
$$f = \widetilde{T_1} \star \widetilde{T_2}^{-1} = \widetilde{T_1} \star \widetilde{T_1}^{-1} = \mathrm{id}_{\mathcal{T}_k}$$

$\square$

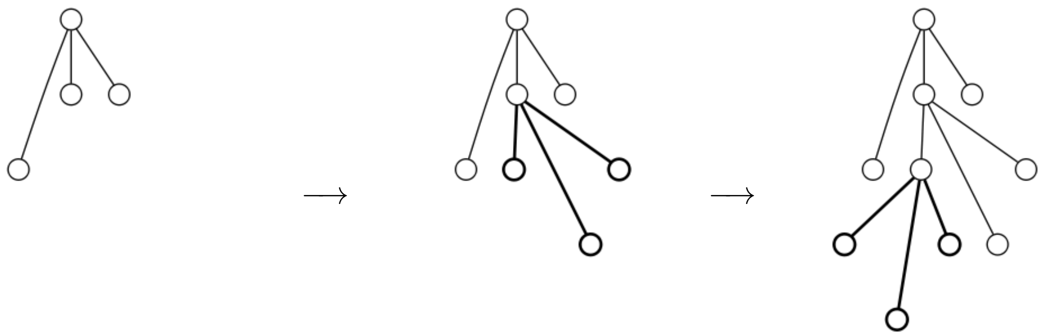
By Lemmas 43 and 44 there therefore exists a unique, surjective group homomorphism:

$$\psi : \frac{\Pi}{\langle\langle R_1 \cup R_2 \rangle\rangle} \to \mathcal{T}_k$$

such that $\psi \circ p = \phi$ where p is the quotient map $\Pi \to \frac{\Pi}{\langle\langle R_1 \cup R_2 \rangle\rangle}$. For notational simplicity we will make the definition: $\Gamma := \frac{\Pi}{\langle\langle R_1 \cup R_2 \rangle\rangle}$.



(a) S (b) $S \star [0]_1$ (c) $S \star [0]_1 \star [0]_2$



(d) S (e) $S \star [1]_1$ (f) $S \star [1]_1 \star [1]_1$

Figure 5.2: The relations in R_2 hold in $\mathcal{T}_k$. In this example, we have $i = 1$. The two k trees produced are different but leaf equivalent.

Our next goal is to show that ψ is injective, and therefore an isomorphism. Once this is established we will have an infinite presentation for $\mathcal{T}_k$, and therefore $\mathcal{F}_{\tau_k}$.

We will first make some definitions and establish some technical results.

Definition 36 (Positive word)**:** A *positive word* w is a word in the generators $\mathbf{n}_i \in \mathcal{G}$, and not in their inverses.

For example, $\mathbf{0}_2^2\mathbf{1}_0$ is a positive word, but $\mathbf{0}_2^2\mathbf{1}_0^{-1}$ is not.

Definition 37 (Type and level of a generator)**:** We define the *type* of a generator $\mathbf{n}_i$ to be n, and its *level* to be i. We will define the level of a word w to be the minimum level of the generators in w. We will use the notation $\mathcal{L}(x)$ to denote the level of a generator or word x. We will adopt the convention that the empty word has level ∞.

Lemma 45: *For any $x \in \Gamma$ there exist positive words u and v such that $x = uv^{-1}$*

Proof. We know that there exists some word w over the generators of Γ such that $x = w$. It suffices to show that if an inverse of one of the generators $\{\mathbf{n}_i \,|\, n \in \{0, \ldots, k\},\ i \geq 0\}$ occurs to the left of one of the positive generators in w then we can rewrite this pair as a positive generator to the left of an inverse, or remove this pair.

Assume $\mathbf{n}_i^{-1}\mathbf{m}_j$ is a subword of w. We will demonstrate the result in five exhaustive cases:

1. $i < j$

2. $i > j$
3. $i = j$ and
   (a) $m = n$
   (b) $m = 1$, $n = 0$
   (c) $m = 0$, $n = 1$

Case 1: $i < j$

$$\begin{aligned}
\mathbf{n}_i^{-1}\mathbf{m}_j =& \mathbf{n}_i^{-1}\mathbf{m}_j\mathbf{n}_i\mathbf{n}_i^{-1} \\
=& \mathbf{n}_i^{-1}\mathbf{n}_i\mathbf{m}_{j+k}\mathbf{n}_i^{-1} && \text{(Using a relation of the first kind)} \\
=& \mathbf{m}_{j+k}\mathbf{n}_i^{-1}
\end{aligned}$$

Case 2: $i > j$

$$\begin{aligned}
\mathbf{n}_i^{-1}\mathbf{m}_j =& \mathbf{n}_i^{-1}\mathbf{m}_j\mathbf{n}_{i+k}\mathbf{n}_{i+k}^{-1} \\
=& \mathbf{n}_i^{-1}\mathbf{n}_i\mathbf{m}_j\mathbf{n}_{i+k}^{-1} && \text{(Using a relation of the first kind)} \\
=& \mathbf{m}_j\mathbf{n}_{i+k}^{-1}
\end{aligned}$$

Case 3a: $i = j$**,** $m = n$

$$\mathbf{n}_i^{-1}\mathbf{m}_i = \mathbf{n}_i^{-1}\mathbf{n}_i = \mathrm{id}_\Gamma$$

and we can therefore remove this subword.

Case 3b: $i = j$**,** $m = 1$**,** $n = 0$ We can achieve our result by the relation of the second kind $r = \mathbf{0}_i\mathbf{0}_{i+1}\mathbf{1}_i^{-2}$:

$$\mathbf{0}_i^{-1}\mathbf{1}_i = \mathbf{0}_i^{-1}r\mathbf{1}_i = \mathbf{0}_{i+1}\mathbf{1}_i^{-1}$$

Case 3c: $i = j$**,** $m = 0$**,** $n = 1$

$$\mathbf{1}_i^{-1}\mathbf{0}_i = \left(\mathbf{0}_i^{-1}\mathbf{1}_i\right)^{-1} = \left(\mathbf{0}_{i+1}\mathbf{1}_i\right)^{-1} = \mathbf{1}_i\mathbf{0}_{i+1}^{-1}$$

We can therefore continue to shift each instance of an inverse of a generator towards the end of the word representing x, giving us a word of the desired form. $\square$

As an example of the above process, consider how the word $\mathbf{0}_1^{-1}\mathbf{1}_1\mathbf{0}_2\mathbf{1}_0$ can be converted to a word of the form uv^{-1}:

$$\mathbf{0}_1^{-1}\mathbf{1}_1\mathbf{0}_2\mathbf{1}_0 = \mathbf{0}_2\mathbf{1}_1^{-1}\mathbf{0}_2\mathbf{1}_0 = \mathbf{0}_2\mathbf{0}_{2+k}\mathbf{1}_1^{-1}\mathbf{1}_0 = \mathbf{0}_2\mathbf{0}_{2+k}\mathbf{1}_0\mathbf{1}_{1+k}^{-1}$$

For the proof of the injectivity of ψ it will be convenient to construct a function from positive words in $\mathcal{G}^*$ to right-aligned k-trees. We will change our notation slightly to distinguish between words as elements of $\mathcal{G}^*$ and words as elements of Γ. Henceforth, if $w \in \mathcal{G}^*$ then we will denote its image in Γ by $\widehat{w}$. In later section where it is clear from the context whether a word w is thought of as an element of $\mathcal{G}^*$ or Γ this notation may not be used.

If $w = \mathbf{n_1}_{i_1} \dots \mathbf{n_t}_{i_t} \in \mathcal{G}^*$, then we define a function $\mathcal{T}(\cdot) : \mathcal{G}^* \to k$TreePair by letting $\mathcal{T}(w)$ be the right aligned k-tree constructed by choosing a minimal spine S, and then attaching to S first a node of type n_1 to the $i_1{}^{\text{th}}$ leaf, then attaching a node of type n_2 to the $i_2{}^{\text{th}}$ leaf and so on. We retrospectively define S as minimal if it is the spine of minimal length such that none of the nodes are attached to the rightmost leaf in this process. Note that with this definition, $\mathcal{T}(w)$ will always be a reduced right-aligned k-tree. If it weren't, we could have picked a smaller spine.

Lemma 46: *For any word $w \in \mathcal{G}^*$, $\psi(\widehat{w}) = \widetilde{\mathcal{T}(w)}$*

Proof. Let S be the minimal spine used to construct $\mathcal{T}(w)$. We will denote the letters of w by $w = \mathbf{n_1}_{i_1} \dots \mathbf{n_t}_{i_t}$. Then:

$$\psi(\widehat{w}) = [n_1]_{i_1} \star \dots \star [n_t]_{i_t} = S \star [n_1]_{i_1} \star \dots \star [n_t]_{i_t} = \widetilde{\mathcal{T}(w)}$$

by Lemma 42. □

Lemma 47: *If $u, v \in \mathcal{G}^*$ satisfy $\mathcal{T}(u) = \mathcal{T}(v)$, then $\widehat{u} = \widehat{v}$.*

Proof. We will prove this by induction on the length of u, noting that the result is trivial in the case where u has length 0; if u has length 0 then $\mathcal{T}(u)$ and $\mathcal{T}(v)$ are both empty, so $v = u$ and thus $\widehat{v} = \widehat{u}$. Now assume the result holds for u with length less than $d \in \mathbb{Z}_{>1}$, and assume $u, v \in \mathcal{G}^*$ with u of length d and $\mathcal{T}(u) = \mathcal{T}(v)$.

Let $\mathbf{n}_i$ be the last letter of u.

Claim: *There exists a word $v' \in \mathcal{G}^*$ such that $\widehat{v} = \widehat{v'\mathbf{n}_i}$ and $\mathcal{T}(v'\mathbf{n}_i) = \mathcal{T}(v) = \mathcal{T}(u)$.*

Proof. Because $\mathcal{T}(u) = \mathcal{T}(v)$, we know there exists a letter in v that represents adding the node N corresponding to $\mathbf{n}_i$ in u. Let $\mathbf{n}_j$ denote this letter in the word v.

We will prove the result by induction on the number of letters p between $\mathbf{n}_j$ and the end of the word v. If $p = 0$, then $v = v'\mathbf{n}_j$. Because $\mathbf{n}_j$ corresponds to adding the node N, it must be the case that $j = i$, and so the result holds. Now assume the result holds when $p < s$ for some $s \in \mathbb{Z}_{>0}$ and assume $\mathbf{n}_j$ is distance s from the end of v. Let $\mathbf{m}_r$ denote the generator following $\mathbf{n}_j$ in v. The level r of this generator cannot be in $\{j, j+1, \dots, j+k\}$ because this would correspond to adding a node to a child of N, and by the fact that N can be added last in the tree $\mathcal{T}(u) = \mathcal{T}(v)$, we conclude that all the children of N are leaves. We therefore have two cases: $r > j + k$ or $r < j$.

Case 1: $r \geq j + k + 1$

In this case we can use the relation $\mathbf{m}_{r-k}\mathbf{n}_j = \mathbf{n}_j\mathbf{m}_r \in R_1$ to replace the subword $\mathbf{n}_j\mathbf{m}_r$ of v with $\mathbf{m}_{r-k}\mathbf{n}_j$ giving us a new word $y \in \mathcal{G}^*$ with $\widehat{y} = \widehat{v}$. By the argument given in Lemma 43 we note that $\mathcal{T}(y) = \mathcal{T}(v)$ and that the generator $\mathbf{n}_j$ in the word y represents the addition of the node N to the tree. The generator $\mathbf{n}_j$ in y is less than s letters away from the end of y, so the result follows by our induction hypothesis.

Case 2: $r < j$

The argument is similar to that for Case 1. Here we can use the relation $\mathbf{n}_j\mathbf{m}_r = \mathbf{m}_r\mathbf{n}_{j+k} \in R_1$ to replace the subword $\mathbf{n}_j\mathbf{m}_r$ of v with $\mathbf{m}_r\mathbf{n}_{j+k}$ giving us a new word

$y \in \mathcal{G}^*$ with $\widehat{y} = \widehat{v}$. By the argument given in Lemma 43 we note that $\mathcal{T}(y) = \mathcal{T}(v)$ and that the generator $\mathbf{n}_{j+k}$ represents the addition of N to the tree. The generator $\mathbf{n}_{j+k}$ in y is less than s letters away from the end of y, so the result follows by our induction hypothesis. $\square$

Now we have that $u = u'\mathbf{n}_i$ and $y = v'\mathbf{n}_i$ satisfy $\mathcal{T}(u'\mathbf{n}_i) = \mathcal{T}(v'\mathbf{n}_i)$ and $\widehat{y} = \widehat{v}$ and so $\mathcal{T}(u') = \mathcal{T}(v')$. This is simply the k-tree $\mathcal{T}(u)$ with the node N removed. The word u' has length less than d, so by our induction hypothesis it follows that $\widehat{u'} = \widehat{v'}$ in Γ, and that:

$$\widehat{u} = \widehat{u'\mathbf{n}_i} = \widehat{v'\mathbf{n}_i} = \widehat{v}$$

$\square$

Lemma 48: *If $u, v \in \mathcal{G}^*$ satisfy that $\mathcal{T}(u)$ is obtained from $\mathcal{T}(v)$ by a single graft operation, then $\widehat{u} = \widehat{v}$.*

Proof. We first note that because $\mathcal{T}(u)$ and $\mathcal{T}(v)$ are both trees with nodes only of types 0 and 1, the only graft operations that could possibly convert $\mathcal{T}(u)$ into $\mathcal{T}(v)$ are left grafts at a node N of type 1, where $N(0)$ is also of type 1, or right grafts at a node N of type 0 where $N(1)$ is also of type 0. Any other graft would produce nodes of types greater than 1. Because these are inverse transformations, it suffices only to check the case where $\mathcal{T}(u)$ is obtained from $\mathcal{T}(v)$ by right-graft, since otherwise $\mathcal{T}(v)$ is obtainable from $\mathcal{T}(u)$ by a right-graft, and we could swap u and v. Note also that the node at which the right graft occurs cannot be a node on the right edge, since this would produce a tree that is not right-aligned.

Assume that $u, v \in \mathcal{G}^*$ satisfy our condition. Then there exists a type 0 node N in $\mathcal{T}(u)$ such that $N(1)$ is also a type 0 node and performing a right graft at N obtains $\mathcal{T}(v)$. We could chose to construct the k-tree $\mathcal{T}(u)$ by adding the type zero node $N(1)$ directly after the node N, which would correspond to a word u' over $\mathcal{G}$ of the form $u_1\mathbf{0}_i\mathbf{0}_{i+1}u_2$, with $u_1, u_2 \in \mathcal{G}^*$. By Lemma 47, it follows that $\widehat{u} = \widehat{u'}$ since $\mathcal{T}(u) = \mathcal{T}(u')$ by construction.

Using a relation of the second kind, $\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2$, we obtain a new word:

$$y = u_1\mathbf{1}_i^2u_2$$

with $\widehat{u} = \widehat{u'} = \widehat{y}$.

From our argument in Lemma 44 we see that $\mathcal{T}(y)$ is the k-tree that is obtained from $\mathcal{T}(u)$ by performing a right graft operation at N, and therefore $\mathcal{T}(y) = \mathcal{T}(v)$. By use again of Lemma 47 we conclude that:

$$\widehat{u} = \widehat{u'} = \widehat{y} = \widehat{v}$$

$\square$

Lemma 49: *If $u, v \in \mathcal{G}^*$ satisfy $\mathcal{T}(u) \sim \mathcal{T}(v)$ then $\widehat{u} = \widehat{v}$.*

Proof. Because $\mathcal{T}(u)$ and $\mathcal{T}(v)$ are both trees with nodes only of types 0 and 1, the fact that $\mathcal{T}(u) \sim \mathcal{T}(v)$ implies that there exists a sequence of graft operations which never produces a node of type greater than 1 and which transforms $\mathcal{T}(u)$ into $\mathcal{T}(v)$, by Lemma **??**. Because grafting never needs to occur on the right edge, the output of each graft operation remains a reduced, right-aligned tree, and thus $\mathcal{T}(w)$ for some word w over $\mathcal{G}$. It follows that there is a sequence of words $u = w_0, w_1, \ldots, w_n = v$ such that each $\mathcal{T}(w_{i+1})$ is obtained from $\mathcal{T}(w_i)$ by a single graft operation. From Lemma 48, it follows that $\widehat{u} = \widehat{w_1} = \cdots = \widehat{w_n} = \widehat{v}$. □

Lemma 50: $\psi : \Gamma \to \mathcal{T}_k$ *is injective.*

Proof. Assume $x \in \ker(\psi)$. By Lemma 45 there exist (positive) words $u, v \in \mathcal{G}^*$, such that:

$$x = \widehat{u}\widehat{v}^{-1}$$

The condition that $x \in \ker(\psi)$ gives us that:

$$\psi(x) = \psi(\widehat{u}) \star \psi(\widehat{v})^{-1} = \mathrm{id}_{\mathcal{T}_k}$$

We now note the following:

1. By Lemma 46, $\widetilde{\mathcal{T}(u)} \star \widetilde{\mathcal{T}(v)} = \mathrm{id}_{\mathcal{T}_k}$
2. Then by Lemma 39, $\mathcal{T}(u) \sim \mathcal{T}(v)$
3. Then by Lemma 49, $\widehat{u} = \widehat{v}$

It follows that $x = \widehat{u}\widehat{v}^{-1} = \widehat{u}\widehat{u}^{-1} = \mathrm{id}_\Gamma$.

We conclude that ψ is injective. □

A presentation for $\mathcal{F}_{\tau_k}$ is therefore given by $\frac{\Pi}{\langle\langle R_1 \cup R_2 \rangle\rangle}$.

5.2 A Finite Presentation for $\mathcal{F}_{\tau_k}$

From our infinite presentation we can construct a finite presentation as has been done for the groups F_k in [BS14]. For notational convenience we will use $(z \downarrow w)$ to denote $w^{-1}zw$ in this section. We will also be making frequent use of the generator $\mathbf{0}_0$ of $\mathcal{F}_{\tau_k}$, so for this section only we will denote this generator by x.

We will make frequent use of the fact that $(z \downarrow w)$ defines a right action of w on z, and so in particular we have:

1. $(zy \downarrow w) = (z \downarrow w)(y \downarrow w)$
2. $(z \downarrow wy) = ((z \downarrow w) \downarrow y)$

Lemma 51: *For* $n, m \in \{0, 1\}$ *and* $0 \leq i < j$, $(\mathbf{n}_j \downarrow \mathbf{m}_i) = \mathbf{n}_{j+k}$

Proof. From the relation of the first kind $\mathbf{n}_j\mathbf{m}_i = \mathbf{m}_i\mathbf{n}_{j+k}$ we have:

$$(\mathbf{n}_j \downarrow \mathbf{m}_i) = \mathbf{m}_i^{-1}\mathbf{n}_j\mathbf{m}_i = \mathbf{m}_i^{-1}\mathbf{m}_i\mathbf{n}_{j+k} = \mathbf{n}_{j+k}$$

□

In particular, for any $n \in \{0, 1\}$ and $i > 0$, $(\mathbf{n}_i \downarrow x) = \mathbf{n}_{i+k}$.

We begin by defining a finite generating set for $\mathcal{F}_{\tau_k}$.

Lemma 52: *$\mathcal{F}_{\tau_k}$ is generated by the finite set:*

$$\Phi_k := \{\mathbf{0}_i, \mathbf{1}_i \,|\, 0 \leq i \leq k\}$$

Proof. It suffices to show that each generator in $\mathcal{G}$ can be expressed as a word in Φ_k. Clearly this is true for each generator in $\mathcal{G}$ of level less than or equal to k. It remains to consider generators of the form $\mathbf{n}_i$ for $i > k$. In this case we can write i uniquely as $\sigma(i) + \eta(i)k$ where $1 \leq \sigma(i) \leq k$ and $\eta(i) \in \mathbb{Z}_{\geq 1}$. From Lemma 51, we therefore have:

$$\mathbf{n}_i = \left(\mathbf{n}_{\sigma(i)} \downarrow x^{\eta(i)}\right)$$

which gives us our result. $\square$

By rewriting all of our relators in R_1 and R_2 on the generating set $\mathcal{G}$ in terms of the generators in Φ_k we will obtain another infinite presentation for $\mathcal{F}_{\tau_k}$. We will then show that we can replace these relators with a finite set of relators and thus obtain a finite presentation.

5.2.1 Removing Relations of the Second Kind

We will see that we can replace our relations of the the second kind with the finite set:

$$\mathcal{R}_2 := \left\{\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2 \,|\, 0 \leq i < k\right\} \cup \left\{\mathbf{0}_k x^{-1} \mathbf{0}_1 x = \mathbf{1}_k^2\right\}$$

Each relation of the second kind is of the form $\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2$. Using the fact that $\mathbf{n}_i = \left(\mathbf{n}_{\sigma(i)} \downarrow x^{\eta(i)}\right)$, we can rewrite each relation of the second kind purely in terms of generators in Φ_k as:

$$\left(\mathbf{0}_{\sigma(i)} \downarrow x^{\eta(i)}\right)\left(\mathbf{0}_{\sigma(i+1)} \downarrow x^{\eta(i+1)}\right) = \left(\mathbf{1}_{\sigma(i)}^2 \downarrow x^{\eta(i)}\right) \tag{‡}$$

For $i \neq nk$ for $n \in \mathbb{Z}_{\geq 1}$, $\eta(i) = \eta(i+1)$ and $\sigma(i+1) = \sigma(i) + 1$ so from (‡) we have:

$$\begin{aligned}
&\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2 \\
\iff &\left(\mathbf{0}_{\sigma(i)} \downarrow x^{\eta(i)}\right)\left(\mathbf{0}_{\sigma(i+1)} \downarrow x^{\eta(i+1)}\right) = \left(\mathbf{1}_{\sigma(i)}^2 \downarrow x^{\eta(i)}\right) \\
\iff &\left(\mathbf{0}_{\sigma(i)} \downarrow x^{\eta(i)}\right)\left(\mathbf{0}_{\sigma(i)+1} \downarrow x^{\eta(i)}\right) = \left(\mathbf{1}_{\sigma(i)}^2 \downarrow x^{\eta(i)}\right) \\
\iff &\left(\mathbf{0}_{\sigma(i)}\mathbf{0}_{\sigma(i)+1} \downarrow x^{\eta(i)}\right) = \left(\mathbf{1}_{\sigma(i)}^2 \downarrow x^{\eta(i)}\right) \\
\iff &\mathbf{0}_{\sigma(i)}\mathbf{0}_{\sigma(i)+1} = \mathbf{1}_{\sigma(i)}^2
\end{aligned}$$

and $\mathbf{0}_{\sigma(i)}\mathbf{0}_{\sigma(i+1)} = \mathbf{1}_{\sigma(i)}^2 \in \mathcal{R}_2$.

For $i = nk$ for $n \in \mathbb{Z}_{\geq 1}$, $\eta(i+1) = \eta(i) + 1$, $\sigma(i) = k$ and $\sigma(i+1) = 1$, so we have:

$$\begin{aligned}
&\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2 \\
\iff &\big(\mathbf{0}_{\sigma(i)} \downarrow x^{\eta(i)}\big)\big(\mathbf{0}_{\sigma(i+1)} \downarrow x^{\eta(i+1)}\big) = \big(\mathbf{1}_{\sigma(i)}^2 \downarrow x^{\eta(i)}\big) \\
\iff &\big(\mathbf{0}_k \downarrow x^{\eta(i)}\big)\big(\mathbf{0}_1 \downarrow x^{\eta(i)+1}\big) = \big(\mathbf{1}_k^2 \downarrow x^{\eta(i)}\big) \\
\iff &\big(\mathbf{0}_k \downarrow x^{\eta(i)}\big)\big(x^{-1}\mathbf{0}_1 x \downarrow x^{\eta(i)}\big) = \big(\mathbf{1}_k^2 \downarrow x^{\eta(i)}\big) \\
\iff &\big(\mathbf{0}_k x^{-1}\mathbf{0}_1 x \downarrow x^{\eta(i)}\big) = \big(\mathbf{1}_k^2 \downarrow x^{\eta(i)}\big) \\
\iff &\mathbf{0}_k x^{-1}\mathbf{0}_1 x = \mathbf{1}_k^2
\end{aligned}$$

and $\mathbf{0}_k x^{-1}\mathbf{0}_1 x = \mathbf{1}_k^2 \in \mathcal{R}_2$.

If follows that the relations of the second kind R_2 on the generating set $\mathcal{G}$ can be replaced by the relations in $\mathcal{R}_2$ on the generators in Φ_k.

5.2.2 Removing Relations of the First Kind

We now consider the relations of the first kind, all of the form $\mathbf{m}_j\mathbf{n}_i = \mathbf{n}_i\mathbf{m}_{j+k}$ for $i < j$. We can rewrite these in terms of generators in Φ_k as:

$$\big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)}\big)\big(\mathbf{m}_{\sigma(i)} \downarrow x^{\eta(i)}\big) = \big(\mathbf{m}_{\sigma(j)} \downarrow x^{\eta(i)}\big)\big(\mathbf{n}_{\sigma(j+k)} \downarrow x^{\eta(j+k)}\big)$$

Because $j \geq 1$ (since $j > i \geq 0$), $\sigma(j+k) = \sigma(j)$ and $\eta(j+k) = \eta(j) + 1$, so we have:

$$\begin{aligned}
&\big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)}\big)\big(\mathbf{m}_{\sigma(i)} \downarrow x^{\eta(i)}\big) = \big(\mathbf{m}_{\sigma(j)} \downarrow x^{\eta(i)}\big)\big(\mathbf{n}_{\sigma(j+k)} \downarrow x^{\eta(j+k)}\big) \\
\iff &\big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)}\big)\big(\mathbf{m}_{\sigma(i)} \downarrow x^{\eta(i)}\big) = \big(\mathbf{m}_{\sigma(j)} \downarrow x^{\eta(i)}\big)\big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)+1}\big) \\
\iff &\big(\mathbf{m}_{\sigma(j)} \downarrow x^{\eta(i)}\big)^{-1}\big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)}\big)\big(\mathbf{m}_{\sigma(i)} \downarrow x^{\eta(i)}\big) = \big(\mathbf{n}_{\sigma(j)} \downarrow x^{\eta(j)+1}\big) \\
\iff &x^{-\eta(i)}\mathbf{m}_{\sigma(j)}^{-1}x^{\eta(i)}x^{-\eta(j)}\mathbf{n}_{\sigma(j)}x^{\eta(j)}x^{-\eta(i)}\mathbf{m}_{\sigma(i)}x^{\eta(i)} = x^{-(\eta(j)+1)}\mathbf{n}_{\sigma(j)}x^{(\eta(j)+1)} \\
\iff &\mathbf{m}_{\sigma(j)}^{-1}x^{-(\eta(j)-\eta(i))}\mathbf{n}_{\sigma(j)}x^{(\eta(j)-\eta(i))}\mathbf{m}_{\sigma(i)} = x^{-(\eta(j)-\eta(i)+1)}\mathbf{n}_{\sigma(j)}x^{(\eta(j)-\eta(i)+1)} \\
\iff &\big(\mathbf{n}_{\sigma(j)} \downarrow x^s\mathbf{m}_{\sigma(i)}\big) = \big(\mathbf{n}_{\sigma(j)} \downarrow x^{s+1}\big)
\end{aligned}$$

where $s = \eta(j) - \eta(i)$. The condition that $j > i$ translates to the condition that $s > 0$ (i.e. $\eta(j) > \eta(i)$) or $\sigma(j) > \sigma(i)$.

We can therefore replace the relations from R_1 over $\mathcal{G}$ by the set of relations over Φ_k given by:

$$\mathcal{R}_1 := \big\{(\mathbf{n}_j \downarrow x^s\mathbf{m}_i) = \big(\mathbf{n}_j \downarrow x^{s+1}\big) \,|\, n, m \in \{0,1\},\ 0 \leq i, j \leq k,\ s > 0 \text{ or } j > i\big\}$$

We can partition this set of relations as $\mathcal{R}_1 = \bigsqcup_{s=0}^{\infty} \mathcal{R}_{1,s}$ where:

$$\begin{aligned}
\mathcal{R}_{1,0} &:= \{(\mathbf{n}_j \downarrow \mathbf{m}_i) = (\mathbf{n}_j \downarrow x) \,|\, n, m \in \{0,1\},\ 0 \leq i, j \leq k,\ j > i\} \\
\mathcal{R}_{1,s} &:= \big\{(\mathbf{n}_j \downarrow x^s\mathbf{m}_i) = \big(\mathbf{n}_j \downarrow x^{s+1}\big) \,|\, n, m \in \{0,1\},\ 0 \leq i, j \leq k\big\} \qquad (s > 0)
\end{aligned}$$

We will show that we only need a finite subset of these relations, given by the set:

$$\mathcal{R}_1' := \mathcal{R}_{1,0} \sqcup \mathcal{R}_{1,1} \sqcup \mathcal{R}_{1,2}$$

Lemma 53: *The relations in $\mathcal{R}_1'$ are equivalent to the relations in $\mathcal{R}_1$.*

Proof. We need to show that each relation in $\mathcal{R}_1$ is implied by the relations in $\mathcal{R}_1'$. We will do this by induction, showing that for $s \in \mathbb{Z}_{\geq 3}$, the relations in $\mathcal{R}_{1,s}$ are implied by those in $\mathcal{R}_{1,d}$ for $d < s$.

For any $s \in \mathbb{Z}_{\geq 3}$, an arbitrary relation in $\mathcal{R}_{1,s}$ is of the form:

$$(\mathbf{n}_j \downarrow x^s \mathbf{m}_i) = \left(\mathbf{n}_j \downarrow x^{s+1}\right)$$

We will show that this relation can be proven using relations in $\mathcal{R}_{1,d}$ for $d < s$. In fact, we will only need relations in $\mathcal{R}_{1,1} \cup \mathcal{R}_{1,s-1} \cup \mathcal{R}_{1,s-2}$

Before we begin, we will give another expression for $x^s \mathbf{m}_i$ which will be useful:

$$\begin{aligned} x^s \mathbf{m}_i &= x^2 \left(\mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right) \left(\mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right)^{-1} x^{s-2} \mathbf{m}_i \\ &= x^2 \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right) \end{aligned} \tag{**}$$

We know prove the necessary equality using relations in $\mathcal{R}_{1,1} \cup \mathcal{R}_{1,s-2} \cup \mathcal{R}_{1,s-1}$:

$$\begin{aligned}
& (\mathbf{n}_j \downarrow x^s \mathbf{m}_i) \\
&= \left(\mathbf{n}_j \downarrow x^2 \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) && \text{(by **)} \\
&= \left(\left(\mathbf{n}_j \downarrow x^2 \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right) \downarrow \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) \\
&= \left(\left(\left(\mathbf{n}_j \downarrow x^2\right) \downarrow \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right) \downarrow \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) \\
&= \left(\left((\mathbf{n}_j \downarrow x \mathbf{n}_j) \downarrow \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right) \downarrow \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) && \text{(from } \mathcal{R}_{1,1}\text{)} \\
&= \left(\left(\mathbf{n}_j \downarrow x \mathbf{n}_j \mathbf{n}_j^{-1} x^{s-2} \mathbf{m}_i\right) \downarrow \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) \\
&= \left(\left(\mathbf{n}_j \downarrow x^{s-1} \mathbf{m}_i\right) \downarrow \left(\mathbf{n}_j \downarrow x^{s-2} \mathbf{m}_i\right)\right) \\
&= \left((\mathbf{n}_j \downarrow x^s) \downarrow \left(\mathbf{n}_j \downarrow x^{s-1}\right)\right) && \text{(from } \mathcal{R}_{1,s-1} \text{ and } \mathcal{R}_{1,s-2}\text{)} \\
&= \left((\mathbf{n}_j \downarrow x^s) \downarrow x^{-(s-1)} \mathbf{n}_j x^{s-1}\right) \\
&= \left(\mathbf{n}_j \downarrow x \mathbf{n}_j x^{s-1}\right) \\
&= \left((\mathbf{n}_j \downarrow x \mathbf{n}_j) \downarrow x^{s-1}\right) \\
&= \left(\left(\mathbf{n}_j \downarrow x^2\right) \downarrow x^{s-1}\right) && \text{(from } \mathcal{R}_{1,1}\text{)} \\
&= \left(\mathbf{n}_j \downarrow x^{s+1}\right)
\end{aligned}$$

We conclude that the relations in $\mathcal{R}_1$ are equivalent to the relations in $\mathcal{R}_1'$. □

It follows that the finite set of relations $\mathcal{R}_1' \sqcup \mathcal{R}_2$ over the finite generating set Φ_k gives a finite presentation for $\mathcal{F}_{\tau_k}$.

6. Further Properties

6.1 Abelianisations of the groups $\mathcal{F}_{\tau_k}$

We can obtain a representation for the abelianisations $\mathcal{A}_{\tau_k}$ of the groups $\mathcal{F}_{\tau_k}$ by "abelianising" all of the relations in the finite presentation for $\mathcal{F}_{\tau_k}$. That is, we translate each relation into additive notation and assume that the group operation is commutative. For example, the relation $\mathbf{0}_0\mathbf{0}_1 = \mathbf{1}_0^2 \in \mathcal{R}_2$ becomes $\mathbf{0}_0 + \mathbf{0}_1 = 2\mathbf{1}_0$.

Under this translation, all of the relations in $\mathcal{R}_1'$ become trivial. This is because $(y \downarrow z) = y$ when the group operation is commutative. Every relation in $\mathcal{R}_1'$ is of the form $(y \downarrow z) = (y \downarrow w)$, so the translating this to a relation on $\mathcal{A}_{\tau_k}$ gives $y = y$, which is trivial.

We can therefore obtain a presentation for $\mathcal{A}_{\tau_k}$ by taking Φ_k as our generating set, and taking the translations of relations in $\mathcal{R}_2$ as the relations on Φ_k. We explicitly describe these relations as relators:

$$\overline{R_2} = \{\mathbf{0}_i + \mathbf{0}_{i+1} - 2\mathbf{1}_i \,|\, 0 \le i < k\} \cup \{\mathbf{0}_k + \mathbf{0}_1 - 2\mathbf{1}_k\}$$

This gives us a finite presentation for $\mathcal{A}_{\tau_k}$. With some work we can give a much simpler presentation with $k+2$ generators and one relator.

Lemma 54: *$\mathcal{A}_{\tau_k}$ is generated by the set*

$$G_2 := \{\mathbf{0}_k\} \cup \{\mathbf{1}_i \,|\, 0 \le i \le k\}$$

and the single relator:

$$r = 2\left(\sum_{s=1}^{k} (-1)^{k-s+1}\mathbf{1}_s\right) + \mathbf{0}_k + (-1)^{k-1}\mathbf{0}_k$$

over G_2 gives a presentation for $\mathcal{A}_{\tau_k}$.

Proof. We will consume k of our $k+1$ relators in $\overline{R_2}$ to eliminate k of the $2k+2$ generators from Φ_k, and then we will see that when we rewrite our last relator in $\overline{R_2}$ using generators in G_2 we get r.

We can label the relators in $\overline{R_2}$ as:

$$\begin{aligned} r_i =& \mathbf{0}_i + \mathbf{0}_{i+1} - 2\mathbf{1}_i \quad i < k \\ r_k =& \mathbf{0}_k + \mathbf{0}_1 - 2\mathbf{1}_k \end{aligned}$$

We first use r_{k-1} to replace the generator $\mathbf{0}_{k-1}$ with $2\mathbf{1}_{k-1} - \mathbf{0}_k$. In doing so, we can remove the relator r_{k-1} from our set of relators.

We can then use r_{k-2} to eliminate the generator $\mathbf{0}_{k-2}$:

$$\mathbf{0}_{k-2} = \mathbf{0}_{k-2} - r_{k-2} = 2\mathbf{1}_{k-2} - \mathbf{0}_{k-1} = 2\mathbf{1}_{k-2} - 2\mathbf{1}_{k-1} + \mathbf{0}_k$$

We can proceed by induction to eliminate the generators $\mathbf{0}_i$ using r_i for each $i < k$. We see that:

$$\mathbf{0}_i = 2\left(\sum_{s=i}^{k-1} (-1)^{k-s+1}\mathbf{1}_s\right) + (-1)^{k-i+1}\mathbf{0}_k$$

This leaves us with the generating set G_2. The only relator we have not exhausted from $\overline{R_2}$ is r_k. Translating r_k into the G_2 generators therefore gives our single relator over G_2:

$$\begin{aligned} r_k &= \mathbf{0}_k + \mathbf{0}_1 - 2\mathbf{1}_k \\ &= \mathbf{0}_k + \left(2\left(\sum_{s=1}^{k-1} (-1)^{k-s+1}\mathbf{1}_s\right) + (-1)^{k-1}\mathbf{0}_k\right) - 2\mathbf{1}_k \\ &= 2\left(\sum_{s=1}^{k} (-1)^{k-s+1}\mathbf{1}_s\right) + (-1)^{k-1}\mathbf{0}_k + \mathbf{0}_k \end{aligned}$$

□

Our final step is to adjust the generators slightly, to make the relator r simpler. First, let $\epsilon(k)$ be 1 if k is odd and 0 if k is even. Then, define:

$$x := \sum_{s=1}^{k} (-1)^{k-s+1}\mathbf{1}_s + \epsilon(k)\mathbf{0}_k$$

Lemma 55: *The relator $r' = 2x$ over the generators:*

$$G_2' := \{\mathbf{0}_k\} \cup \{\mathbf{1}_i \,|\, 0 \leq i \leq k-1\} \cup \{x\}$$

gives a presentation for $\mathcal{A}_{\tau_k}$.

Proof. The fact that G_2' is a generating set for $\mathcal{A}_{\tau_k}$ follows from the fact that we can recover the missing generator $\mathbf{1}_k$ from G_2 as:

$$\mathbf{1}_k = \sum_{s=1}^{k-1} (-1)^{k-s+1}\mathbf{1}_s + \epsilon(k)\mathbf{0}_k - x$$

Rewriting the relator r over G_2 in terms of the generators of G_2' gives us:

$$\begin{aligned}
r &= 2\left(\sum_{s=1}^{k}(-1)^{k-s+1}\mathbf{1}_s\right) + (-1)^{k-1}\mathbf{0}_k + \mathbf{0}_k \\
&= 2\left(\sum_{s=1}^{k}(-1)^{k-s+1}\mathbf{1}_s\right) + 2\epsilon(k)\mathbf{0}_k \\
&= -2\mathbf{1}_k + 2\left(\sum_{s=1}^{k-1}(-1)^{k-s+1}\mathbf{1}_s\right) + 2\epsilon(k)\mathbf{0}_k \\
&= -2\left(\sum_{s=1}^{k-1}(-1)^{k-s+1}\mathbf{1}_s + \epsilon(k)\mathbf{0}_k - x\right) + 2\left(\sum_{s=1}^{k-1}(-1)^{k-s+1}\mathbf{1}_s\right) + 2\epsilon(k)\mathbf{0}_k \\
&= 2x
\end{aligned}$$

We conclude that the relator $r' = 2x$ over G_2' gives a presentation for $\mathcal{A}_{\tau_k}$. □

Corollary 10: $\mathcal{A}_{\tau_k} \cong \mathbb{Z}_2 \oplus \mathbb{Z}^{k+1}$

Corollary 11: *For $n \neq m$, $\mathcal{F}_{\tau_n} \ncong \mathcal{F}_{\tau_m}$.*

6.2 A normal form for the groups $\mathcal{F}_{\tau_k}$

Normal forms for F_1 and $\mathcal{F}_{\tau_1}$ have been constructed in [Bur18] and [BNR18] in a way that can be naturally extended to the groups $\mathcal{F}_{\tau_k}$ for arbitrary k. Most of the work required to establish this normal form for $\mathcal{F}_{\tau_k}$ has already been accomplished in establishing that k-tree pair has a normal representative in Lemma 41. In [Bur18] and [BNR18], the normal forms are used to give estimates for the length of words in the group. Our main application of the normal form we develop will be to the construct embeddings from F_k to $\mathcal{F}_{\tau_k}$.

Definition 38 (Monotone Positive word)**:** A *monotone positive word* w is a positive word satisfying that if $\mathbf{n}_i$ appears before $\mathbf{m}_j$ in w, then $i \leq j$.

Definition 39 (D)**:** If w is a word over the generators of $\mathcal{F}_{\tau_k}$ and their inverses of level greater than or equal to k, then we define $(w)^{-}$ to be the word obtained by decreasing the level of all generators in w by k. Similarly we define $(w)^{--1}$ to be the the word obtained by increasing the level of all the generators in w by k.

Recall that we use the notation $\mathcal{L}(x)$ to indicate the level of a word or letter x.

Definition 40 (Seminormal form)**:** We will say that a word w in the generators $\mathcal{G}$ and their inverses is in *seminormal form* if it can written as $w = uv^{-1}$ where:

1. u and v are both monotone positive words.
2. All the generators in u are of type 0 or 1.

3. If a type one generator $\mathbf{1}_i$ occurs in u, the following generator in u must have level greater than i.

4. All the generators in v are of type 0

Proposition 2: *Each element $f \in \mathcal{F}_{\tau_k}$ can be represented as a word in seminormal form.*

Proof. From Lemma 41 we know that we have $f = \psi^{-1}([T_1, T_2]) = \psi^{-1}\left(\widetilde{T}_1 \star \widetilde{T}_2^{-1}\right)$ where:

1. T_1 and T_2 are right-aligned
2. T_1 only has nodes of type 1 and 0, and nodes of type 1 have a leaf as their left-most child
3. T_2 has only nodes of type 0

We can construct T_1 by starting with a right-aligned spine and adding nodes from left to right. First, we continue adding nodes of T_1 to the left-most leaf of the tree until we reach the left-most leaf of T_1. Then, to this tree, continue to add nodes of T_1 to the leaf second from the left (if any) and so on. In other words, we construct T_1 in such a way that we never add a node to the left of a node that we have already added. When constructed in this way, if a node is added to an i^{th} leaf after a node is added to an j^{th} leaf, then $i \geq j$. We can then recover a positive word u corresponding to this construction process such that $\widetilde{\mathcal{T}(u)} = \widetilde{T}_1$, and by the fact that we always added nodes from left to right, u is monotone. Moreover, because we never add a node to the left-most child of a type 1 node, any type 1 generator $\mathbf{1}_i$ in u can only be followed by a generator whose level is strictly greater than i. Similarly, constructing T_2 from left to right gives a corresponding word v such that $\widetilde{\mathcal{T}(v)} = \widetilde{T}_2$, and v is therefore a monotone positive word with all generators of type 0. It follows that:

$$f = \psi^{-1}\left(\widetilde{\mathcal{T}(u)} \star \widetilde{\mathcal{T}(v)}^{-1}\right) = \psi^{-1}\left(\psi\left(\widehat{u}\widehat{v}^{-1}\right)\right) = \widehat{u}\widehat{v}^{-1}$$

and the monotone positive words u and v satisfy the requirements of the seminormal form. $\square$

Definition 41 (Normal form)**:** We will say a word w over the generators of $\mathcal{F}_{\tau_k}$ is in *normal form* if it is in seminormal form and additionally satisfies:

1. If w contains a subword of the form $\mathbf{0}_i u \mathbf{0}_i^{-1}$ for some word u, then $\mathcal{L}(u) \leq i + k$.
2. If w contains a subword of the form $\mathbf{0}_i \mathbf{1}_i \mathbf{0}_{i+k+1} u \mathbf{0}_{i+1}^{-1} \mathbf{0}_i^{-1}$ then $\mathcal{L}(u) \leq i + 2k$

Before we give a proof for the existence of a normal form for each element of $\mathcal{F}_{\tau_k}$, we will establish a technical result:

Proposition 3: *If w is a monotone positive word in the generators of $\mathcal{F}_{\tau_k}$ of the form $w = \mathbf{n}_i u$ where $\mathcal{L}(u) \geq i + k + 1$, then the word $w' = (u)^{-}\mathbf{n}_i$ satisfies $\widehat{w'} = \widehat{w}$.*

Proof. We first note that if $j > i + k + 1$, and thus $i < j - k$ then for any $0 \leq m, n \leq k$ we get the following equality using a relator of the first kind:

$$\mathbf{n}_i \mathbf{m}_j = \left(\mathbf{m}_{j-k} \mathbf{n}_i \mathbf{m}_j^{-1} \mathbf{n}_i^{-1}\right) \mathbf{n}_i \mathbf{m}_j = \mathbf{m}_{j-k} \mathbf{n}_i$$

This means we can commute $\mathbf{n}_i$ with any of the generators to its right in w, decreasing the level of the generator in the process. Doing this for every generator in the word u gives us our result. $\square$

Corollary 12: *If w is a monotone positive word in the generators of $\mathcal{F}_{\tau_k}$ with $\mathcal{L}(w) = i+1$, then $\widehat{\mathbf{n}_i^{-1} w} = \widehat{(w)^{--1}} \mathbf{n}_i^{-1}$ for any $0 \leq n \leq k$.*

Proof. We have:

$$\mathbf{n}_i^{-1} w = \mathbf{n}_i^{-1} \left((w)^{--1} \right)^{-} \mathbf{n}_i \mathbf{n}_i^{-1} = \mathbf{n}_i^{-1} \mathbf{n}_i (w)^{--1} \mathbf{n}_i^{-1} = (w)^{--1} \mathbf{n}_i^{-1}$$

where each word is interpreted as an element of $\mathcal{F}_{\tau_k}$.

$\square$

Proposition 4: *Each element $f \in \mathcal{F}_{\tau_k}$ can be represented as a word in normal form*

Proof. We have already shown in Proposition 2 that any element $f \in \mathcal{F}_{\tau_k}$ can be represented by a word in seminormal form. We will show that if w is a seminormal form word for f that is not in normal form, then there exists another word w' that also represents f, is in seminormal form, and satisfies that the maximum level of generators in w' is strictly less than the maximum level of generators in w. It follows that f must be representable by a word in normal form.

Assume that w is a word in seminormal form representing f that is not in normal form. There are two cases, depending on which condition of the normal form w violates.

Case 1: w contains a subword of the form $\mathbf{0}_i u \mathbf{0}_i^{-1}$ with $\mathcal{L}(u) \geq i + k + 1$.

In this case, w must be of the form $w = (r_0 \mathbf{0}_i r_1)(l_0 \mathbf{0}_i l_1)^{-1}$ where r_0, r_1, l_0, l_1 are all monotone positive words, and $\mathcal{L}(r_1),\ \mathcal{L}(l_1) \geq i + k + 1$.

By Proposition 3 this element is also represented by the word:

$$w' = \left(r_0 (r_1)^{-} \mathbf{0}_i \right) \left(l_0 (l_1)^{-} \mathbf{0}_i \right)^{-1}$$

Cancelling the type zero generators at the end we get gives another word that represents the same element of $\mathcal{F}_{\tau_k}$:

$$w'' = \left(r_0 (r_1)^{-} \right) \left(l_0 (l_1)^{-} \right)^{-1}$$

w'' is again in seminormal form, and the maximum level of its generators is k less than the maximum level of the generators in w.

Case 2: w contains a subword of the form $\mathbf{0}_i \mathbf{1}_i \mathbf{0}_{i+k+1} u \mathbf{0}_{i+1}^{-1} \mathbf{0}_i^{-1}$ with $\mathcal{L}(u) \geq i + 2k + 1$.

In this case, w must be of the form $w = (r_0 \mathbf{0}_i \mathbf{1}_i \mathbf{0}_{i+k+1} r_1)(l_0 \mathbf{0}_i \mathbf{0}_{i+1} l_1)^{-1}$ where r_0, l_0, r_1 and l_1 are all monotone positive words, and $\mathcal{L}(r_1)$, $\mathcal{L}(l_1) \geq i + 2k + 1$.

Our first step is to use a relation of the first kind to commute $\mathbf{1}_i$ and $\mathbf{0}_{i+k+1}$, giving:

$$w_1 = (r_0 \mathbf{0}_i \mathbf{0}_{i+1} \mathbf{1}_i r_1)(l_0 \mathbf{0}_i \mathbf{0}_{i+1} l_1)$$

We can then use a relation of the second kind: $\mathbf{0}_i\mathbf{0}_{i+1} = \mathbf{1}_i^2$, to deduce that $\mathbf{0}_i\mathbf{0}_{i+1}\mathbf{1}_i = \mathbf{1}_i\mathbf{0}_i\mathbf{0}_{i+1}$, giving:

$$w_3 = (r_0\mathbf{1}_i\mathbf{0}_i\mathbf{0}_{i+1}r_1)(l_0\mathbf{0}_i\mathbf{0}_{i+1}l_1)$$

with $\widehat{w_3} = \widehat{w}$. Using Proposition 3 (twice) then gives us:

$$w_4 = \big(r_0\mathbf{1}_i(r_1)^{-2}\mathbf{0}_i\mathbf{0}_{i+1}\big)\big(l_0(l_1)^{-2}\mathbf{0}_i\mathbf{0}_{i+1}\big)$$

Finally, cancelling the type 0 generators at the ends gives us:

$$w_5 = \big(r_0\mathbf{1}_i(r_1)^{-2}\big)\big(l_0(l_1)^{-2}\big)$$

This is a new word in seminormal form representing the same element as w with maximum generator level $2k$ less than that of w. $\square$

Proposition 5: *Every element of $\mathcal{F}_{\tau_k}$ is represented by a unique normal form word over the generators.*

Proof. We have shown the existence of a normal form word for each element in Proposition 4, so all that remains to show is uniqueness.

To do so, we follow the approach taken in [Bur18] to produce a normal form for F and in [BNR18] to produce a normal form for $\mathcal{F}_{\tau_1}$.

We will prove uniqueness by contradiction. Let S denote the set of pairs of distinct normal form words $w_1, w_2 \in \mathcal{G}^*$ with $\widehat{w_1} = \widehat{w_2}$. If we assume that normal form representatives are not unique, then S is not empty. Define a function $\sigma : S \to \mathbb{Z}_{\geq 0}$ by letting $\sigma(w_1, w_2)$ be the sum of the lengths of the words w_1 and w_2. By the fact that S is non-empty, there exists some $(u, v) \in S$ with $\sigma(u, v)$ minimal. Moreover, we can pick the pair (u, v) such that the level of either u or v is zero. If we have such a (u, v) with neither $\mathcal{L}(u)$ nor $\mathcal{L}(v)$ equal to zero, then we can decrease the level of all generators in u and v to obtain another pair $(u', v') \in S$. These two words will again satisfy $\widehat{u'} = \widehat{v'}$ because the set of relations on $\mathcal{G}$ are closed under decreasing the levels of the generators. We will assume for notational simplicity that u is the word of level 0. Our two words u and v are therefore of the form:

$$u = (\mathbf{0}_0^{a_0}\mathbf{1}_0^{\epsilon_0}r)\big(\mathbf{0}_0^{b_0}l\big)^{-1} \qquad v = (\mathbf{0}_0^{c_0}\mathbf{1}_0^{\eta_0}\rho)\big(\mathbf{0}_0^{d_0}\lambda\big)^{-1}$$

where $\mathcal{L}(r), \mathcal{L}(l), \mathcal{L}(\rho), \mathcal{L}(\lambda) \geq 1$.

We can compute the gradient at zero of the group element represented by these words. We know that the interval represented by a leaf of a k-tree has length given by τ_k^i where i is the height of the leaf. From the definition of our map H from pairs of k-trees to $\mathcal{F}_{\tau_k}$ we know that the word $u = H\big(\mathcal{T}(\mathbf{0}_0^{a_0}\mathbf{1}_0^{\epsilon_0}r), \mathcal{T}\big(\mathbf{0}_0^{b_0}l\big)\big)$. The left-most leaf of $T_1 = \mathcal{T}(\mathbf{0}_0^{a_0}\mathbf{1}_0^{\epsilon_0}r)$ is hanging under a_0+1 type 0 nodes (the "+1" comes from the root node) and ϵ_0 type 1 nodes. Because the left-most child of a type 0 node has length 2, it follows that the height of this leaf node is given by $2(a_0+1)+\epsilon_0$. Similarly, the height of the left-most leaf of $T_2 = \mathcal{T}\big(\mathbf{0}_0^{b_0}l\big)$ is given by $2(b_0+1)$. The element f represented by u therefore maps the interval $\Big[\,0\ ,\ \tau_k^{2(a_0+1)+\epsilon_0}\,\Big]$ to the interval $\Big[\,0\ ,\ \tau_k^{2(b_0+1)}\,\Big]$ and thus has gradient $\tau_k^{2(b_0-a_0)-\epsilon_0}$. By the same argument, the gradient of the element of $\mathcal{F}_{\tau_k}$ represented by v is then $\tau_k^{2(d_0-c_0)-\eta_0}$. The condition that these words represent the same element of $\mathcal{F}_{\tau_k}$ gives us the equation:

$$2(b_0-a_0)-\epsilon_0=2(d_0-c_0)-\eta_0$$

We now observe that:

1. $\epsilon_0=\eta_0$: because $\epsilon_0=\eta+0\in\{0,1\}$. If $\epsilon=0$, then $2(b_0-a_0)-\epsilon_0=2(d_0-c_0)-\eta_0$ is even, and so $\eta_0=0$. Similarly, if $\epsilon_0=1$ then $2(b_0-a_0)-\epsilon_0=2(d_0-c_0)-\eta_0$ is odd, and so $\eta_0=1$.
2. $b_0-a_0=d_0-c_0$: by the fact that $\epsilon_0=\eta_0$.
3. At least one of a_0,b_0,c_0,d_0 is non-zero: if they were all zero, then by the fact that $\mathcal{L}(u)=0$, $\epsilon_0=1$ and so $\eta_0=1$. From this it follows that $(rl^{-1},\rho\lambda^{-1})\in S$ with $\sigma(rl^{-1},\rho\lambda^{-1})<\sigma(u,v)$, which contradicts the minimality of (u,v).
4. One of a_0,c_0 must be zero, and one of b_0,d_0 must be zero: otherwise we could cancel an $\mathbf{0}_0$ or an $\mathbf{0}_0^{-1}$ from the beginnings or ends of u,v to contradict the minimality of (u,v).
5. Assume without loss of generality that $c_0=0$. Then $d_0=0$: if b_0 were zero, then we would have $-a_0=d_0$. But a_0 and d_0 are both non-negative and cannot both be zero, so this is impossible. Since b_0 is not zero, it must be the case that $d_0=0$.
6. $a_0=b_0$: Since $a_0-b_0=c_0-d_0=0$.

With these observations we can rewrite our words u and v as:

$$u=(\mathbf{0}_0^{a_0}\mathbf{1}_0^{\epsilon_0}r)(\mathbf{0}_0^{a_0}l)^{-1}\qquad v=(\mathbf{1}_0^{\epsilon_0}\rho)(\lambda)^{-1}$$

We will obtain a contradiction by considering the two cases for the value of ϵ_0 independently.

Case 1: $\epsilon_0=0$

In this case we can write u and v as:

$$u=(\mathbf{0}_0^{a_0}r)(\mathbf{0}_0^{a_0}l)^{-1}\qquad v=(\rho)(\lambda)^{-1}$$

Conjugating both u and v by $\mathbf{0}_0^{-a_0}$ gives another pair of words $(u',v')\in S$ with $\sigma(u',v')=\sigma(u,v)$:

$$u'=rl^{-1},\qquad v'=\mathbf{0}_0^{-a_0}v\mathbf{0}_0^{a_0}=\left(\mathbf{0}_0^{-a_0}\rho\right)\left(\mathbf{0}_0^{-a_0}\lambda\right)^{-1}$$

Using Corollary 12 we can construct another normal form word v'' with $\widehat{v''}=\widehat{v}=\widehat{u}$. We will take $v''=(\rho)^{--a_0}\big((\lambda)^{--a_0}\big)^{-1}$ and note that, as elements of $\mathcal{F}_{\tau_k}$, we have:

$$\begin{aligned}v'=\left(\mathbf{0}_0^{-a_0}\rho\right)\left(\mathbf{0}_0^{-a_0}\lambda\right)^{-1}=&\big((\rho)^{--a_0}\mathbf{0}_0^{-a_0}\big)\big((\lambda)^{--a_0}\mathbf{0}_0^{-a_0}\big)^{-1}\\=&(\rho)^{--a_0}\big((\lambda)^{--a_0}\big)^{-1}\\=&v''\end{aligned}$$

It follows that the normal form words u' and v'' both represent the same element in $\mathcal{F}_{\tau_k}$ and have total length less than u and v. By the minimality of $(u,v) \in S$, it must be the case that $u' = v''$ (as words). But clearly $\mathcal{L}(v'') \geq 1 + a_0 k \geq 1 + k$, and so $\mathcal{L}(u') = \mathcal{L}(rl^{-1}) \geq 1 + k$, which violates condition 1 of the normal form for the word u. This gives us a contradiction.

Case 2: $\epsilon_0 = 1$

We proceed by a similar argument, we conjugate both u and v by $\mathbf{0}_0^{-a_0}$ to get two new words representing the same element:

$$u' = \mathbf{1}_0 rl, \qquad v' = \mathbf{0}_0^{-a_0} v \mathbf{0}_0^{a_0} = \left(\mathbf{0}_0^{-a_0}\mathbf{1}_0\rho\right)\left(\mathbf{0}_0^{-a_0}\lambda\right)^{-1}$$

We now make the observation that when we conjugate a word in normal form $w = \mathbf{1}_0 rl^{-1}$ where $\mathcal{L}(r), \mathcal{L}(l) \geq 1$ by $\mathbf{0}_0^{-1}$ we can represent the resulting element of the group $\mathcal{F}_{\tau_k}$ by the normal form word $w' \equiv \left(\mathbf{1}_0\mathbf{0}_{i+k}(r)^{-2}\right)\left(\mathbf{0}_1(l)^{-2}\right)^{-1}$ which is a word of the same length as $\mathbf{0}_0^{-1}w\mathbf{0}_0$. We do this by the following manipulation, where equality is interpreted as equality as elements of $\mathcal{F}_{\tau_k}$:

$$\begin{aligned}
\mathbf{0}_0^{-1}w\mathbf{0}_0 &= \left(\mathbf{0}_0^{-1}\mathbf{1}_0 r\right)\left(\mathbf{0}_0^{-1}l\right)^{-1} \\
&= \left(\mathbf{0}_0^{-1}\mathbf{1}_0\left(\mathbf{0}_0\mathbf{0}_1\mathbf{0}_1^{-1}\mathbf{0}_0^{-1}\right)r\right)\left(\mathbf{0}_0^{-1}\left(\mathbf{0}_0\mathbf{0}_1\mathbf{0}_1^{-1}\mathbf{0}_0^{-1}\right)l\right)^{-1} \\
&= \left(\mathbf{0}_0^{-1}\mathbf{1}_0\mathbf{0}_0\mathbf{0}_1(r)^{--2}\mathbf{0}_1^{-1}\mathbf{0}_0^{-1}\right)\left(\mathbf{0}_0^{-1}\mathbf{0}_0\mathbf{0}_1(l)^{--2}\mathbf{0}_1^{-1}\mathbf{0}_0^{-1}\right)^{-1} \\
&= \left(\mathbf{0}_0^{-1}\mathbf{1}_0\mathbf{0}_0\mathbf{0}_1(r)^{--2}\right)\left(\mathbf{0}_1(l)^{--2}\right)^{-1} \\
&= \left(\mathbf{0}_0^{-1}\mathbf{1}_0^3(r)^{--2}\right)\left(\mathbf{0}_1(l)^{--2}\right)^{-1} \\
&= \left(\mathbf{0}_0^{-1}\mathbf{0}_0\mathbf{0}_1\mathbf{1}_0(r)^{--2}\right)\left(\mathbf{0}_1(l)^{--2}\right)^{-1} \\
&= \left(\mathbf{0}_1\mathbf{1}_0(r)^{--2}\right)\left(\mathbf{0}_1(l)^{--2}\right)^{-1} \\
&= \left(\mathbf{1}_0\mathbf{0}_{1+k}(r)^{--2}\right)\left(\mathbf{0}_1(l)^{--2}\right)^{-1}
\end{aligned}$$

In particular, we have that $\widehat{\mathbf{0}_0^{-1}w\mathbf{0}_0} = \widehat{(\mathbf{1}_0\mathbf{0}_{1+k}r')(\mathbf{0}_1 l')^{-1}}$ can be expressed in the form $(\mathbf{1}_0\mathbf{0}_{1+k}r')(\mathbf{0}_1 l')^{-1}$ where $\mathcal{L}(r'), \mathcal{L}(l') \geq 2k+1$.

We can repeatedly apply this rule to the word v' (which is just v conjugated $a_0 \neq 0$ times by $\mathbf{0}_0^{-1}$) to see that v' has an expression as a normal word of the form:

$$v'' = (\mathbf{1}_0\mathbf{0}_{1+k}\rho')(\mathbf{0}_1\lambda')^{-1}$$

where the length of v'' is equal to that of v', and therefore $\sigma(u') = \sigma(v'')$. Because $\widehat{u'} = \widehat{v'} = \widehat{v''}$, we can cancel the $\mathbf{1}_0$ from the front of u' and v'' to obtain new words:

$$u''' = rl^{-1}, \qquad v''' = (\mathbf{0}_{1+k}\rho')\mathbf{0}_1\lambda'^{-1}$$

with $\widehat{u'''} = \widehat{v'''}$ and $\sigma(u''', v''') < \sigma(u,v)$. From the minimality of (u,v) it follows that $u''' = v'''$ as words. This means that the word u was of the form:

$$(\mathbf{0}_0^{a_0}\mathbf{1}_0\mathbf{0}_{1+k}\rho')(\mathbf{0}_0^{a_0}\mathbf{0}_1\lambda')^{-1}$$

where $\mathcal{L}(\rho')$, $\mathcal{L}(\lambda') \geq 2k+1$. The word u was therefore not in normal form, since it violates condition 2, so we get a contradiction.

We conclude that S is empty, and so there are no two distinct normal form words representing the same element of $\mathcal{F}_{\tau_k}$, and so any a normal form representation is unique. $\square$

7. Connections between $\mathcal{F}_{\mathcal{T}_k}$ and F_k

7.1 Embeddings and Amenability

We first state some facts about the groups F_k.

Lemma 56 ([BS14])**:** *Each group F_k has an infinite presentation given by:*

$$\left\langle x_i \ i \geq 0 \,|\, x_j x_i x_{j+k}^{-1} x_i^{-1} \ i < j \right\rangle$$

There are many proof of this result for the case F_1, for example in [Bur18] and [CFP96]. These proofs use representations of elements of Thompson's group by pairs of binary trees, and the result for F_k can be proven analogously considering instead pairs of $(k+1)$-ary trees.

Lemma 57: *The abelianisation of F_k is isomorphic to $\mathbb{Z}^{k+1}$*

There are many ways to show this. The most direct is to consider the finite presentations for the groups F_k given by Bieri and Strebel in [BS14] which consists of $k+1$ generators $\{x_0, \ldots, x_k\}$ and the relations:

$$\begin{aligned}
(x_j \downarrow x_i) &= (x_j \downarrow x_0) && (1 \leq i < j \leq p) \\
(x_j \downarrow x_0 x_i) &= \left(x_j \downarrow x_0^2\right) && (1 \leq i, j \leq p,\ j \leq i+1) \\
\left(x_1 \downarrow x_0^2 x_k\right) &= \left(x_1 \downarrow x_0^3\right) &&
\end{aligned}$$

These are expressed in terms of our conjugation notation developed in 5. All these relations become trivial when the commutator relations are added, so the abelianisation of F_k is isomorphic to the free abelian group on $k+1$ generators: $\mathbb{Z}^{k+1}$.

Lemma 58: *Each element f of F_k has a unique representation as a word w over the generators $\{x_i \ i \geq 0\}$ of F_k satisfying:*

1. *$w = uv^{-1}$, where u and v are monotone positive words*
2. *If $x_i w' x_i^{-1}$ is a subword of w, then w' has level less than or equal to $i+k$*

This follows from the proof given in [Bur18] for the normal form of Thompson's group.

Lemma 59: *If p divides k, then F_k contains a subgroup isomorphic to F_p.*

Proof. Assume $k = np$. Then we can define a group homomorphism $\phi : F_p \to F_k$ by mapping each generator x_i of F_2 to the generator $x_{ni} \in F_k$. To check that this is a well-defined homomorphism, we need to check that the image of each relator in F_1 is the identity in F_k. The image of a relator $x_j x_i x_{j+p}^{-1} x_i^{-1}$ in F_1 for $i < j$ under ϕ is:

$$\phi\big(x_j x_i x_{j+p}^{-1} x_i^{-1}\big) = x_{nj} x_{ni} x_{nj+np}^{-1} x_{ni}^{-1} = x_{nj} x_{ni} x_{nj+k}^{-1} x_{ni}^{-1}$$

which is a relator in F_k, and thus equal to the identity. To see that this homomorphism is injective, we use the fact that the map on the generators of F_p induces a map on words over the generators which sends normal form words in F_p to normal form words in F_k. If w is a monotone positive word in F_p, then $\phi(w)$ is clearly a monotone positive word in F_k, so if $w \equiv uv^{-1}$ is a normal form word in F_p, then $\phi(w) \equiv \phi(u)\phi(v)^{-1} \equiv \hat{u}\hat{v}^{-1}$ where $\hat{u}$ and $\hat{v}$ are monotone positive words. By the hypothesis that w is a normal form word in F_p, any subword $x_i w' x_i^{-1}$ of w for some $i \in \mathbb{Z}_{\geq 0}$ satisfies that $L(w') \leq i + p$. It follows that the word $\hat{w} = \phi(w)$ satisfies that for any subword $x_{ki} \hat{w}' x_{ki}^{-1}$, $L(\hat{w}') \leq ni + np = ni + k$, and $\hat{w}$ is therefore a normal form word in F_k.

The unique normal form word in F_k representing the identity is the empty word, so if $\phi(w) = \mathrm{id}_{F_k}$ for a normal form word w then w is the empty word in F_1, and therefore id_{F_p}.

It follows that $\phi : F_p \to F_k$ is injective, and therefore its image is a subgroup of F_k isomorphic to F_p. □

Lemma 60: *Each group $\mathcal{F}_{\tau_k}$ contains a subgroup isomorphic to F_k.*

Proof. We can again resort to an argument based on the normal forms for each group. We define a map $\xi : F_k \to \mathcal{F}_{\tau_k}$ by sending a generator x_i of F_k to the generator $\mathbf{0}_i$ of $\mathcal{F}_{\tau_k}$. This map is a well-defined homomorphism by the fact that each relator $x_j x_i x_{j+k}^{-1} x_i^{-1}$ on F_k is mapped by ξ to the relator $\mathbf{0}_j \mathbf{0}_i \mathbf{0}_{j+k}^{-1} \mathbf{0}_i^{-1}$ on $\mathcal{F}_{\tau_k}$. As was the case with our map ϕ, we see that the induced map on words over generators in F_k sends normal form words to normal form words over the generators in $\mathcal{F}_{\tau_k}$. As noted for ϕ, ξ clearly sends a word of the form uv^{-1} in F_k with u and v positive and monotone, to a word $\hat{w} = \hat{u}\hat{v}^{-1}$ over the type 0 generators of $\mathcal{F}_{\tau_k}$, where $\hat{u}$ and $\hat{v}$ are monotone and positive. Note that such a word $\hat{w}$ is therefore a word in seminormal form.

If we assume that w is a normal word in F_k, then it also satisfies the condition that is contains no subword of the form $x_i w' x_i^{-1}$, where $L(w') \leq i + k$. It follows that the image $\xi(w)$ is a word that contains no subword of the form $\mathbf{0}_i \hat{w}' \mathbf{0}_i^{-1}$ with $L(\hat{w}') \leq i + k$, which is precisely condition 1 from our definition of the normal form on $\mathcal{F}_{\tau_k}$. Condition 2 is automatically satisfied by the fact that the image $\hat{w}$ of w contains no generators of type 1.

Because the unique normal form for $\mathrm{id}_{\mathcal{F}_{\tau_k}}$ is the empty word, it follows that if $\xi(w) = \mathrm{id}_{\mathcal{F}_{\tau_k}}$, then w is the empty word, and so represents id_{F_k}. We conclude that $\xi : F_k \to \mathcal{F}_{\tau_k}$ is an injective group homomorphism, and therefore its image is a subgroup of $\mathcal{F}_{\tau_k}$ isomorphic to F_k. □

Corollary 13: *If p divides k, then $\mathcal{F}_{\tau_k}$ contains a subgroup isomorphic to F_p.*

Corollary 14: *Each group $\mathcal{F}_{\tau_k}$ contains a subgroup isomorphic to Thompson's group F_1.*

Theorem ([CFP96])**:** *Thompson's group F_1 is not elementary amenable.*

A proof for this result was given in [CFP96] based on previous work by [Cho+80].

Corollary 15: *$\mathcal{F}_{\tau_k}$ is not elementary amenable for any $k \in \mathbb{Z}_{\geq 1}$.*

Proof. This follows from the fact that subgroups of elementary amenable groups are elementary amenable (by the definition of elementary amenability). Thus, if $\mathcal{F}_{\tau_k}$ were elementary amenable, this would imply that F_1 is elementary amenable, which is false. □

Corollary 16: *If $\mathcal{F}_{\tau_k}$ is amenable for any $k \in \mathbb{Z}_{\geq 1}$, then so is F_1.*

Proof. This follows from the fact that subgroups of amenable (discrete) groups are amenable. A proof of this fact can be found in [Run04], Theorem 1.2.7. □

7.2 Unanswered Questions

One important property of Thompson's group F_1 that we have not investigated here is whether the commutator subgroup of each F_k is simple. This result was proven in [BNR18] for the group $\mathcal{F}_{\tau_1}$ by a method exploiting a property called transitivity which $\mathcal{F}_{\tau_1}$ shares with F_1 but not necessarily with $\mathcal{F}_{\tau_k}$ or F_k for $k > 1$.

Another question prompted by the embedding of F_k in $\mathcal{F}_{\tau_k}$ is whether $\mathcal{F}_{\tau_n}$ is embeddable in $\mathcal{F}_{\tau_m}$ of F_m for $n \neq m$. In particular, the question of whether $\mathcal{F}_{\tau_1}$ is embedded in F_1 has generated some interest and remains open.

Two natural extensions of this investigation into the groups $\mathcal{F}_{\tau_k}$ would be the investigation of the related groups $T(1, \mathbb{Z}[\tau_k], \langle \tau_k \rangle)$ and $G(1, \mathbb{Z}[\tau_k], \langle \tau_k \rangle)$ and an investigation into the groups $F(1, \mathbb{Z}[\tau], \langle \tau \rangle)$ for other algebraic integers.

Bibliography

[Bel07] James Belk. "Thompson's group F". In: *arXiv preprint arXiv:0708.3609* (2007).

[BNR18] José Burillo, Brita Nucinkis, and Lawrence Reeves. "An Irrational-slope Thompson's Group". In: *arXiv preprint arXiv:1806.00108* (2018).

[Bro87] Kenneth S. Brown. "Finiteness properties of groups". In: *Journal of Pure and Applied Algebra* 44.1 (Feb. 1, 1987), pp. 45–75. ISSN: 0022-4049. DOI: `10.1016/0022-4049(87)90015-6`. URL: `http://www.sciencedirect.com/science/article/pii/0022404987900156` (visited on 09/12/2017).

[BS14] Robert Bieri and Ralph Strebel. "On groups of PL-homeomorphisms of the real line". In: *arXiv preprint arXiv:1411.2868* (2014).

[BS85] Matthew G Brin and Craig C Squier. "Groups of piecewise linear homeomorphisms of the real line". In: *Inventiones mathematicae* 79.3 (1985), pp. 485–498.

[Bur18] José Burillo. *Introduction to Thompson's Group F*. 2018. URL: `https://mat-web.upc.edu/people/pep.burillo/book_en.php`.

[CFP96] James W Cannon, William J Floyd, and Walter R Parry. "Introductory notes on Richard Thompson's groups". In: *Enseignement Mathématique* 42 (1996), pp. 215–256.

[Cho+80] Ching Chou et al. "Elementary amenable groups". In: *Illinois Journal of Mathematics* 24.3 (1980), pp. 396–407.

[Cle00] Sean Cleary. "Regular subdivision in $\mathbf{Z}\left(\frac{1+\sqrt{5}}{2}\right)$". In: *Illinois Journal of Mathematics* 44.3 (2000), pp. 453–464.

[FL10] Marcelo Fiore and Tom Leinster. "An abstract characterization of Thompsons group F". In: *Semigroup Forum*. Vol. 80. 2. Springer. 2010, pp. 325–340.

[GG06] Ross Geoghegan and Fernando Guzmán. "Associativity and Thompson's group". In: *Contemporary Mathematics* 394 (2006), pp. 113–136.

[Gri98] Rostislav I Grigorchuk. "An example of a finitely presented amenable group not belonging to the class EG". In: *Sbornik: Mathematics* 189.1 (1998), p. 75.

[OS03] Alexander Yu Olshanskii and Mark V Sapir. "Non-amenable finitely presented torsion-by-cyclic groups". In: *Publications Mathématiques de l'Institut des Hautes Études Scientifiques* 96.1 (2003), pp. 43–169.

[Run04] Volker Runde. *Lectures on amenability*. Springer, 2004.

[Ste92] Melanie Stein. "Groups of Piecewise Linear Homeomorphisms". In: *Transactions of the American Mathematical Society* 332 (1992).