New York Journal of Mathematics

New York J. Math. **13** (2007) 437–481.

Thompson's group $F(n)$ is not minimally almost convex

Claire Wladis

Abstract. We prove that Thompson's group $F(n)$ is not minimally almost convex with respect to the standard finite generating set. A group G with Cayley graph Γ is not minimally almost convex if for arbitrarily large values of m there exist elements $g, h \in B_m$ such that $d_\Gamma(g,h) = 2$ and $d_{B_m}(g,h) = 2m$. (Here B_m is the ball of radius m centered at the identity.) We use tree-pair diagrams to represent elements of $F(n)$ and then use Fordham's metric to calculate geodesic length of elements of $F(n)$. Cleary and Taback have shown that $F(2)$ is not almost convex and Belk and Bux have shown that $F(2)$ is not minimally almost convex; we generalize these results to show that $F(n)$ is not minimally almost convex for all $n \in \{2, 3, 4, \dots\}$.

Contents

Received August 28, 2006.
Mathematics Subject Classification. 20F65.
Key words and phrases. Thompson's group, almost convexity.

ISSN 1076-9803/07

1. Introduction

1.1. Thompson's group $F(n)$. Thompson's group $F(n)$ is a generalization of the group F, which R. Thompson introduced in the early 1960's (see [14]). At that time Thompson invented three separate groups $F \subseteq T \subseteq V$, each of which is often referred to in the literature as Thompson's group; this paper deals only with generalizations of the group F. Thompson showed that T and V were infinite, simple, finitely presented groups, the first known examples of this kind.

F, which we will henceforward refer to as $F(2)$, represents the group of piecewise-linear orientation-preserving homeomorphisms of the closed unit interval with finitely many breakpoints in $\mathbb{Z}[\frac{1}{2}]$ and slopes in the cyclic multiplicative group $\langle 2\rangle$ in each linear piece. In [12], Higman defined an infinite class of groups $G_{n,r}$ which were a generalization of V (also known as G), where $n \in \{2, 3, 4, \dots\}$ and $r \in \mathbb{Z}[\frac{1}{n}]$; $G_{n,r}$ is then the group of piecewise-linear orientation-preserving right-continuous bijections of $[0, r)$ onto itself with finitely many breakpoints in $\mathbb{Z}[\frac{1}{n}]$, slopes in the cyclic multiplicative group $\langle n\rangle$ in each linear piece, and which maps $\mathbb{Z}[\frac{1}{n}] \cap [0, r)$ onto itself.

Brown then expanded this construction of Higman's by creating similar infinite families of groups $F_{n,r}$ and $V_{n,r}$, generalizing the groups F and V respectively (see [4]). In this paper we consider the groups $F_{n,1}$ for $n \in \{2, 3, 4, \dots\}$, and for simplicity we use the notation $F(n)$ instead of $F_{n,1}$. Thompson's group $F(n)$ is therefore defined in the following way:

Definition 1.1 (Thompson's group $F(n)$)**.** Thompson's group $F(n)$, for $n \geq 2$, is the group of piecewise-linear orientation-preserving homeomorphisms of the closed unit interval with finitely many breakpoints in $\mathbb{Z}[\frac{1}{n}]$ and slopes in the cyclic multiplicative group $\langle n\rangle$ in each linear piece.

Brown proved that each of the groups $F(n)$ for $n \geq 2$ is finitely presented, infinite-dimensional, torsion-free and of type FP_∞; this was an extension of the work done in [5], where Brown and Geoghegan showed that F is the first known example of a group with these properties. The automorphism groups of these groups have also been studied by Brin and Guzmán in [3]. Further information about Thompson's groups can be found in [7].

1.2. Almost convexity conditions. The concept of almost convexity was first developed by Cannon in [6] to develop algorithms for drawing the Cayley graphs of groups. If a group G is almost convex with respect to a given generating set, then an algorithm exists which can be used to draw the portion of the Cayley graph of G which can be depicted by the ball of radius m centered at the identity [6]. Minimal

almost convexity is a weaker condition than almost convexity. In fact, an entire range of almost convexity conditions exist, of which Cannon's almost convexity condition is the strongest, and the minimally almost convexity condition is the weakest nontrivial condition; therefore we begin with a more general definition of almost convexity conditions as a whole.

Throughout this paper, we let B_m denote the ball of radius m centered at the identity in the Cayley graph Γ of a group G. We use $d_\Gamma(g,h)$ to denote the distance between the elements g and h in the Cayley graph Γ, and $d_{B_m}(g,h)$ to denote the distance between the elements g and h in the ball B_m.

Definition 1.2 (almost convexity condition)**.** A group G satisfies an almost convexity condition with respect to the finite generating set X and a given function $f:\mathbb{N}\to\mathbb{R}_+$ if there exists a constant N such that for all $m>N$ and for all g and h in B_m satisfying $d_\Gamma(g,h)=2$, $d_{B_m}(g,h)\le f(m)$.

Almost convexity (Cannon) is the almost convexity condition in which $f(m)$ is a fixed constant. Minimal almost convexity is the almost convexity condition in which $f(m)=2m-1$. And since for any g and h in B_m there is always a path from g to h in B_m through the identity which has length in B_m bounded by $2m$ (i.e., $g^{-1}h$ or its inverse), we will always have $d_{B_m}(g,h)\le 2m$. So showing that a group is not minimally almost convex with respect to a given finite generating set is equivalent to stating that for arbitrarily large m there exist $g,h\in B_m$ with $d_\Gamma(g,h)=2$ such that any minimal length path in B_m between g and h in B_m will be of length $2m$, the same distance as the path $h^{-1}g$ or $g^{-1}h$ through the identity. The condition of minimal almost convexity is the weakest possible nontrivial generalization of almost convexity, and therefore any group which is not minimally almost convex satisfies no nontrivial almost convexity condition.

Definition 1.3 (minimally almost convex)**.** A group G is minimally almost convex with respect to the finite generating set X if there exists a constant N such that for all $m>N$ and for all g and h in B_m satisfying $d_\Gamma(g,h)=2$, $d_{B_m}(g,h)\le 2m-1$.

If a group is minimally almost convex (and therefore if it satisfies any nontrivial almost convexity condition), then it is finitely presented and its word problem is solvable (see [6] and [13]). Because of these consequences, this property has been studied for several groups already. Cleary and Taback have shown that the lamplighter groups are not minimally almost convex in [8]. Elder and Hermiller have shown in [10] that the Baumslag–Solitar groups $BS(1,2)$ and $BS(1,q)$, for $q\ge 7$, and that Stalling's non-FP_3 group are not minimally almost convex. Also, in considering a wide range of almost convexity conditions, it is obvious that stronger almost convexity conditions always imply weaker ones, but it is not known for all cases if this implication is biconditional. Some cases are known; for example, the work of Elder and Hermiller in [10] established the result that Poenaru's almost convexity condition (i.e., that the function $f(m)$ in the definition of almost convexity conditions is sublinear) is strictly stronger than minimal almost convexity.

1.3. Outline of results. In this paper we will show that Thompson's group $F(n)$ is not minimally almost convex for any $n\in\{2,3,4,\dots\}$. The result that Thompson's group $F(2)$ is not almost convex has already been proven by Cleary and Taback in [9], and the result that $F(2)$ is not minimally almost convex has already been proven by Belk and Bux in [2]. This paper essentially generalizes Belk and Bux's argument for $F(n)$ when $n>2$.

To show that $F(n)$ is not minimally almost convex, we find two elements, l and r (which are generalizations of the two elements used by Belk and Bux in [2]) which are distance 2 apart in the Cayley graph Γ but distance $2m$ apart in B_m (for arbitrarily large m). Using the metric on $F(n)$ developed by Fordham in [11], we then show that any minimal length path from l to r which remains in B_m must pass through a specific vertex h_r. We then define an abstract vertex h_l which must be on the path, and we use this to show that $d_\Gamma(h_r, h_l) \geq m+1$. Some straightforward algebra calculations then lead us to the main result:

Main Theorem 1 ($F(n)$ is not minimally almost convex)**.** *Let Γ be the Cayley graph of $F(n)$ with respect to the generating set $\{x_0, x_1, \ldots, , x_{n-1}\}$. For all even $m \geq 4$ there exist $l, r \in F(n)$ such that:*

(1) $d_\Gamma(l, r) = 2$.
(2) $|l|_{\{x_0, x_1, \ldots, , x_{n-1}\}} = |r|_{\{x_0, x_1, \ldots, , x_{n-1}\}} = m$.
(3) *For any path γ from l to r which remains in B_m,*

$$|\gamma|_{\{x_0, x_1, \ldots, , x_{n-1}\}} \geq 2m.$$

The fundamental outline of our proof is identical to that of Belk and Bux; however, our methods for proving each of the steps is somewhat different. Whereas Belk and Bux use forest diagrams to represent elements of $F(2)$, and as there is no known way to extend this method to $F(n)$ for $n = 2, 3, 4, \ldots$ in any meaningful way, this paper uses tree-pair diagrams to represent elements of $F(n)$ and uses Fordham's metric on $F(n)$, which is based on the use of tree-pair diagram representatives, to calculate length. As a result, many of the individual steps in this paper may look substantially different from the corresponding steps in the Belk and Bux proof for $F(2)$, even though the fundamental logic behind the two proofs is identical.

Acknowledgements. The author would like to thank Sean Cleary for his support and advice in the preparation of this article and the anonymous reviewer for thoughtful and comprehensive suggestions during the revision process.

2. Representation of $F(n)$ by tree-pair diagrams

Tree-pair diagrams consist of a pair of simple directed graphs, each of which is a subset of the plane. Each of the graphs in the diagram is referred to as a tree. In order to formally define these diagrams, we first begin with some basic definitions.

2.1. Basic definitions. An *n-ary caret* is a graph which has $n+1$ vertices joined by n edges: one vertex has degree n (the *parent*) and the rest have degree 1 (the *children*).

Another n-ary caret may then be attached to any of the n child vertices of the original caret so that the child vertex of the original caret serves simultaneously as the parent vertex of the new caret. A caret whose parent vertex is also the child vertex of a second caret will be referred to as a *child caret* of the second caret, and likewise, a caret whose child vertex is also the parent vertex of a second caret will be referred to as the *parent caret* of the second caret. We use the word "child" alone in this paper to refer sometimes to a child vertex and sometimes to a child caret; when this convention is used, which meaning is intended should be clear from the context.

A graph formed by joining any number of n-ary carets by using the child vertex of one caret as the parent vertex of another caret is referred to as an *n-ary tree*. An n-ary tree is generally depicted so that for any given caret in the tree, the child vertices are at the bottom and the parent vertex is at the top; when depicted in this way, the topmost caret is referred to as the *root caret* (or just the *root*) and its parent vertex is called the root node. When an n-ary caret is oriented in this way, its rightmost or leftmost edge is called the *right* or *left* edge, respectively. For any two vertices a and b on an n-ary tree, vertex a is the *ancestor* of vertex b if it is on the directed path from the root node to vertex b. Similarly, vertex b is the *descendent* of vertex a if vertex a is the ancestor of vertex b. If a vertex in the tree has degree 1, it is referred to as a *leaf*; if it has degree n or $n+1$, it is referred to as a *node*.

Because the distinction between our usage of the words *vertex*, *leaf*, and *node* will be essential in understanding the proofs that follow, we emphasize this in the following definition:

Definition 2.1 (vertex, leaf, node)**.** It is important to distinguish between vertices, nodes and leaves. Both leaves and nodes are vertices. Leaves are those vertices of degree 1 and nodes are those vertices of degree n or $n+1$ in an n-ary tree-pair diagram. We note that a node, in the context of this paper is not a synonym for vertex, but rather a proper subset of the set of all vertices in a tree-pair diagram.

We also codify the following:

Notation 2.2. We write $\mathbb{Z}^*$ for the nonnegative integers and $\mathbb{N}$ for the positive integers.

An element of $F(n)$ can be represented by an n-ary tree pair diagram. A *n-ary tree pair diagram* is a pair of n-ary trees containing the same number of leaves (which is equivalent to containing the same number of carets). The first tree in the pair is called the *negative tree* and the second tree in the pair is called the *positive tree*. This pair of trees is denoted (T_-, T_+). (The motivation for this choice of names will become clear when we see how the normal form for an element of $F(n)$ may be derived from the minimal tree-pair diagram representative of that element.) The leaves of each tree are numbered in increasing order from left to right (see subsequent subsection on leaf ordering for more detail), and the i^{th} leaf of the negative tree is paired with the i^{th} leaf of the positive tree.

2.2. Equivalence of elements of $F(n)$ and tree-pair diagrams. This representation of elements of $F(n)$ by n-ary tree-pair diagrams corresponds to the definition of $F(n)$ as the set of piecewise-linear orientation-preserving homeomorphisms of the closed unit interval in the following way: each leaf of each caret represents one subinterval of the closed unit interval. A single root caret, for example, has n leaves which represent n equal subintervals of the closed unit interval: $[0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \ldots, [\frac{n-1}{n}, 1]$. Then for any given leaf which represents a subinterval $[a, b]$, if we attach a child caret to that leaf then the n leaves of this child caret will represent the n equal subintervals $[a, a + \frac{b-a}{n}], [a + \frac{b-a}{n}, a + \frac{2(b-a)}{n}], \ldots, [a + \frac{(n-1)(b-a)}{n}, b]$. Then by starting with a single root caret and proceeding by adding child carets to its leaves and child carets to the leaves of its child carets, etc., we can build a tree whose leaves represent any subdivision of the closed unit interval whose subintervals all have endpoints in $\mathbb{Z}[\frac{1}{n}]$ and length of the form $\frac{1}{n^r}$ such that

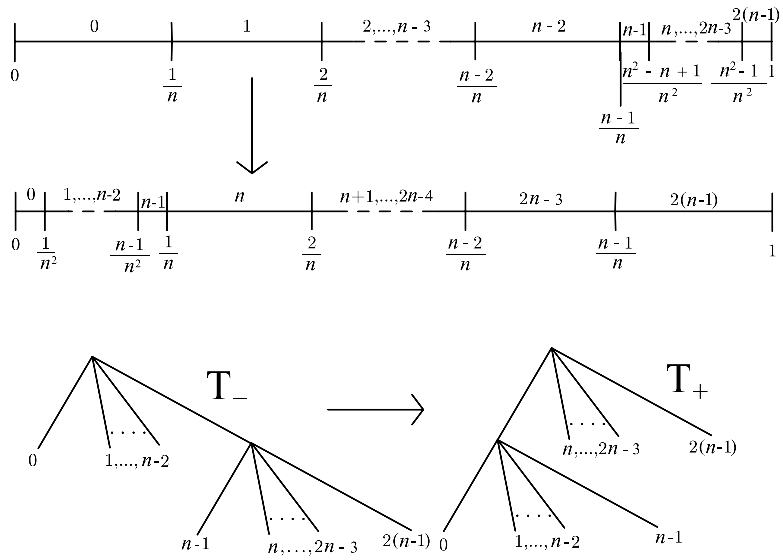

FIGURE 1. The homeomorphism of the closed unit interval and the tree-pair diagram representing the element x_0 in $F(n)$; the i^{th} interval is mapped to the i^{th} interval, and the i^{th} leaf is mapped to the i^{th} leaf.

$r \in \mathbb{Z}^*$. From this we can see that any element of $F(n)$ can be represented by an n-ary tree-pair diagram, and we can see that for any n-ary tree-pair diagram, an element of $F(n)$ must exist which can be represented by it. For example, Figure 1 shows one example of how an element in $F(n)$ can be represented by a tree-pair diagram.

In fact, for any element of $F(n)$, we can see that there must be an infinite number of n-ary tree-pair diagram representatives. To see this, we begin by considering Figure 2; we will see that the two tree-pair diagrams in this figure represent the same element of $F(n)$.

We can see that both diagrams send the domain subinterval $[\frac{a}{n}, \frac{a+1}{n}]$ to the range subinterval $[\frac{a}{n^2}, \frac{a+1}{n^2}]$, for $a \in \{0, 1, 2, \ldots, n-2\}$ and the domain subinterval $[\frac{n^2-b-1}{n^2}, \frac{n^2-b}{n^2}]$ to the range subinterval $[\frac{n-b-1}{n}, \frac{n-b}{n}]$, for $b \in \{0, 1, 2, \ldots, n-2\}$. The only difference we can see between the two diagrams is that the top tree-pair diagram sends each domain interval $[\frac{n^3-n^2+c}{n^3}, \frac{n^3-n^2+c+1}{n^3}]$ to the range interval $[\frac{n^2-n+c}{n^3}, \frac{n^2-n+c+1}{n^3}]$ for $c \in \{0, 1, 2, \ldots, n-1\}$ whereas the bottom tree-pair diagram sends the domain interval $[\frac{n-1}{n}, \frac{n^2-n+1}{n^2}]$ to the range interval $[\frac{n-1}{n^2}, \frac{1}{n}]$. However, if we look closely at the two maps

$$f_1 : \begin{cases} [\frac{n^3-n^2+c}{n^3}, \frac{n^3-n^2+c+1}{n^3}] \to [\frac{n^2-n+c}{n^3}, \frac{n^2-n+c+1}{n^3}] & c = 0 \\ \vdots & \vdots \\ [\frac{n^3-n^2+c}{n^3}, \frac{n^3-n^2+c+1}{n^3}] \to [\frac{n^2-n+c}{n^3}, \frac{n^2-n+c+1}{n^3}] & c = n-1, \end{cases}$$

$$f_2 : \left[\frac{n^2-n}{n^2}, \frac{n^2-n+1}{n^2}\right] \to \left[\frac{n-1}{n^2}, \frac{1}{n}\right],$$

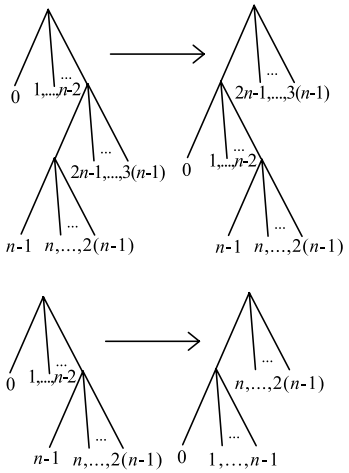

FIGURE 2. Two equivalent n-ary tree-pair diagrams.

we can see that they are in fact identical; we can "simplify" the top tree-pair diagram by replacing it with the bottom tree-pair diagram, and we can say that the two tree-pair diagrams in this figure are "equivalent."

Formally, we say that two n-ary tree-pair diagrams are *equivalent* if they both represent the same element of $F(n)$; this then induces an equivalence class on the set of n-ary tree-pair diagrams. So we say that an n-ary tree-pair diagram is *minimal* (or *reduced*) if it has the smallest number of leaves of any n-ary tree-pair diagram in its equivalence class. It is clear that any two equivalent n-ary tree-pair diagrams with the same number of leaves will be identical, so we know that for any element x of $F(n)$, we have a unique representative: the minimal tree-pair diagram of the given equivalence class of n-ary tree-pair diagrams which represent x.

2.3. Leaf ordering in a tree-pair diagram. We can number the leaves of each of the trees in an n-ary tree-pair diagram by thinking of each leaf as a subinterval of the closed unit interval; we number the leaves of the tree in increasing order from left to right with respect to their position as subintervals of the closed unit interval, and we begin our numbering with 0 (see Figure 1).

To see another example of a tree-pair diagram with all of its leaves numbered, see Figure 3.

2.4. Finding a minimal tree-pair diagram. If we have a tree-pair diagram which represents a given element of $F(n)$, we can judge whether or not it is minimal. We say that a caret on a tree is *exposed* if all of its children are leaves. In an n-ary tree-pair diagram, if we have an exposed caret in the positive tree and an exposed caret in the negative tree and all the leaf index numbers for both carets are identical, then we can remove each of the exposed carets from their respective trees without changing the element that the tree-pair diagram represents. This removal of unnecessary carets is equivalent to the removal of occurrences of n unnecessary equally sized subdivisions in the domain and range of the linear homeomorphism which maps the subinterval represented by the parent node of the removed caret

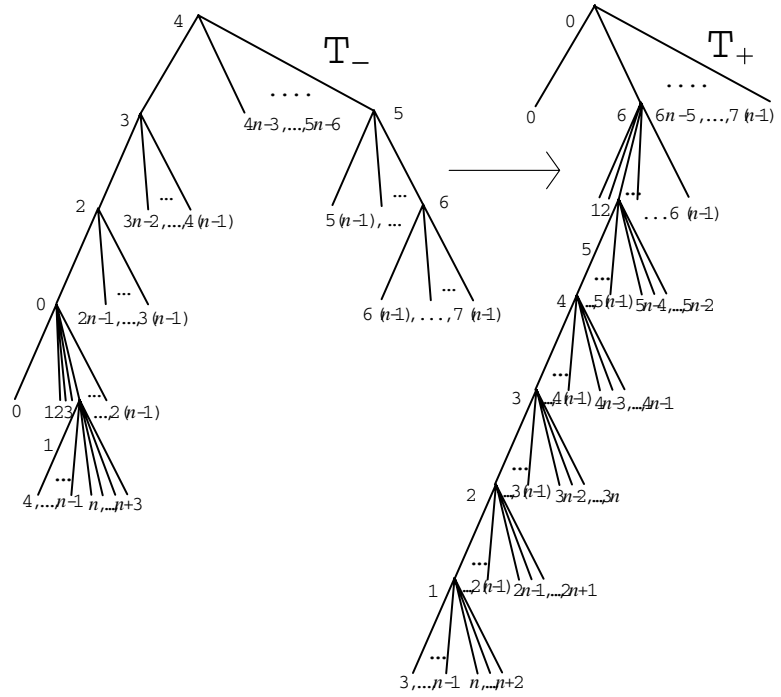

FIGURE 3. The minimal tree-pair diagram representative of the element $x_1 x_3^5 x_4^{-1} x_0^{-3}$ in $F(n)$ with all carets and leaves numbered.

in the negative tree onto the subinterval represented by the parent node of the removed caret in the positive tree.

By repeating this process as many times as is possible on a given tree-pair diagram representative of an element of $F(n)$, we can reduce it to the minimal representative. For example, we could remove the exposed caret pair with leaf index numbers $n-1, \ldots, 2(n-1)$ in the top tree-pair diagram of Figure 2 and replace it with the bottom tree-pair diagram in this figure using exactly this method.

We will often use the convention of writing $x = (T_-, T_+)$; this means that (T_-, T_+) is the minimal tree-pair diagram representative of the element x of $F(n)$.

2.5. Multiplying tree-pair diagrams. In this paper all multiplication is on the right, where the multiplication convention is that of function composition. Multiplying x by y on the right will be denoted xy, which actually denotes $x \circ y$. To see what this looks like for the tree-pair diagram representatives, let x and y be elements of $F(n)$ which are represented by the minimal tree-pair diagrams (T_-, T_+) and (S_-, S_+) respectively; then xy would be performed as multiplication on the tree-pair diagrams by performing the steps outlined in the following paragraph.

We want to make S_+ identical to T_- so the range of y is identical to the domain of x. This is possible if we notice that we can add a caret to a leaf of the tree S_+ as long as we add a caret to the leaf with the same index number in the tree S_-. This is allowed because it is just the reverse process of that of simplification of the tree-pair diagrams, analogous to subdividing the same subinterval in a homeomorphism

of the closed unit interval in exactly the same way in the domain and the range. Likewise, we can add a caret to a leaf of the tree T_- as long as we add a caret at the leaf with the same index number in the tree T_+. By repeatedly adding carets to S_+ and T_- (and therefore by extension to S_- and T_+), we can eventually turn them into identical trees. If we let S'_- denote the tree formed from S_- by adding carets to correspond to any carets added to S_+ in the process of making it identical to T_-, and if we let T'_+ denote the tree formed from T_+ by adding carets to correspond to any carets added to T_- in the process of making it identical to S_+, then the new tree-pair diagram for xy will be (S'_-, T'_+). We also may often denote the new tree pair diagram for xy by $((Ty)_-, (Ty)_+)$. To emphasize this point, we include it here as a separate remark:

Remark 2.3. When computing the product xy where $x = (T_-, T_+)$, the notation $((Ty)_-, (Ty)_+)$ denotes the tree-pair diagram which results from the composition of x and y, before it has been reduced by removing any exposed caret pairs. The notation $((Ty)'_-, (Ty)'_+)$ then denotes the tree-pair diagram which results from the composition of x and y, after it has been reduced by removing any exposed caret pairs.

To see an example of tree-pair multiplication, see Figure 4.

2.5.1. Presentations of $F(n)$. Thompson's group $F(n)$ has the following infinite presentation (Brown [4]):

$$F(n) = \{x_0, x_1, \cdots \mid x_j x_i = x_i x_{j+n-1} \text{ for } i < j\}$$

where the generators can be depicted by the tree-pair diagrams given in Figure 5.

The group $F(n)$ also has a finite presentation (Brown [4]) which will be needed to calculate length:

$$\left\{ x_0, x_1, \ldots, x_{n-1} \;\middle|\; \begin{array}{l} [x_0 x_i^{-1}, x_j] \text{ when } i < j, [x_0^2 x_i^{-1} x_0^{-1}, x_j] \text{ when } i \geq j-1, \\ [x_0^3 x_{n-1}^{-1} x_0^{-2}, x_1]. \text{ Here, } i, j = 0, \ldots, n-1. \end{array} \right\}$$

The generators for this finite presentation can be depicted by the first three tree-pair diagrams given at the left in Figure 5. The infinite presentation can be obtained from the finite presentation by induction. We refer to these two presentations as the standard infinite and finite presentations respectively.

2.6. Normal form of elements of $F(n)$. By looking at the relators for the standard infinite presentation for $F(n)$, it becomes clear that all elements of $F(n)$ can be put into the form

$$x_{i_1}^{r_1} x_{i_2}^{r_2} \ldots \; x_{i_n}^{r_n} x_{j_m}^{-s_m} \ldots \; x_{j_2}^{-s_2} x_{j_1}^{-s_1} \tag{1}$$

where the generators are taken from the standard infinite presentation such that

$$i_1 < i_2 < \cdots < i_n \neq \; j_m > \cdots > j_2 > j_1$$

To ensure uniqueness of this normal form we need only add the condition that if both x_i and x_i^{-1} appear in the above expression, then a generator x_j (or its inverse) where $i < j < i + n$, must also appear in the above expression (otherwise, we can use one of the relators in the infinite presentation to cancel x_i and x_i^{-1} or to substitute an equivalent expression in the infinite generators which still has the form given in Equation (1)). This normal form was first proved in [5].

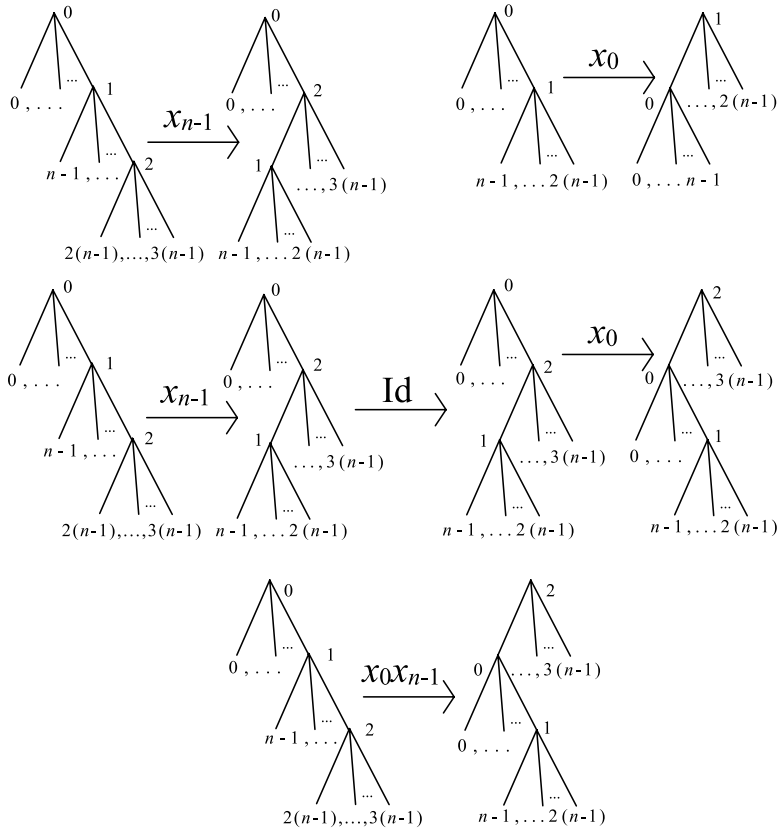

FIGURE 4. Multiplication of tree-pair diagrams for the product x_0x_{n-1} in $F(n)$ (before composition of the diagrams can be performed, one caret must be added to the leaf numbered $n-1$ in the tree-pair diagram for x_0 to make the domain tree of x_0 identical to the range tree of x_{n-1}). Here Id denotes the identity map.

Before we can explain how we can go from an n-ary tree-pair diagram representative to the normal form of the element of $F(n)$ which it represents (and vice versa), we need a few definitions. A *left side* is an edge which is the left edge of some caret in the tree which is neither the root caret, nor of type $\mathcal{R}$ (see Subsection 3.1). The *leaf exponent* of the i^{th} leaf in a tree is the number of consecutive left sides on the directed path from the i^{th} leaf to the root node (any left edges that appear on this path after the appearance of a nonleft edge will be excluded from this number). If the directed path from the i^{th} leaf to the root node does not begin with any left sides, then the leaf exponent for the leaf numbered i is zero.

To find all generators with positive exponents in the normal form, we look at the positive tree, and to find all generators with negative exponents in the normal form, we look at the negative tree. The positive (negative) exponent of x_i in the normal form is the leaf exponent of the leaf numbered i in the positive (negative) tree of the minimal tree-pair diagram representative. Using this fact, it is straightforward

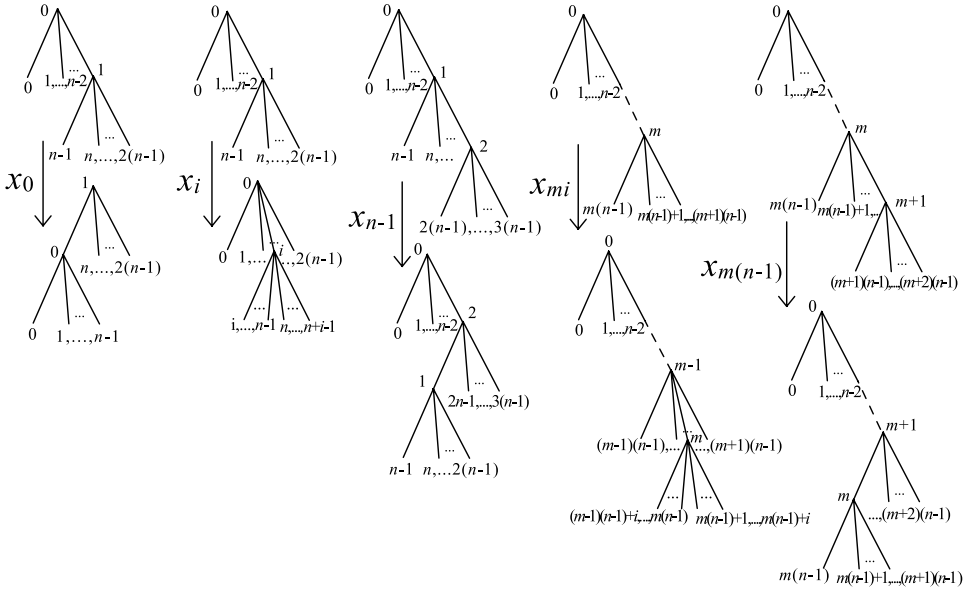

FIGURE 5. The generators $\{x_0, x_1, x_2, \dots\}$ for the standard infinite presentation of $F(n)$ (where $i = 1, 2, \dots, n-2$ and $m \in \mathbb{N}$).

to go from an element of $F(n)$ in normal form to a tree-pair diagram representative and vice versa. For example, the element $x_1 x_3^5 x_4^{-1} x_0^{-3}$ in $F(n)$ can be depicted by the tree-pair diagram given in Figure 3.

2.7. Action of generators on tree-pair diagrams and the critical leaf. When we multiply an element $w = (T_-, T_+)$ of $F(n)$ on the right by another element y, we can think of the multiplication in this way: the element y is acting on the tree-pair diagram (T_-, T_+) in some way to turn it into the tree-pair diagram $((Ty)_-, (Ty)_+)$ which, if not already minimal, will be simplified to the tree-pair diagram $((Ty)'_-, (Ty)'_+)$. If we let S_- and S_+ represent the tree obtained from T_- and T_+ respectively by adding any carets to (T_-, T_+) which will be needed in order to multiply it by y, then for $y^{\pm 1} \in \{x_0, x_1, \dots, , x_{n-1}\}$, we can think of the action of y on S_- as a kind of rotation. When $y = x_0$, the action of y on S_- is a kind of clockwise rotation along the path from the leftmost child vertex of the root to the root vertex to the rightmost child vertex of the root. When $y = x_i$ for $i \in \{1, \dots, n-2\}$, the action of y on S_- is a kind of clockwise rotation along the path from the i^{th} child vertex of the root to the root vertex to the rightmost child vertex of the root. When $y = x_{n-1}$, the action of y on S_- is a kind of clockwise rotation along the path from the leftmost child vertex of the rightmost child caret of the root to the rightmost child vertex of the root, to the rightmost child vertex of the rightmost child caret of the root. The inverse of each of these generators has the same action on S_- except in the reverse direction, going counterclockwise. This action of the generators on S_- can be seen in Figure 6.

Throughout this paper, we will refer to the rotation action of the generator x_0 described above as *clockwise rotation through the root* and the rotation action of its inverse as *counterclockwise rotation through the root.*

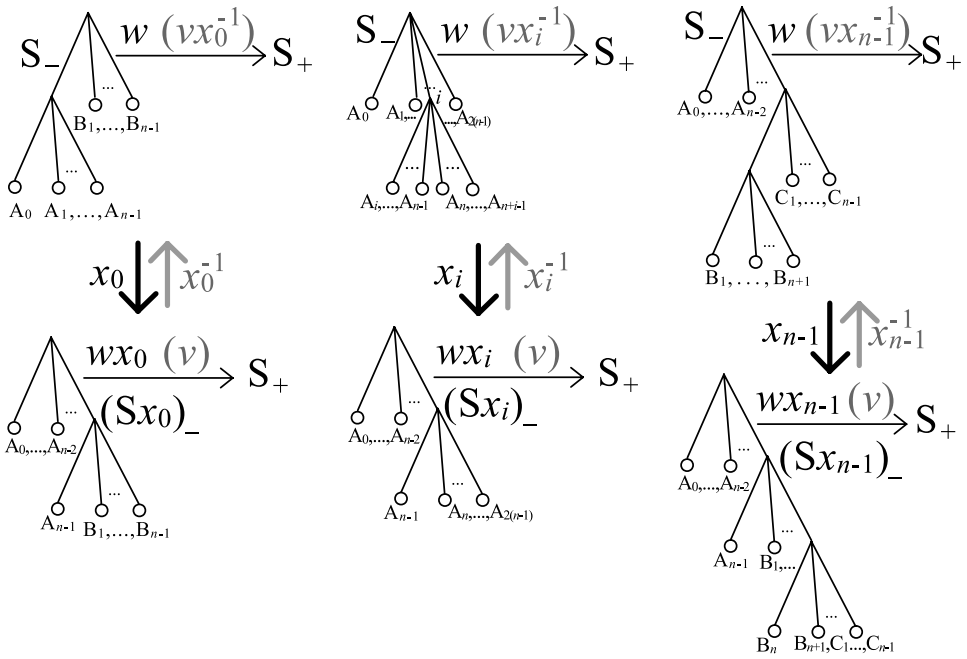

FIGURE 6. The action of a given generator (or its inverse) in the standard finite generating set of $F(n)$ on an arbitrary n-ary tree-pair diagram. The black arrows and labels indicate the action of the generator on the tree-pair diagram representative of an arbitrary word w, and the grey arrows and labels indicate the action of that generator's inverse on the tree-pair diagram representative of an arbitrary word v. (Here $i \in \{1, \ldots, n-2\}$.)

We want to be able to identify a way to describe this action of x_0 on S_- because this will be one of the central ideas of the proof of the main theorem in this paper, so this motivates the following definition, which allows us to assign a special status to a specific leaf in a tree; when x_0 then acts on that tree through rotation, the index number of this special leaf will change.

Definition 2.4 (right foot)**.** Let $\wedge_s$ be the first caret of type $\mathcal{R}$ (see Subsection 3.1) (if one exists) in an n-ary tree.

(1) If the leftmost child vertex of $\wedge_s$ is a node rather than a leaf, then we consider the subtree with the leftmost child vertex of $\wedge_s$ as the root node. Within this new subtree, the critical leaf is the leaf with the highest index number in the subtree.
(2) If $\wedge_s$ has a leaf as its leftmost child vertex, then the critical leaf is the leftmost leaf of $\wedge_s$.
(3) If the tree has no carets of type $\mathcal{R}$, then the critical leaf is the rightmost leaf of the root caret.

We let $\mathrm{crit}(T)$ denote the leaf index of the critical leaf in the tree T.

To see an example of several trees with the critical leaf labeled, see Figure 7.

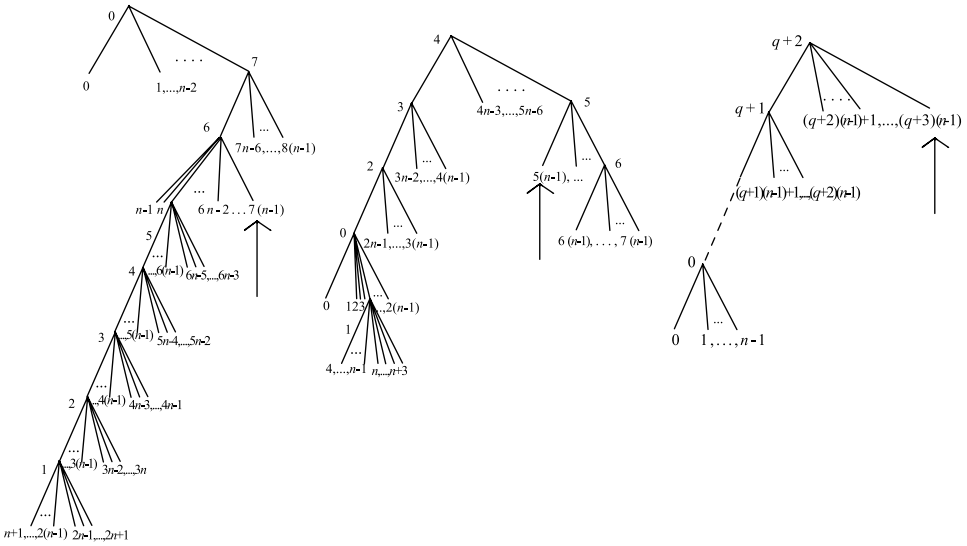

FIGURE 7. Critical leaf indicated by arrow in several n-ary trees.

So in any given tree-pair diagram, we call the critical leaf in the negative and positive tree the negative and positive critical leaf respectively. We let $\mathrm{crit}(T_-)$ and $\mathrm{crit}(T_+)$ denote the leaf index of the negative and positive critical leaves respectively in (T_-, T_+). We say that an element $x = (T_-, T_+)$ of $F(n)$ (or its minimal tree-pair diagram representative (T_-, T_+)) is *balanced* if $\mathrm{crit}(T_-) = \mathrm{crit}(T_+)$. Because we will use this terminology at several key points in the proof of the major theorem of this paper, we highlight this definition here.

Definition 2.5 (balanced, positive, negative)**.** An element $x = (T_-, T_+)$ of $F(n)$ is balanced when $\mathrm{crit}(T_-) = \mathrm{crit}(T_+)$. Similarly, we say that x is positive when $\mathrm{crit}(T_-) < \mathrm{crit}(T_+)$, and that x is negative when $\mathrm{crit}(T_-) > \mathrm{crit}(T_+)$. (The definitions of positive and negative here bear no relation to the definitions of positive and negative given in Belk and Bux's paper in [2].)

We note that for $w = (T_-, T_+)$ of $F(n)$, after any carets have been added to (T_-, T_+) to get (S_-, S_+) so that multiplication by x_0 can take place, the action of x_0 on S_- will be to change $\mathrm{crit}(S_-)$ by decreasing it by a multiple of $n-1$. The action of x_0 on (S_-, S_+) however, will leave $\mathrm{crit}(S_+)$ unchanged. (The act of adding carets to (T_-, T_+) will only increase the critical leaf index in each tree by $n-2$ for each caret added.)

3. Fordham's metric on $F(n)$

In [11], Fordham developed a method to calculate the length of any given element of $F(n)$ with respect to the standard finite generating set $\{x_0, x_1, \ldots, , x_{n-1}\}$. Fordham's method depends upon numbering the carets in each tree of a tree-pair diagram and then classifying each of the caret pairs into one of several different types: the motivating idea being that certain types of caret pairs can only be obtained if a certain set of generators with a certain cardinality has been used to

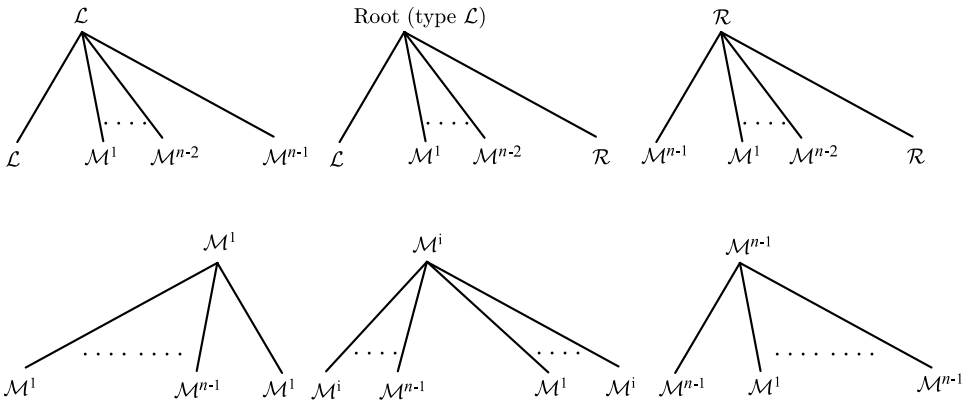

FIGURE 8. For each of the 6 parent caret types given above, the caret type listed below each child vertex is the type of the child caret, if one exists.

obtain the tree-pair diagram from the empty diagram which represents the identity element. So before we introduce Fordham's metric, we first show how the carets in a tree-pair diagram can be numbered and classified into distinct categories or types. The rules for caret numbering and classification are all paraphrased here from [11].

3.1. Basic classification of caret types. We begin by classifying carets in any n-ary tree into one of three major types. Later we will subdivide these categories further into more specific subtypes. In the following, when we refer to the *left* (*right*) *edge* of a tree, we mean the subgraph of the tree which consists of the path from the root to the leftmost (rightmost) leaf.

The three main types of carets in an n-ary tree-pair diagram are:

(1) $\mathcal{L}$. This is a left caret; a left caret is any caret that has one edge on the left edge of the tree. The root caret is considered to be of this type.
(2) $\mathcal{R}$. This is a right caret; a right caret is any caret (except the root caret) that has one edge on the right edge of the tree.
(3) $\mathcal{M}$. This is a middle caret; a middle caret is any caret that is neither a left nor a right caret.

3.2. Further classification of carets of type $\mathcal{M}$. Carets of type $\mathcal{M}$ can be further classified depending upon their placement with respect to other caret types in the tree. We will take all carets of type $\mathcal{M}$ and subdivide them into $n-1$ different subtypes each of which we will call type $\mathcal{M}^i$, for $i = 1, \ldots, n-1$. The value of i depends upon the caret type of the middle caret's parent caret. To see how i is determined for different parent caret types, see Figure 8. For example, if the parent caret is of type $\mathcal{L}$, then all except the leftmost child vertices are numbered from left to right so that any caret hanging off of the first of these vertices is type $\mathcal{M}^1$, the type of any caret hanging off the second of these vertices is type $\mathcal{M}^2$, etc., and the type of any caret hanging off the last of these vertices is type $\mathcal{M}^{n-1}$.

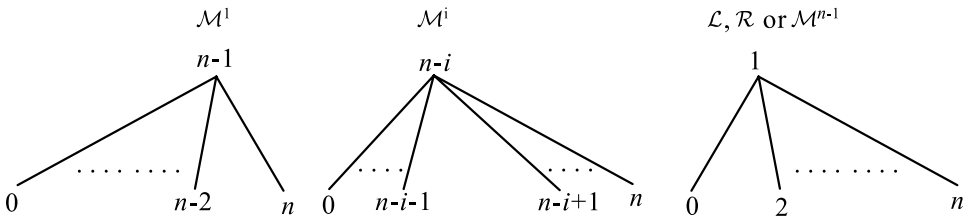

FIGURE 9. The ordering of child nodes (if they exist) with respect to the parent node in different caret types in tree-pair diagrams of elements of $F(n)$.

3.3. Ordering the carets in an n-ary tree-pair diagram. The nodes (recall that a node is always a vertex of degree n or $n+1$) of each tree are ordered in the following way. If we were to draw a vertical line through the parent node of a caret, the children drawn to the left of this line would be referred to as the *left children* of the parent, and the children drawn to the right of this line would be referred to as the *right children* of the parent. The caret type will determine which children of a caret are drawn to the left of the parent node and which are drawn to the right (see Figure 9). When numbering the nodes on the tree, we will number any left child nodes first from left to right, then number the parent node, then number any right child nodes from left to right. For example, if a caret is of type $\mathcal{M}^1$, then any existing child nodes will be left children, and therefore numbered before it, except any child caret which has as its parent the child vertex which is farthest to the right; this child caret, if it exists, will be a right child and will therefore be numbered after the parent node. Because all carets in an n-ary tree have at least one node which is the parent or child of another caret within the tree, the ordering of the nodes of a single caret induces an ordering of all the nodes in a tree.

The numbering of the nodes induces a numbering of the carets if we let a caret's number be the same as the number given to its parent node. We will denote the i^{th}-caret by $\wedge_i$. For example, to see an element of $F(n)$ with all its carets ordered numerically, see Figure 3.

3.4. Final classification of caret types. Now that we have a method for ordering the carets in a tree, we can proceed to further subclassify the caret types $\mathcal{L}$, $\mathcal{R}$, $\mathcal{M}^1,\ldots,\mathcal{M}^{n-1}$ in an n-ary tree into more specific caret subtypes. This further subcategorization is necessary in order to proceed with Fordham's method for calculating word length. In order to construct these categories, we need to define the following key terms: a caret $\wedge_i$ is a *successor* of the caret $\wedge_j$ if and only if $i > j$. A caret $\wedge_i$ is the *immediate successor* of the caret $\wedge_j$ if and only if $i = j+1$. A caret $\wedge_i$ is a *predecessor* of the caret $\wedge_j$ if and only if $\wedge_j$ is a successor of $\wedge_i$. We note here that the definitions of successor (predecessor) and child or descendent (parent or ancestor) should not be confused; the successor, even the immediate successor, of a caret may not be the child or the descendent of that caret, and vice versa. (For example, in Figure 3, in T_+ $\wedge_5$ is the child but not a successor of $\wedge_6$, and in T_-, $\wedge_2$ is a successor of $\wedge_0$, even though it is not a descendent of $\wedge_0$.)

Here is the final list of caret types:

(1) $\mathcal{L}_\emptyset$. This is the first left caret (and therefore the first caret) in a tree, which will therefore have index number 0. Every nonempty tree has one and only one caret of this type.
(2) $\mathcal{L}_{\mathrm{L}}$. This is any left caret except the single $\mathcal{L}_\emptyset$ caret.
(3) $\mathcal{R}_\emptyset$. This is any right caret with all successor carets of type $\mathcal{R}$.
(4) $\mathcal{R}_{\mathrm{R}}$. This is a right caret whose immediate successor is of type $\mathcal{R}$, but which has at least one successor which is not of type $\mathcal{R}$.
(5) $\mathcal{R}_j$. This is a right caret whose immediate successor is not a right caret and whose leftmost child successor is of type $\mathcal{M}^j$, (where clearly we must have $j < n-1$). If the leftmost child successor is of type $\mathcal{R}$, we let $j = n-1$.
(6) $\mathcal{M}^i_\emptyset$. This is a middle caret of type $\mathcal{M}^i$ that has no child successor carets.
(7) $\mathcal{M}^i_j$. This is a middle caret of type $\mathcal{M}^i$ with leftmost child successor of type $\mathcal{M}^j$. (Note that $j \leq i$.)

3.5. Fordham's method for computing word length in $F(n)$. We now describe Forham's method [11] for computing the length of words in $F(n)$ with respect to the standard finite generating set. We recall that in a tree-pair diagram, all carets in the positive and negative trees are numbered, and the caret numbered i in the negative tree is paired with the caret numbered i in the positive tree. We will refer to this as the i^{th} *caret pair* in the tree-pair diagram and will denote it by $\wedge_i$. (We note that the notation $\wedge_i$ may be used to represent a single caret numbered i in a single tree, or the pair of carets numbered i in a tree-pair diagram; when this notation is used, which of these is meant should be clear from the context.)

Throughout this paper, we use the notation $|x|$ to denote the length of the element x in $F(n)$ with respect to the standard finite generating set. Because this notation is used so often, we set it apart in a separate remark:

Remark 3.1. For a given element x in $F(n)$, the notation $|x|$ always represents the length of x with respect to the standard finite generating set $\{x_0, x_1, \ldots, , x_{n-1}\}$.

To determine $|x|$ of an element x in $F(n)$, we consider the minimal tree-pair diagram representative of x. We make a list of each caret pair in the diagram giving the type of each caret in the pair, and then we consult a table that assigns a "weight" to each possible caret pairing that could be obtained in an n-ary tree-pair diagram. The *weight* of a caret pair in a tree-pair diagram is the contribution of that caret pairing to the length of the element of $F(n)$ which the diagram represents. The weight which is assigned to a caret pair comes from the cardinality of the set of generators which is required to create such a caret pair. Table 1 displays these weights.

We will use the notation $w(\wedge_i)$ or $w(\tau_1, \tau_2)$ to denote the weight, given by Fordham's table, of the i^{th} caret pair in the tree-pair diagram, where the types of each caret in the pair are denoted by τ_1 and τ_2. Since the table is symmetric, we will always have $w(\tau_1, \tau_2) = w(\tau_2, \tau_1)$ for any caret types τ_1 and τ_2. We note that carets of type $\mathcal{L}_\emptyset$ are not listed on the table; since there is only one caret of this type in any given tree, and since this will always be the type of the first caret in each tree, the only pairing possible is ($\mathcal{L}_\emptyset$,$\mathcal{L}_\emptyset$), which will occur only once in any given tree-pair diagram and have weight $w(\mathcal{L}_\emptyset, \mathcal{L}_\emptyset) = 0$.

To calculate the length of an element, we then need only sum the weights of the caret pairs taken from its minimal tree-pair diagram representative.

TABLE 1. Weight of types of caret pairs in the n-ary tree-pair diagram ($j_1 \leq i < j_2, i_1 < j \leq i_2$).

$(\ ,\ )$	$\mathcal{L}$	$\mathcal{R}_\emptyset$	$\mathcal{R}_{\mathrm{R}}$	$\mathcal{R}_j$	$\mathcal{M}^i_\emptyset$	$\mathcal{M}^i_j$
$\mathcal{L}$	2	1	1	1	2	2
$\mathcal{R}_\emptyset$	1	0	2	2	1	3
$\mathcal{R}_{\mathrm{R}}$	1	2	2	2	1	3
$\mathcal{R}_{j_1}$	1	2	2	2	3	3
$\mathcal{M}^{i_1}_\emptyset$	2	1	1	1	2	2
$\mathcal{M}^{i_1}_{j_1}$	2	3	3	3	4	4
$\mathcal{R}_{j_2}$	1	2	2	2	1	3
$\mathcal{M}^{i_2}_\emptyset$	2	1	1	3	2	4
$\mathcal{M}^{i_2}_{j_2}$	2	3	3	3	2	4

Theorem 3.2 (Fordham [11], Theorem 2.0.11)**.** *Given an element w in $F(n)$ represented by the minimal tree-pair diagram (T_-, T_+), the length $|w|$ of the element w with respect to the generating set $\{x_0, x_1, \ldots, , x_{n-1}\}$ is the sum of the weights of each of the pairs of carets in the tree-pair diagram.*

3.6. Effect of multiplication on caret type pairings. When we multiply an element $x = (T_-, T_+)$ of $F(n)$ on the right by another element y, we can think of the multiplication in this way: the element y is acting on the tree-pair diagram (T_-, T_+) in some way to turn it into the tree-pair diagram $((Ty)_-, (Ty)_+)$ which, if not already minimal, will be simplified to the tree-pair diagram $((Ty)'_-, (Ty)'_+)$.

Remark 3.3. When multiplying an arbitrary element $x = (T_-, T_+)$ by an arbitrary element y on the right, if no carets need be added to (T_-, T_+) to compute the product xy, then the type of caret $\wedge_i$ is the same in both T_+ and $(Ty)_+$ for all caret index numbers i. In fact, the only case in which the type of caret $\wedge_i$ will be different in $(Ty)'_+$ than in T_+ for some caret index number i, is the case in which $((Ty)_-, (Ty)_+)$ is not minimal, i.e., $((Ty)_-, (Ty)_+) \neq ((Ty)'_-, (Ty)'_+)$.

In the case in which $((Ty)_-, (Ty)_+) \neq ((Ty)'_-, (Ty)'_+)$, each caret pair which must be removed from $((Ty)_-, (Ty)_+)$ in order to obtain $((Ty)'_-, (Ty)'_+)$ will cause only one of the following changes to the list of caret types in T_+ to obtain the list of caret types for $(Ty)'_+$. One of the items on the list will be removed, and exactly one of the following will occur:

(1) All other caret types will remain the same.
(2) One caret of the form $\mathcal{M}^i_j$ will become type type $\mathcal{M}^i_\emptyset$ or type $\mathcal{M}^i_k$ for some $k > j$.
(3) One caret of the form $\mathcal{R}_j$ will become type $\mathcal{R}_{\mathrm{R}}$, or type $\mathcal{R}_\emptyset$, or type $\mathcal{R}_k$ for some $k > j$, and zero or more carets of type $\mathcal{R}_{\mathrm{R}}$ will become type $\mathcal{R}_\emptyset$.

This fact will be used repeatedly in proofs of several of the theorems presented later in this paper.

For cases when exact calculation of the weights of caret pairs in a tree-pair diagram is not possible or not desirable, the following theorems may be helpful in evaluating the effect multiplication by a generator will have on the length of an element. We begin by making a definition.

Definition 3.4. Let x be a generator of $F(n)$ and let $w = (T_-, T_+) \in F(n)$ be a reduced q-caret tree-pair diagram.

(1) We say that wx satisfies the *subtree condition* if we can compute the product wx without adding any carets to (T_-, T_+).
(2) We say that wx satisfies the *minimality condition* if $((Tx)_-, (Tx)_+)$ is minimal.

Theorem 3.5 (Fordham [11], Theorem 2.1.1)**.** *Let x be a generator of $F(n)$ and let $w = (T_-, T_+) \in F(n)$ be a reduced q-caret tree-pair diagram. If wx satisfies both the subtree condition and the minimality condition, then there is exactly one caret $\wedge_i$ (where $i < q$) that changes type; that is, if we let $\tau_{T_-}(\wedge_i)$ denote the caret type of $\wedge_i$ in T_- in the tree-pair diagram (T_-, T_+), then $\exists\ i < q$ such that*

$$\tau_{T_-}(\wedge_i) \neq \tau_{(Tx)_-}(\wedge_i) \text{ and } \tau_{T_-}(\wedge_j) = \tau_{(Tx)_-}(\wedge_j) \forall\ j \neq i.$$

We note that the caret $\wedge_i$ which changes type when the conditions of Definition 3.4 are met will always be in the negative tree (see Remark 3.3).

We note that when the subtree condition is satisfied, $(Tx)_+$ will be a subtree of T_+, with equality only when the minimality condition is met. When the conditions in Defintion 3.4 fail, we have two alternate theorems:

Theorem 3.6 (Fordham [11], Theorem 2.1.3)**.** *If x is a generator of $F(n)$ and $w = (T_-, T_+) \in F(n)$, and the product wx does not fulfill the subtree condition of Definition* 3.4*, then $|wx| > |w|$.*

Theorem 3.7 (Fordham [11], Theorem 2.1.4)**.** *If x is a generator of $F(n)$ and $w = (T_-, T_+) \in F(n)$, and the product wx does not fulfill the minimality condition of Definition* 3.4*, then $|wx| = |w| - 1$.*

4. Proof of the main theorem

For the duration of this paper, we let Γ denote the Cayley graph of $F(n)$ with respect to the standard finite generating set, we let B_m denote the ball of radius m centered at the identity in the Cayley graph, and we use $d_\Gamma(g, h)$ and $d_{B_m}(g, h)$ to denote the distance between the elements g and h in the Cayley graph and the ball B_m respectively. We use the convention that the edges of the Cayley graph denote multiplication by a generator (or its inverse) of the standard finite generating set on the right of the element represented by a given vertex. A *path* in Γ is therefore a directed path which goes from one vertex to another along edges which represent multiplication on the right by a generator (or its inverse). For all tree-pair diagrams given in the proof of the main result of this paper, circles are used to denote (possibly empty) subtrees, and exposed child vertices without circles are used to denote leaves. We restate the main theorem of our paper here for ease of reading.

We seek to prove the following:

Main Theorem 1 ($F(n)$ is not minimally almost convex)**.** *Let Γ be the Cayley graph of $F(n)$ with respect to the standard finite generating set. For all even $m \geq 4$ there exist $l, r \in F(n)$ such that:*

(1) $d_\Gamma(l, r) = 2$.
(2) $|l| = |r| = m$.
(3) *For any path γ from l to r which remains in B_m, $|\gamma| \geq 2m$.*

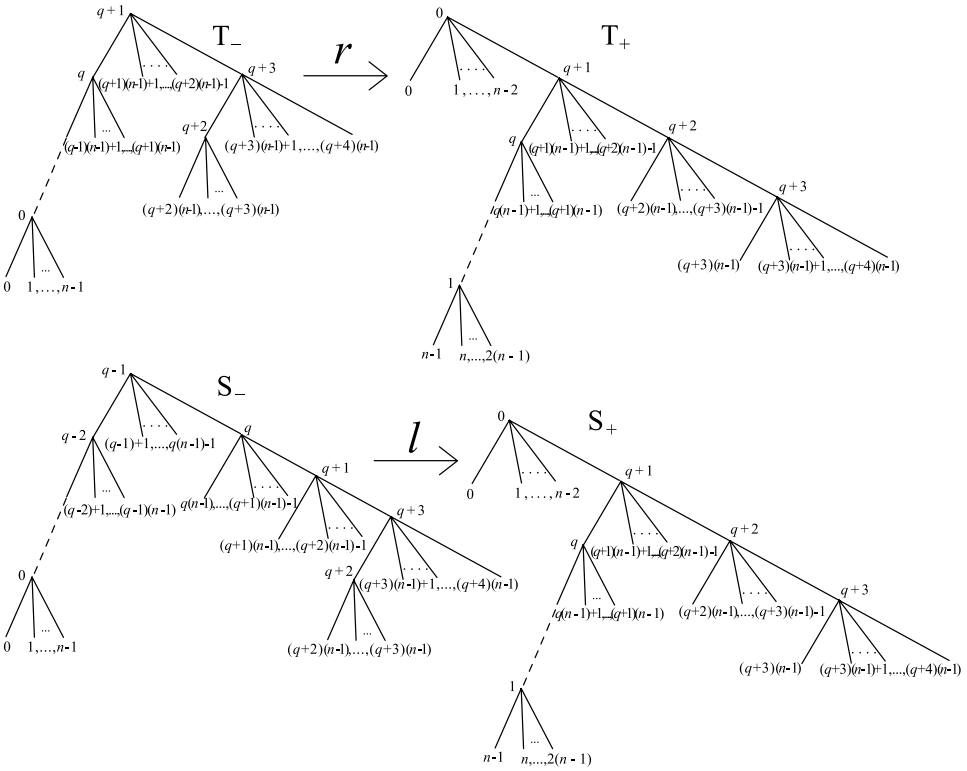

FIGURE 10. r (top) and l (bottom) in $F(n)$.

4.1. Choosing r and l. We begin by choosing two specific elements r and l, which are those elements used by Belk and Bux in [2] in their proof that $F(2)$ is not minimally almost convex, but which we have generalized to the case $n \in \{2, 3, 4, \dots\}$:

$$r = x_{n-1}^{q} x_0^{-(q+1)} x_{n-1}^{-1}$$

and

$$l = r x_0^2$$

and we let $m = 2q + 2$ where $q \geq 1$. The minimal tree-pair diagram representatives of r and l can be seen in Figure 10.

4.1.1. Intuitive motivation behind the proof of the main theorem. The intuitive motivation behind our proof of the main theorem is identical to that used by Belk and Bux to prove that $F(2)$ is not minimally almost convex in [2]. However, their usage of forest diagrams as representatives of elements of $F(2)$ makes the actions of the generators much more transparent: the action of one generator is always to delete or add caret pairs, and the action of the other generator is always to move the arrow in the diagram. (For more details on how forest diagrams can be used to represent elements of $F(2)$, see [1].) However, there is no known way to use forest diagrams as representatives of $F(n)$ that gives us such a simple view of the actions of the generators. So in order to observe how generators act on elements in

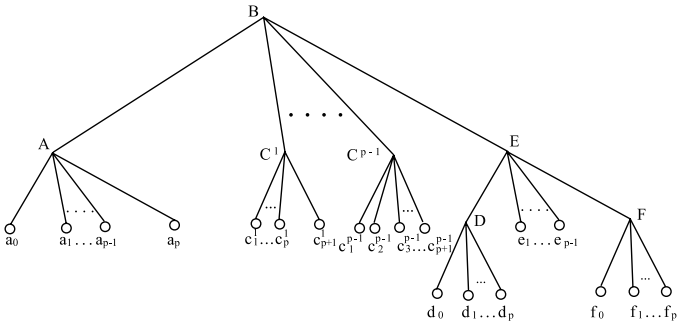

FIGURE 11. A (possibly nonminimal) generic representation of a negative tree in an arbitrary n-ary tree-pair diagram (where $p = n - 1$).

$F(n)$, we have to look at how the action of generators on tree-pair diagrams can be viewed as rotations within the tree, and we need to be adept at looking at how this can affect the types of carets in the tree-pair diagram and how Fordham's metric can then be applied to see how rotation in one of the trees of the minimal tree-pair diagram can affect the length of an element.

To see that this is the case, we begin by making some observations about how generators (in the standard finite generating set) act on arbitrary tree-pair diagrams. Figure 11 can be used to represent a negative tree in an arbitrary n-ary tree-pair diagram (this diagram may not be minimal).

Once a negative tree has been written in the form given in Figure 11, the action of any given generator will change only one of the types of the carets in that tree. (To understand what follows, it may be helpful to refer again to Figure 6.) Each generator will produce the following type change:

(1) x_0 takes the type of $\wedge_B$ from $\mathcal{L}_{\mathrm{L}}$ to $\mathcal{R}_*$.
(2) x_0^{-1} takes the type of $\wedge_E$ from $\mathcal{R}_*$ to $\mathcal{L}_{\mathrm{L}}$.
(3) x_i for $i = 1, \ldots, n-2$ takes the type of $\wedge_{C^i}$ from $\mathcal{M}_*^i$ to $\mathcal{R}_*$.
(4) x_i^{-1} for $i = 1, \ldots, n-2$ takes the type of $\wedge_E$ from $\mathcal{R}_*$ to $\mathcal{M}_*^i$.
(5) x_{n-1} takes the type of $\wedge_D$ from $\mathcal{M}_*^{n-1}$ to $\mathcal{R}_*$.
(6) x_{n-1}^{-1} takes the type of $\wedge_E$ from $\mathcal{R}_*$ to $\mathcal{M}_*^{n-1}$.

Similarly, each generator can only expose the carets in certain positions in the negative tree. If a negative tree has the form given in Figure 11, then the only caret(s) which may be exposed by the action of the given generator are:

(1) x_0 may expose $\wedge_B$ (and if $\wedge_B$ cancels, then $\wedge_A$ may cancel as well).
(2) x_0^{-1} may expose $\wedge_B$ (and if $\wedge_B$ cancels, then $\wedge_E$ may cancel as well).
(3) x_i for $i = 1, \ldots, n-2$ may expose $\wedge_{C^i}$ (and if $\wedge_{C^i}$ cancels, then $\wedge_B$ may cancel as well).
(4) $x_i^{-1}$for $i = 1, \ldots, n-2$ may expose $\wedge_E$ (and if $\wedge_E$ cancels, then $\wedge_B$ may cancel as well).
(5) x_{n-1} may expose $\wedge_E$ (and if $\wedge_E$ cancels, then $\wedge_D$, followed by $\wedge_B$ may cancel as well).
(6) x_{n-1}^{-1} may expose $\wedge_E$ (and if $\wedge_E$ cancels, then $\wedge_F$, followed by $\wedge_B$ may cancel as well).

The intuitive idea behind the proof of the main theorem of this paper requires that we think of movement along the arbitrary fixed path γ in B_m from l to r as a series of actions on the tree-pair diagram which begins as (S_-, S_+) (the minimal tree-pair diagram representative for l). We think of each edge on the path as an action by that generator (or its inverse) on the tree-pair diagram. What follows is an informal presentation of those ideas which motivate the proof of the main theorem and the given choice of r and l. In this section we have forgone a certain degree of formality in an attempt to describe some basic intuitive ideas in a simple, straightforward way, but this informal approach will be abandoned once we return to the formal proof of the main theorem.

The main idea is this: to go from l to r while remaining in B_m, we must first cancel the caret pair $\wedge_1$ in (S_-, S_+), then cancel the caret pair $\wedge_{q+2}$ (where $q+2$ refers to its original index number in (S_-, S_+), not its current index number at the time of cancelation), then add back the caret pair $\wedge_1$ (again, the index here refers to its original index number in (S_-, S_+)), then add back the caret pair $\wedge_{q+2}$, in that order, and this series of steps will always require a path of length $2m$ or greater.

To see why this might be the case, we go into a little more detail here, although the structure of our formal proof will look somewhat different.

First we notice how the weights of each of the carets in (S_-, S_+) determines the length of l (see Proof of part (2) of Main Theorem 1 for details):

(1) The first and last caret pairs contribute nothing to the length, which is the least possible contribution that can be made in any tree by the first and last caret pairs.
(2) The caret pairs with index numbers $q, q+2$ each contribute 1 to the total length of the element, which is the least possible contribution by these carets, as all caret pairs that are not the first or last pair in a tree-pair diagram must contribute a weight of at least 1 to the total length.
(3) The caret pairs with index numbers $1, 2, 3, \ldots, q-1, q+1$ contribute a weight of 2 each to the length of the element.

In order to get from l to r, we need to move the caret $\wedge_q$ in S_- counterclockwise through the root. But this will change the type of that caret from $\mathcal{R}_{\mathrm{R}}$ to $\mathcal{L}_{\mathrm{L}}$, changing the type pair from $(\mathcal{R}_{\mathrm{R}}, \mathcal{M}_{\emptyset}^{n-1})$ to $(\mathcal{L}_{\mathrm{L}}, \mathcal{M}_{\emptyset}^{n-1})$, increasing the weight of this caret pair from 1 to 2. But this change in type will increase the length of the element to $m+1$ and therefore the resulting element will be outside B_m. So before we can move $\wedge_q$ counterclockwise, we must cancel some caret pair in the tree-pair diagram which contributes at least a weight of 1 to the total length (i.e., not the first or last caret pair in the diagram) or we must change the type of one of the carets in one of the caret pairs with index $1, 2, 3, \ldots, q-1$, or $q+1$ in such a way that the weight contribution of that caret pair is reduced.

The types of the carets in the positive tree of a tree-pair diagram can be changed only by adding carets as descendants (see Remark 3.3), which will always increase the length of the element (see Theorem 3.6) and therefore force the resulting element to be outside B_m, so if type change is necessary, we must change the type of some caret in the negative tree. We cannot cancel or change the type of $\wedge_{q+2}$ (or $\wedge_{q+3}$) in the negative tree without moving it counterclockwise, which will prevent $\wedge_q$ in the negative tree from being able to be of type $\mathcal{R}$, and if $\wedge_q$ is any type other than

$\mathcal{R}_{\mathrm{R}}$ or $\mathcal{R}_{\emptyset}$, the weight contributed by the pair $\wedge_q$ will increase, again resulting in an element that is outside B_m.

So we must cancel or change the type of $\wedge_i$ for some $i \in \{1, \ldots, q-1\}$ in the negative tree. If we try to cancel one of these, we must cancel $\wedge_1$ first, because this caret is a descendent of all the other $\wedge_i$ carets in the positive tree, and we cannot change the relative position of carets in a positive tree. If we want to change the type of one of these carets in the negative tree in order to reduce the weight of that caret pair, we must change the type to $\mathcal{R}_{\mathrm{R}}$ or $\mathcal{R}_{\emptyset}$, since these are the only two type changes which will reduce the weight from 2 to 1. But if we change one of these caret types to $\mathcal{R}_{\mathrm{R}}$ or $\mathcal{R}_{\emptyset}$, in order to take $\wedge_q$ from its current position to the root, we will have to move the caret back to a position where it cannot be of type $\mathcal{R}$. So changing the type of one of these carets will not be fruitful, and therefore our only remaining option is to cancel the caret pair $\wedge_1$.

Once we have eliminated $\wedge_1$ from the tree-pair diagram, we have reduced the length of the element enough that we can rotate $\wedge_q$ counterclockwise through the root without leaving B_m, but now our tree-pair diagram does not represent r because we are missing $\wedge_1$, which we had to eliminate in order to perform this rotation in B_m. But we can't immediately go back and add $\wedge_1$ back again, because that will take us back to where we were before eliminating $\wedge_1$, so to ensure the resulting element is in B_m when we add $\wedge_1$ back, we have to cancel or change the type of another nontrivially weighted caret (i.e., one of the carets that had index $2, \ldots, q-1$ or $q+1$ in the original tree-pair diagram (S_-, S_+)).

Since $\wedge_i$ (for $i = 2, \ldots, q-1$ with respect to the index in (S_-, S_+)), if canceled, will have to be added back before $\wedge_1$ can be added back, canceling one of these carets is not an option. And changing the types of any of these carets presents exactly the same problems that prevented us from changing the type of $\wedge_1$ rather than eliminating it. So our only remaining option is to cancel or change the type of $\wedge_{q+2}$ (where $q+2$ refers to its index in the original tree (S_-, S_+)) in the negative tree. Since no type change will reduce the weight of this caret, we must cancel it.

After canceling $\wedge_{q+2}$, to obtain a tree-pair diagram for r from the resulting tree-pair diagram, we will have to add back $\wedge_1$ first (since adding back $\wedge_{q+2}$ will just take us back to where we were before we eliminated it) and then we will need to add back $\wedge_{q+2}$.

So in total we will have to remove two caret pairs from the tree and add two caret pairs to the tree. Each of these removals/additions will contribute a weight of 1 to the length of the path from l to r. To cancel $\wedge_1$, this caret will have to be the child hanging off the rightmost child vertex of the root in the negative tree, which will require $q-1$ clockwise rotations through the root. Then once $\wedge_1$ has been canceled, to cancel $\wedge_{q+2}$, $\wedge_{q+2}$ will have to be the root of the negative tree, which will require q counterclockwise rotations through the root. Then to add back $\wedge_1$, which can only be done when $\wedge_2$ is hanging off the rightmost child vertex of the root in the negative tree, requires q clockwise rotations through the root. And after $\wedge_1$ has been added back, to add $\wedge_{q+2}$ back, we must have $\wedge_{q+1}$ as the root of the negative tree, which will require $q+1$ counterclockwise rotations through the root. Each rotation will add length one to the path, so in total this will give us a path length of $4q + 4 = 2m$.

For example, one geodesic path within B_m from l to r (which does not pass through the identity vertex in Γ) is:

$$x_0^{q-1}x_{n-1}^{-1}x_0^{-q}x_{n-1}x_0^q x_{n-1}x_0^{-(q+1)}x_{n-1}^{-1}$$

Since the sums of the absolute values of the exponents is $4q+4$, this path has length $2m$.

4.1.2. Return to formal proof of the main theorem. Our formal approach to the proof of the main theorem of this paper, while motivated by the intuitive informal motivation described above, will appear somewhat different at first glance. We begin by defining h_r and h_l respectively as the last and first balanced vertices on an arbitrary fixed path γ in B_m from l to r. We recall from our explanation of the intuitive motivation behind the proof that removing $\wedge_1$, then $\wedge_{q+2}$, then adding back $\wedge_1$ followed by $\wedge_{q+2}$ requires us to go from a series of positive vertices to a series of negative vertices to a series of positive vertices to a series of negative vertices as we move along γ. So our definitions of h_r and h_l are a way of marking specific vertices on γ where we encounter the first nonpositive vertex as we move from l and the last nonnegative vertex as we approach r. In fact, we will compute h_r explicitly (which will show that the last step on γ must be to add back $\wedge_{q+2}$). We then need not compute h_l explicitly; rather, we simply show that its minimal tree-pair diagram representative must have a specific form, and that this form will allow us to estimate $d_\Gamma(h_r, h_l)$ closely enough to show that it is at least $m+1$. Then by the triangle inequality and some basic algebra, the fact that $|\gamma| \geq 2m$ will immediately follow.

We begin by proving parts (1) and (2) of Main Theorem 1.

Proof of part (1) of Main Theorem 1. . Since $l = rx_0^2$,

$$\begin{aligned} r^{-1}l &= r^{-1}rx_0^2, \text{ so} \\ d_\Gamma(l,r) &= |r^{-1}l| = 2. \end{aligned}$$

□

Proof of part (2) of Main Theorem 1. . We compute the lengths of r and l:
For r:

$$\begin{aligned} \wedge_0 &= (\mathcal{L}_\emptyset, \mathcal{L}_\emptyset) \text{ so } w(\wedge_0) = 0 \\ \wedge_i &= (\mathcal{L}_\mathrm{L}, \mathcal{M}_\emptyset^{n-1}) \text{ so } w(\wedge_i) = 2 \text{ for } i = 1, \ldots, q \\ \wedge_{q+1} &= (\mathcal{L}_\mathrm{L}, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+1}) = 1 \\ \wedge_{q+2} &= (\mathcal{M}_\emptyset^{n-1}, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+2}) = 1 \\ \wedge_{q+3} &= (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+3}) = 0. \end{aligned}$$

Summing the weights for each of these yields the length of r:
$|r| = 2q + 2 = m$

For l:

$$\begin{aligned}
\wedge_0 &= (\mathcal{L}_\emptyset, \mathcal{L}_\emptyset) \text{ so } w(\wedge_0) = 0 \\
\wedge_i &= (\mathcal{L}_{\mathrm{L}}, \mathcal{M}_\emptyset^{n-1}) \text{ so } w(\wedge_i) = 2 \text{ for } i = 1, \ldots, q-1 \\
\wedge_q &= (\mathcal{R}_{\mathrm{R}}, \mathcal{M}_\emptyset^{n-1}) \text{ so } w(\wedge_q) = 1 \\
\wedge_{q+1} &= (\mathcal{R}_{n-1}, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+1}) = 2 \\
\wedge_{q+2} &= (\mathcal{M}_\emptyset^{n-1}, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+2}) = 1 \\
\wedge_{q+3} &= (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) \text{ so } w(\wedge_{q+3}) = 0.
\end{aligned}$$

Summing the weights for each of these yields the length of l:

$$|l| = 2q + 2 = m \qquad \square$$

The rest of this paper will now be devoted to a proof of part (3) of Main Theorem 1.

We begin with the following lemma, which is originally from Belk and Bux (Lemma 4.1 in [2]) for the case $n = 2$, but which we have generalized to all $n \in \{2, 3, 4, \ldots, \}$:

Lemma 4.1. *In $F(n)$, if there are two vertices h_r and h_l on an arbitrary fixed path γ in B_m between l and r such that $d_\Gamma(h_l, h_r) \geq m + 1$, then $|\gamma| \geq 2m$.*

Proof. By definition,

$$|\gamma| \geq d_\Gamma(l, h_l) + d_\Gamma(h_l, h_r) + d_\Gamma(h_r, r)$$

And by the triangle inequality,

$$d_\Gamma(h_l, h_r) \leq d_\Gamma(h_l, l) + d_\Gamma(l, r) + d_\Gamma(r, h_r)$$

By definition $d_\Gamma(l, r) = 2$. Substituting this into the second inequality and solving each inequality for $d_\Gamma(l, h_l) + d_\Gamma(r, h_r)$ yields the following two equations:

$$|\gamma| - d_\Gamma(h_l, h_r) \geq d_\Gamma(l, h_l) + d_\Gamma(r, h_r) \text{ and } d_\Gamma(h_l, h_r) - 2 \leq d_\Gamma(l, h_l) + d_\Gamma(r, h_r)$$

Putting these together with $d_\Gamma(h_l, h_r) \geq m + 1$ yields:

$$|\gamma| \geq 2d_\Gamma(h_l, h_r) - 2 \geq 2m \qquad \square$$

So according to Lemma 4.1, if we can prove that an arbitrary fixed path γ between l and r which remains in the ball B_m contains two vertices h_r and h_l which are distance $m + 1$ apart, then the main theorem of this paper will immediately follow. We now proceed to define these vertices.

4.2. Finding the vertex h_r. We begin by letting γ denote a fixed arbitrary directed path in Γ from l to r which does not leave B_m. We define h_r to be the last balanced vertex along γ. (This definition is a generalized version of the same definition which Belk and Bux chose for h_r in [2], although our terminology, chosen because it is clearer when using tree-pair diagram representatives, is somewhat different.) We show that such a vertex does exist on γ, and that, in fact, h_r can be computed explicitly. To do this, we begin by showing through explicit computation that any path from l to r which remains in B_m must pass through the vertex $rx_{n-1}x_0$: we first show that all paths beginning at r must pass through rx_{n-1} or leave B_m, then we show that all paths which pass through $rx_{n-1}x_0^{-1}$, $rx_{n-1}x_i^{\pm 1}$ for $i \in \{1, \ldots, n-2\}$, or rx_{n-1}^2 must leave B_m, and at last we show that $h_r = rx_{n-1}x_0$.

TABLE 2. Let $g^{\pm} \in \{x_0, x_1, \ldots, , x_{n-1}\}$ such that rg satisfies the two conditions in Definition 3.4, and hence Theorem 3.5 applies. Then right multiplication by g induces the following type changes to carets in T_-. Here $\wedge_j$ is the caret that changes type in T_-, $\tau_r(\wedge_j)$ denotes the type pair of $\wedge_j$ in the minimal tree-pair diagram representative for r, and $w_r(\wedge_j)$ denotes the weight of that caret pair given its type pair in the minimal tree-pair diagram representative for r. Here $i \in \{1, \ldots, n-2\}$.

g	$\wedge_j$	$\tau_r(\wedge_j)$	$\tau_{rg}(\wedge_j)$	$w_r(\wedge_j)$	$w_{rg}(\wedge_j)$	$\lvert rg\rvert$
x_0	$\wedge_{q+1}$	$(\mathcal{L}_{\mathrm{L}},\mathcal{R}_\emptyset)$	$(\mathcal{R}_{n-1},\mathcal{R}_\emptyset)$	1	2	$m+1$
x_0^{-1}	$\wedge_{q+3}$	$(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$	$(\mathcal{L}_{\mathrm{L}},\mathcal{R}_\emptyset)$	0	1	$m+1$
x_i^{-1}	$\wedge_{q+3}$	$(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$	$(\mathcal{M}^i_\emptyset,\mathcal{R}_\emptyset)$	0	1	$m+1$
x_{n-1}	$\wedge_{q+2}$	$(\mathcal{M}^i_\emptyset,\mathcal{R}_\emptyset)$	$(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$	1	0	$m-1$

Lemma 4.2. *Any path in Γ which begins at r must either pass through the vertex rx_{n-1} or leave B_m.*

Proof. We have $|r| = m$. We begin by computing the length of rg where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$.

When $g = x_i$ for $i \in \{1, \ldots, n-2\}$ or x_{n-1}^{-1}, rg does not satisfy the subtree condition of Definition 3.4, so $|rg| > m$ in these cases by Theorem 3.6. So we need only check the remaining cases of g. In these cases, both conditions of Definition 3.4 are satisfied. When these conditions are satisfied, only one caret changes type in the negative tree (Theorem 3.5); Table 2 outlines how this type change affects the length of the resulting element in the remaining cases for g.

Because rx_0, rx_0^{-1}, rx_i and rx_i^{-1} $\forall i \in \{1, \ldots, n-2\}$, and rx_{n-1}^{-1} all have length greater than m, we know that any path from l to r which passes through one of these vertices must leave B_m. Therefore, any path from l to r which does not leave B_m must pass through the vertex rx_{n-1}. □

Lemma 4.3. *All vertices in Γ of the form $rx_{n-1}g$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$ are in B_m.*

Proof. Since $|rx_{n-1}| = m-1$, it is obvious that $|rx_{n-1}g| \leq m$ for all g such that $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$, so the explicit calculations which follow are unnecessary for the proof of this lemma. However, since explicit calculation of the length of $rx_{n-1}g$ for all $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$ will be necessary to complete the proof of Lemma 4.4 which is to follow, we include those calculations here. We proceed by considering $|rx_{n-1}g|$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$. We can see the minimal tree-pair diagram representative of rx_{n-1} in Figure 12, and we recall that $|rx_{n-1}| = m-1$.

(1) $rx_{n-1}x_0$: Multiplying rx_{n-1} by x_0 satisfies both conditions of Definition 3.4 and changes $\wedge_{q+1}$ in T_- from type $\mathcal{L}_{\mathrm{L}}$ to type $\mathcal{R}_\emptyset$. This changes the caret pairing from $(\mathcal{L}_{\mathrm{L}},\mathcal{R}_\emptyset)$, which has weight 1, to $(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$, which has weight 0, thereby decreasing the length by one. So $|rx_{n-1}x_0| = m-2$.

(2) All other cases: When $g = x_{n-1}^{-1}$, $rx_{n-1}g = r$, so we need not consider this case. When $g = x_0^{-1}, x_i$ or x_i^{-1} for $i \in \{1, \ldots, n-2\}$, or x_{n-1}, it does not satisfy the subtree condition of Definition 3.4, so by Theorem 3.6,

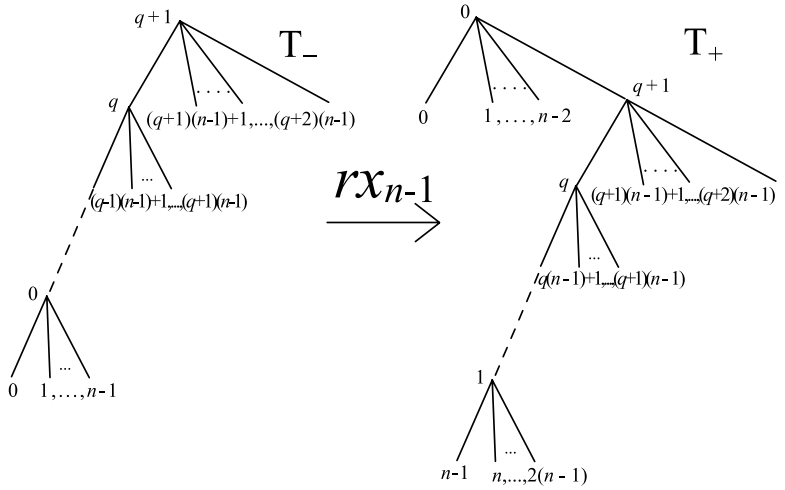

FIGURE 12. Minimal tree-pair diagram representative for rx_{n-1} in $F(n)$.

$|rx_{n-1}g| > m-1$. However, since we are multiplying by a generator, the length can only increase by at most 1, so we will have $|rx_{n-1}g| = m$ in these cases.

Since all elements of the form $rx_{n-1}g$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$ are of length less than or equal to m, they all remain inside B_m. □

The following lemma is a generalization of a similar lemma (Lemma 4.3 in [2]) used by Belk and Bux in their proof that $F(2)$ is not minimally almost convex:

Lemma 4.4. *Any path in* Γ *which begins at* r *must either pass through the vertex* $rx_{n-1}x_0$ *or leave* B_m.

Proof. Because all elements of the form $rx_{n-1}g$ (where g is one of the standard finite generators or its inverse) are inside the ball B_m, in order to show that all paths beginning at r which remain in B_m must pass through the vertex $rx_{n-1}x_0$, we must proceed by considering the length of all elements of the form $rx_{n-1}g_1g_2$ where $g_1^{\pm 1}, g_2^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$.

(1) $rx_{n-1}x_0^{-1}g$: We look at the minimal tree-pair diagram of $rx_{n-1}x_0^{-1}$, which we can see in Figure 13 and we consider the length of $rx_{n-1}x_0^{-1}g$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$. $rx_{n-1}x_0^{-1}g$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$ does not satisfy the subtree condition of Definition 3.4 unless $g = x_0$, in which case $rx_{n-1}x_0^{-1}g$ will reduce to rx_{n-1}. So we know from Theorem 3.6 that $|rx_{n-1}x_0^{-1}g| > m$ whenever $rx_{n-1}x_0^{-1}g \neq rx_{n-1}$. Therefore any path from l to r which remains in B_m cannot pass through $rx_{n-1}x_0^{-1}$.

(2) $rx_{n-1}x_ig\ \forall i \in \{1, \ldots, n-2\}$: Now we look at the minimal tree-pair diagram for $rx_{n-1}x_i\ \forall i \in \{1, \ldots, n-2\}$, which we can see in Figure 14 and consider the length of $rx_{n-1}x_ig$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$. $rx_{n-1}x_ig$ does not satisfy the subtree condition of Theorem 3.4 in the cases when $g = x_j \forall j \in \{1, \ldots, n-1\}$, or x_{n-1}^{-1}, so we know from Theorem 3.6 that $|rx_{n-1}x_ig| > m$ whenever $g = x_j \forall j \in \{1, \ldots, n-1\}$, or x_{n-1}^{-1}.

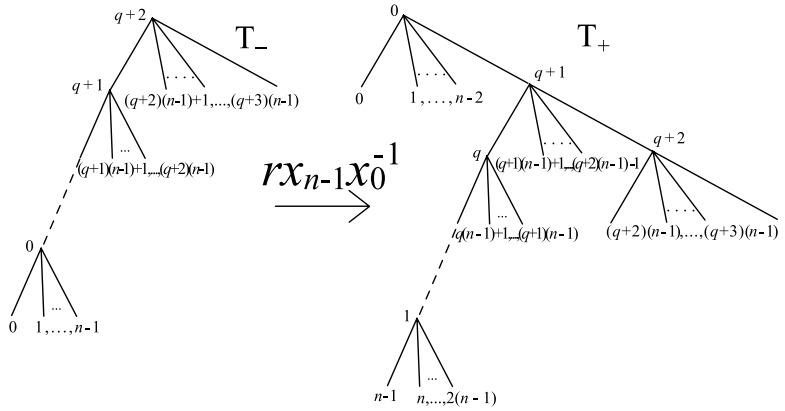

FIGURE 13. Minimal tree-pair representative for $rx_{n-1}x_0^{-1}$ in $F(n)$.

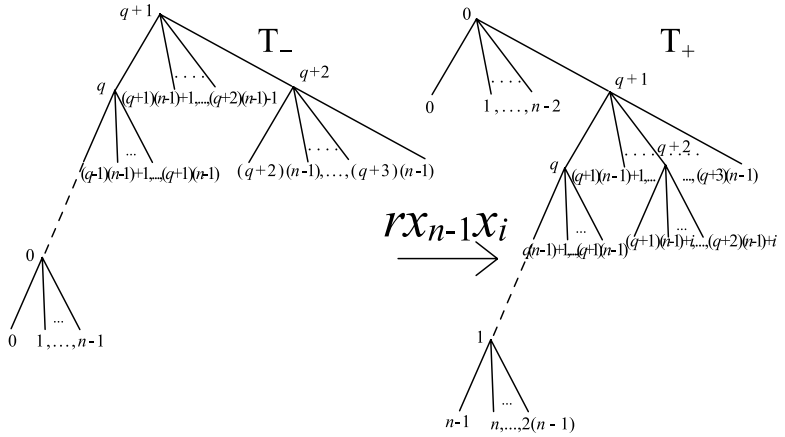

FIGURE 14. Minimal tree-pair diagram representative for $rx_{n-1}x_i$ $\forall i \in \{1, \ldots, n-2\}$ in $F(n)$.

Now all that remains is to check the length of $rx_{n-1}x_i g \; \forall i \in \{1, \ldots, n-2\}$ when $g = x_0, x_0^{-1}$ or x_j^{-1} for $j \in \{1, \ldots, n-2\}$. Since both conditions of Definition 3.4 are met in these cases, all that will change as a result of the multiplication by g is the type of a single caret in the negative tree. We outline these changes and their effect on the length of the element in Table 3.

Therefore, $\forall i \in \{1, \ldots, n-2\}$, any path from l to r which remains in B_m cannot pass through $rx_{n-1}x_i$.

(3) $rx_{n-1}x_i^{-1}g \; \forall i \in \{1, \ldots, n-2\}$: Now we look at the minimal tree-pair diagram for $rx_{n-1}x_i^{-1} \; \forall i \in \{1, \ldots, n-2\}$, which we can see in Figure 15, and we consider the length of $rx_{n-1}x_i^{-1}g$ where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$. $rx_{n-1}x_i^{-1}g$ does not satisfy the subtree condition of Definition 3.4 in the cases when $g = x_0^{-1}$, x_j with $i \neq j \in \{1, \ldots, n-1\}$, or x_j^{-1} with $j \in \{1, \ldots, n-1\}$, so we know from Theorem 3.6 that $|rx_{n-1}x_i^{-1}g| > m$ whenever $g = x_0^{-1}$, x_j with $i \neq j \in \{1, \ldots, n-1\}$, or x_j^{-1} with $j \in \{1, \ldots, n-1\}$.

TABLE 3. Type changes to carets in the negative tree of $rx_{n-1}x_i$ when $rx_{n-1}x_ig$, $g^{\pm} \in \{x_0, x_1, \ldots, , x_{n-1}\}$, satisfies both conditions of Definition 3.4. Here $\wedge_k$ is the caret that changes type in the negative tree of $rx_{n-1}x_i$, $\tau_{rx_{n-1}x_i}(\wedge_k)$ denotes the type pair of $\wedge_k$ in the minimal tree-pair diagram representative for $rx_{n-1}x_i$, and $\Delta w(\wedge_k)$ denotes the change in weight of that caret pair from the minimal tree-pair diagram representative for $rx_{n-1}x_i$ to the minimal tree-pair diagram representative for $rx_{n-1}x_ig$. Here $j \in \{1, \ldots, n-2\}$.

g	$\wedge_k$	$\tau_{rx_{n-1}x_i}(\wedge_k)$	$\tau_{rx_{n-1}x_ig}(\wedge_k)$	$\Delta w(\wedge_k)$	$\lvert rx_{n-1}x_ig\rvert$
x_0	$\wedge_{q+1}$	$(\mathcal{L}_{\mathrm{L}},\mathcal{R}_i)$	$(\mathcal{R}_\emptyset,\mathcal{R}_i)$	$+1$	$m+1$
x_0^{-1}	$\wedge_{q+2}$	$(\mathcal{R}_\emptyset,\mathcal{M}_\emptyset^i)$	$(\mathcal{L}_{\mathrm{L}},\mathcal{M}_\emptyset^i)$	$+1$	$m+1$
x_j^{-1}	$\wedge_{q+2}$	$(\mathcal{R}_\emptyset,\mathcal{M}_\emptyset^i)$	$(\mathcal{M}_\emptyset^j,\mathcal{M}_\emptyset^i)$	$+1$	$m+1$

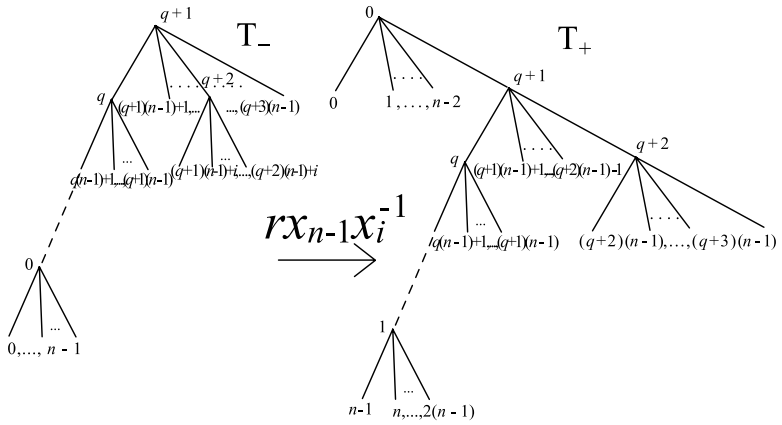

FIGURE 15. Minimal tree-pair diagram representative for

$$rx_{n-1}x_i^{-1} \ \forall i \in \{1, \ldots, n-2\}$$

in $F(n)$.

Now all that remains is to check the length of $rx_{n-1}x_i^{-1}x_0$, $i = 1, \ldots, n-2$. Multiplying $rx_{n-1}x_i^{-1}$ by x_0 satisfies both conditions of Definition 3.4 and changes $\wedge_{q+1}$ in T_- from type $\mathcal{L}_{\mathrm{L}}$ to type $\mathcal{R}_i$. This changes the caret pairing from $(\mathcal{L}_{\mathrm{L}},\mathcal{R}_\emptyset)$, which has weight 1, to $(\mathcal{R}_i,\mathcal{R}_\emptyset)$, which has weight 2, thereby increasing the length by one. So $|rx_{n-1}x_i^{-1}x_0| = m+1$. Therefore, $\forall i \in \{1, \ldots, n-2\}$, any path from l to r which remains in B_m cannot pass through $rx_{n-1}x_i^{-1}$.

(4) rx_{n-1}^2: Now we look at the minimal tree-pair diagram for rx_{n-1}^2, which we can see in Figure 16 and consider the length of rx_{n-1}^2g where $g^{\pm 1} \in \{x_0, x_1, \ldots, , x_{n-1}\}$. rx_{n-1}^2g does not satisfy the subtree condition of Definition 3.4 in the cases when $g = x_i$, $i \in \{1, \ldots, n-1\}$, so we know from Theorem 3.6 that $|rx_{n-1}^2g| > m$ for $g = x_i$, $i \in \{1, \ldots, n-1\}$.

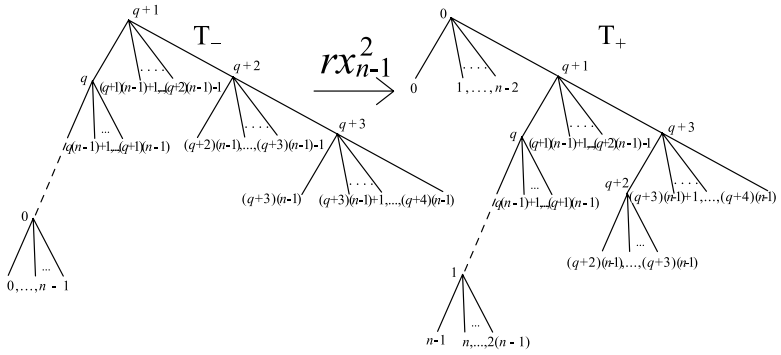

FIGURE 16. Minimal tree-pair diagram representative for rx_{n-1}^2 in $F(n)$.

TABLE 4. Type changes to carets in the negative tree of rx_{n-1}^2 when $rx_{n-1}^2 g$, $g^{\pm} \in \{x_0, x_1, \ldots, , x_{n-1}\}$, satisfies both conditions of Definition 3.4. Here $\wedge_j$ is the caret that changes type in the negative tree of rx_{n-1}^2, $\tau_{rx_{n-1}^2}(\wedge_j)$ denotes the type pair of $\wedge_j$ in the minimal tree-pair diagram representative for rx_{n-1}^2, and $\Delta w(\wedge_j)$ denotes the change in weight of that caret pair from the minimal tree-pair diagram representative for rx_{n-1}^2 to the minimal tree-pair diagram representative for $rx_{n-1}^2 g$. Here $i \in \{1, \ldots, n-2\}$.

g	$\wedge_j$	$\tau_{rx_{n-1}^2}(\wedge_j)$	$\tau_{rx_{n-1}^2 g}(\wedge_j)$	$\Delta w(\wedge_j)$	$\lvert rx_{n-1}^2 g\rvert$
x_0	$\wedge_{q+1}$	$(\mathcal{L}_{\mathrm{L}},\mathcal{R}_{n-1})$	$(\mathcal{R}_{\emptyset},\mathcal{R}_{n-1})$	$+1$	$m+1$
x_0^{-1}	$\wedge_{q+2}$	$(\mathcal{R}_{\emptyset},\mathcal{M}_{\emptyset}^{n-1})$	$(\mathcal{L}_{\mathrm{L}},\mathcal{M}_{\emptyset}^{n-1})$	$+1$	$m+1$
x_i^{-1}	$\wedge_{q+2}$	$(\mathcal{R}_{\emptyset},\mathcal{M}_{\emptyset}^{n-1})$	$(\mathcal{M}_{\emptyset}^{i},\mathcal{M}_{\emptyset}^{n-1})$	$+1$	$m+1$

Now all that remains is to check the length of $rx_{n-1}^2 g$ when $g = x_0, x_0^{-1}$ or x_i^{-1}, $i \in \{1, \ldots, n-2\}$. Since both conditions of Definition 3.4 are met in these cases, Theorem 3.5 applies and we need only outline the effect of the change of the type of a single caret in the negative tree on the length, which we do in Table 4.

Therefore any path from l to r which remains in B_m cannot pass through rx_{n-1}^2.

Because any path from l to r which remains in B_m cannot pass through $rx_{n-1}g$ whenever $g \in \{x_0^{-1}, x_1^{\pm 1}, x_2^{\pm 1}, \ldots, x_{n-1}^{\pm 1}\}$, any path from l to r which remains in B_m must pass through $rx_{n-1}x_0$. □

Corollary 4.5. *Any path γ from l to r which remains in B_m must pass through the vertex $rx_{n-1}x_0$, and this vertex is the last balanced vertex along the arbitrary fixed path γ (i.e., $h_r = rx_{n-1}x_0$).*

Proof. By Lemma 4.4, any minimal length path from l to r must go through the vertex $rx_{n-1}x_0$; then any minimal length path from $rx_{n-1}x_0$ to r must go from the vertex $rx_{n-1}x_0$ to the vertex rx_{n-1}, to the vertex r. By looking at the minimal

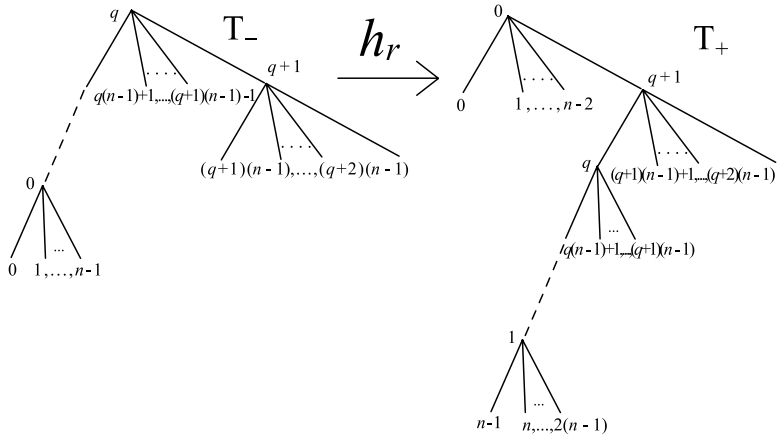

FIGURE 17. Minimal tree-pair diagram representative of h_r in $F(n)$.

tree-pair diagram representatives, it is clear that r and rx_{n-1} are both positive and $rx_{n-1}x_0$ is balanced, so clearly $rx_{n-1}x_0$ is the last balanced vertex along γ. □

So we can conclude that $h_r = rx_{n-1}x_0$. We can see the minimal tree-pair diagram for $h_r = rx_{n-1}x_0$ in Figure 17.

4.3. Finding the vertex h_l. Just as we defined the vertex h_r to be the last balanced vertex on the fixed arbitrary path γ in B_m from l to r, we define h_l to be the first balanced vertex on γ. (Again, this definition is a generalized version of the same definition which Belk and Bux chose for h_l in [2], although our terminology, chosen because it is clearer when using tree-pair diagram representatives, is somewhat different.) Now we want to show that h_l is a distinct vertex from h_r such that $d(h_r, h_l) \geq m+1$, because by Lemma 4.1, this will show that $|\gamma| \geq 2m$. To do this, we first show that any balanced or positive vertex x on the arbitrary path γ in B_m from l to r for which any previous vertex on the subpath from l to x is positive, has a minimal tree-pair diagram representative with a specific form. We then show that any vertex which comes before h_l on the path must be positive and therefore that the minimal tree-pair diagram representative of h_l has this same specific form. This allows us to conclude that $h_l \neq h_r$ and to estimate the distance between h_l and h_r.

Lemma 4.6. *For some $s \in \mathbb{Z}^*$, let $w_j = lg_1 \cdots g_j$ be a vertex on the fixed arbitrary path γ in B_m from l to r such that for all $j \in \{0, 1, \ldots, s-1\}$ w_j is positive (where we use the convention that $w_0 = l$) and w_s is either positive or balanced. (It is clear that such an s always exists because l is positive.) Then the minimal tree-pair diagram representative of w_j has the form given in Figure* 18.

In particular, we note that the last three carets in both the positive and negative trees of the minimal tree-pair diagram representative of w_j have the same type and *relative position* to each other as the last three carets in the respective tree in the minimal tree-pair diagram representative of l (i.e., the subtrees of the negative and positive trees consisting of the carets $\wedge_{q+1}, \wedge_{q+2}, \wedge_{q+3}$ and their successors will be identical to the subtrees of S_- and S_+ of $l = (S_-, S_+)$ consisting of the carets

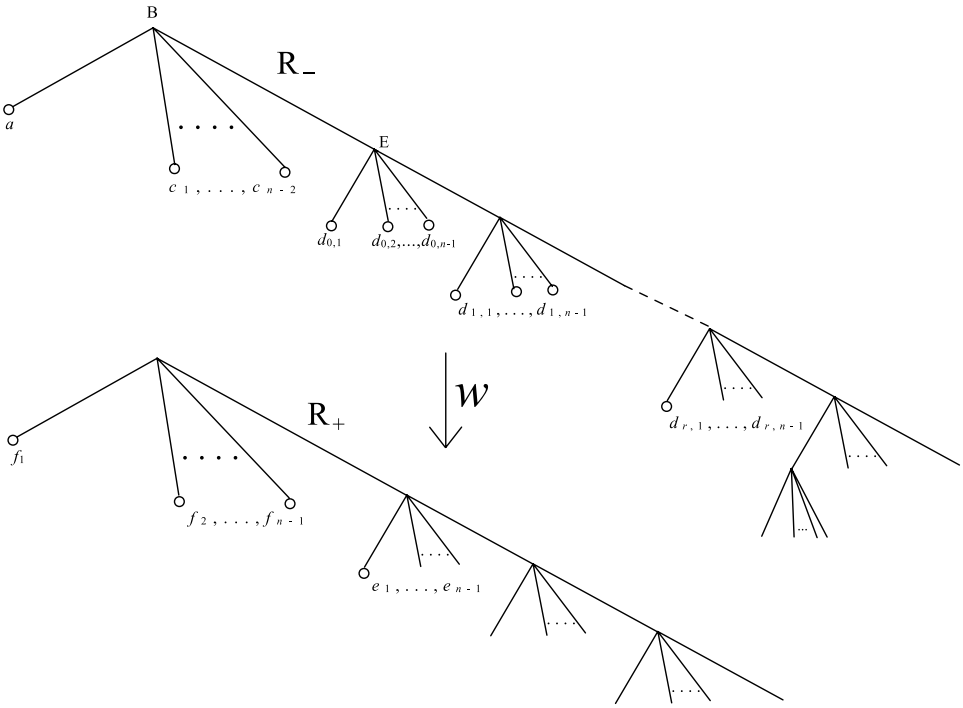

FIGURE 18. Minimal tree-pair diagram representative of arbitrary vertex w_j in Cayley graph Γ of $F(n)$ between l and h_l (including l and h_l) on the arbitrary fixed path γ from l to r in B_m. Here $r \in \mathbb{Z}^*$.

$\wedge_{q+1}, \wedge_{q+2}, \wedge_{q+3}$ and their successors). These carets will have leaves with index numbers which are never less than the index number of either the positive or the negative critical leaf in the tree-pair diagram.

Proof. We prove this by induction. Our hypothesis clearly holds for w_0, since $w_0 = l$.

We proceed to the induction step. We show that for arbitrary $j \in \{1, \ldots, s-1\}$, multiplying w_j on the right by a generator or its inverse will not modify the type or the relative position of the last three carets in the positive or negative trees of the minimal tree-pair diagram representative of w_j. Multiplication by any one of the generators will modify a tree in an arbitrary tree-pair diagram as follows.

If the tree-pair diagram and generator satisfy the subtree condition of Definition 3.4, the following changes will be made to the negative tree of w_j: multiplication by x_0 will change the caret type of the root, multiplication by x_i will change the caret type of the caret hanging off the i^{th} child vertex of the root (where the numbering of child vertices begins with 0 and increases from left to right), multiplication by x_{n-1} will change the caret type of the leftmost child of the rightmost child of the root, and multiplication by x_0^{-1}, x_i^{-1}, or x_{n-1}^{-1} will change the caret type of the rightmost child of the root. So the positive tree in the tree-pair diagram will remain completely unchanged, and since w_j is positive for all $j \in \{1, 2, \ldots, s-1\}$

the caret which changes type will not be one of the last three carets in the tree or a successor of one of the last three carets in the tree, so this multiplication will leave any carets in the negative tree whose leaves all have index numbers greater than or equal to the index number of the positive and negative critical leaves unchanged in type and relative position in the negative tree.

If the tree-pair diagram and generator do not satisfy the subtree condition in Definition 3.4, it will be necessary to add carets before considering the changes which would follow from multiplication by a generator as described in the preceding paragraph. From the definitions of the various caret types, it is obvious that for an arbitrary caret $\wedge_v$ in either tree to have its type changed by the addition of carets in the tree, $\wedge_v$ must be type $\mathcal{R}$ or $\mathcal{M}$ and the added caret must have index number greater than v. But the only places that carets might need to be added in order for multiplication by a generator to take place are as follows (each of these added carets are described by their placement in the negative tree in the tree-pair diagram; a caret is then also added at the leaf with the same index in the positive tree of the tree-pair diagram, which will usually not be in the same location with respect to the root of the positive tree): multiplication by x_0 may require the addition of a caret at the leftmost child of the root, multiplication by x_i may require the addition of a caret at the i^{th} child vertex of the root (where the numbering of child vertices begins with 0 and increases from left to right), multiplication by x_{n-1} may require the addition of a caret at the leftmost child of the rightmost child of the root. Each of these carets will have index number less than or equal to the index number of the caret which contains the positive critical leaf, so even after this addition of carets, the tree-pair diagram for w_j for $j \in \{1, \ldots, s-1\}$ will still have the form given in Figure 18, so our argument from the previous paragraph also follows in these cases. Because w_j is positive for all $j \in \{1, \ldots, s-1\}$, we will never need to add a caret when multiplying by x_0^{-1}, x_i^{-1}, or x_{n-1}^{-1} because we will already have a caret present in the tree which is the rightmost child of the root, and for w_j for $j \in \{1, \ldots, s-1\}$, this caret will not be one of the last three carets in the negative tree or a successor of one of the last three carets in the negative tree. So we can conclude that for any w_j for $j \in \{1, \ldots, s-1\}$, w_{j+1} will also have the form given in Figure 18, and will be either positive or balanced. □

Now if we can establish that there is no negative or balanced vertex on the subpath from l to h_l of the fixed arbitrary path γ from l to r in B_m except the balanced vertex h_l, we will be able to conclude that h_l satisfies the same conditions as the vertex w_s in 4.6 and therefore has minimal tree-pair diagram of the form given in Figure 18. We now proceed to prove this.

Lemma 4.7. *Let x be a vertex on the fixed arbitrary path γ from l to r in B_m which lies on the subpath from l to h_l. If $x \neq h_l$, then x is positive.*

In other words, this lemma states that h_l (which we recall is balanced by definition) is the first vertex on γ which is not positive.

Proof. We note that it is clear from the definition of critical leaf that the indexes of the negative and positive critical leaves will always be a multiple of $n-1$ apart. We must consider when, if ever, as we proceed along γ from l to r, we might encounter a negative vertex before we encounter a balanced vertex.

For this proof, we will use the following convention: let $y_j = lg_1 \cdots g_j$ (where $j \in \{0, \ldots, t\}$ for some $t \in \mathbb{N}$) be a vertex on the subpath of γ which begins with l and ends with h_l such that $l = y_0$ by convention, $h_l = y_t$, and for all $i, k \in \{0, \ldots, t\}$, y_i comes before y_k on γ if and only if $i < k$ (i.e., $l = y_0 \to y_1 \to y_2 \to \cdots \to y_t = h_l$ is exactly the subpath of γ beginning with l and ending with h_l).

We know from Lemma 4.6 that there exists $s \in \mathbb{N}$, $s \leq t$ such that $y_j = lg_1 \cdots g_j$ for all $j \in \{0, \ldots, s-1\}$ is positive and y_s is either positive or balanced. We choose the largest possible value of s for which this is true; in other words, we choose s so that y_{s+1} is not positive. Our task now is to consider whether $s = t$ or whether it is possible that $s < t$. We also recall that y_j for all $j \in \{0, \ldots, s\}$ has a minimal tree-pair diagram representative of the form given in Figure 18. We proceed by considering what the effect is on the minimal tree-pair diagram representative for y_s when we multiply it by a generator on the right.

First we consider the effect of adding a caret to the tree-pair diagram representative for y_s, which will be required before we can multiply y_s by a generator in any case in which y_s and the generator do not satisfy the subtree condition of Definition 3.4. For the following cases, we let $c_\pm$ be the index of the positive/negative critical leaf of the tree-pair diagram representative of y_s.

(1) If we add a caret on a leaf with index less than or equal to c_-, then the new index of the negative critical leaf will be $c_- + (n-1)$ and the new index of the positive critical leaf will be $c_+ + (n-1)$. Because $(c_+ + (n-1)) - (c_- + (n-1)) = c_+ - c_-$ we can see that the relative positions of the positive and negative critical leaves do not change with the addition of this caret.
(2) If we add a caret on a leaf with index i such that $c_- < i \leq c_+$, then the index of the negative critical leaf remains c_- and the index of the positive critical leaf becomes $c_+ + (n-1)$. Since $(c_+ + (n-1)) - c_- = c_+ - c_- + (n-1)$, the relative difference between the positive and negative critical leaves increases by $n-1$, so the resulting tree-pair diagram is still positive.
(3) If we add a caret on a leaf with index greater than c_+, both the index of the positive and negative critical leaves will remain unchanged.

So the act of adding carets alone to the minimal tree-pair diagram of y_s will not cause the resulting tree-pair diagram to become negative or balanced.

Now we consider the movement of carets in the tree-pair diagram of y_s that can be induced by multiplication by a generator on the right after any carets have been added as needed, and we explore when this movement will produce a positive or balanced tree-pair diagram as a result. For simplicity in this section, we will let $c_\pm(g)$ denote the index of the positive/negative critical leaf of the (possibly nonminimal) tree-pair diagram representative of y_s after any carets have been added as needed to the tree-pair diagram so that multiplication by g can be performed.

To proceed, we will consider Figure 19, which is the tree-pair diagram of y_s once any carets have been added as needed to satisfy the subtree condition of Definition 3.4. We note that by Lemma 4.6, the index of the rightmost leaf of subtree $d_{r,1}$ and the index of the rightmost leaf of subtree e_1 must be the same. We will use the convention that the subtree $d_{0,1}$ is the subtree consisting of $\wedge_D$ and its descendants. Since multiplication by a generator will only change the structure of the negative tree (because we have already added any carets to both trees that will be needed for the multiplication), when multiplying by each generator, we know

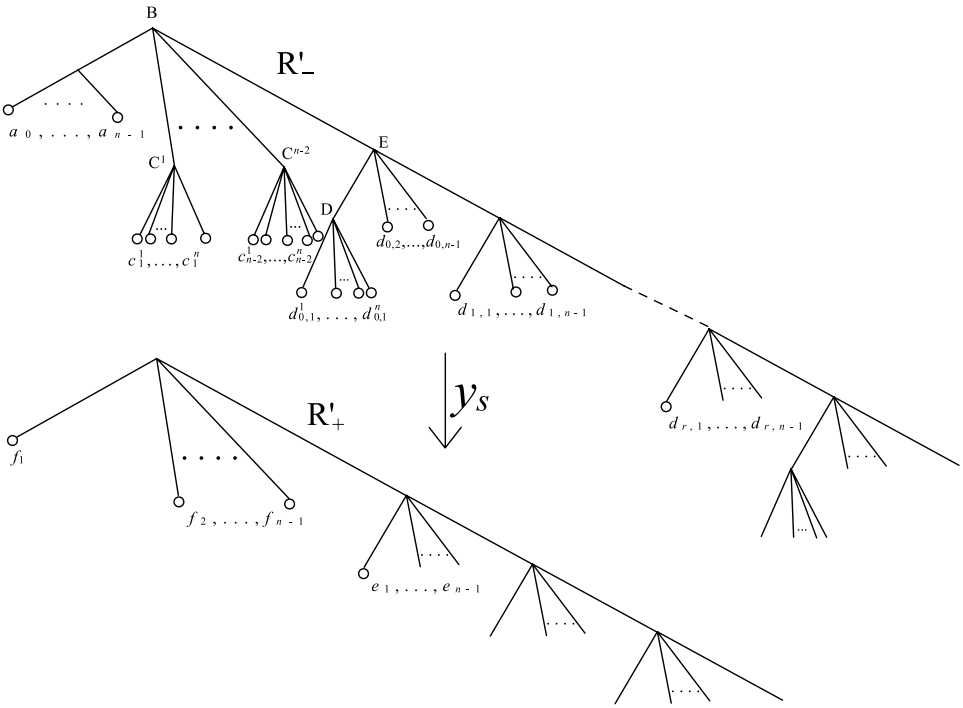

FIGURE 19. A (possibly nonminimal) tree-pair diagram representative of y_s in $F(n)$ after carets have been added as needed so that R'_- satisfies the subtree condition of Definition 3.4 for a given g such that $g^{\pm} \in \{x_0, x_1, \ldots, , x_{n-1}\}$.

that the index of the positive critical leaf will remain $c_+(g)$, and only the index of the negative critical leaf will change with the multiplication. We also note that in Figure 19, the right foot will be the rightmost leaf in the subtree labeled c_1.

Now we consider the effect of multiplying y_s by the following generators on the right:

(1) Multiplication by x_0: The negative critical leaf will become the rightmost leaf in subtree a_{n-1}, which clearly has index less than $c_-(x_0)$, so the tree pair diagram resulting from multiplication by x_0 will still be positive.
(2) Multiplication by x_0^{-1}: The negative critical leaf will become the rightmost leaf in subtree $d_{1,1}$; if $r = 1$ then the index of the rightmost leaf in $d_{1,1}$ will equal the index of the rightmost leaf in e_1, and then the resulting tree-pair diagram will be balanced. If, however, $r > 1$, then the index of the negative critical leaf will remain lower than the index of the positive critical leaf, and the tree-pair diagram resulting from multiplication by x_0^{-1} will still be positive.
(3) Multiplication by x_i for some $i \in \{1, \ldots, n-2\}$: The negative critical leaf will become the rightmost leaf in subtree c_i^{n-i}, which clearly has index less than $c_-(x_i)$, so the tree-pair diagram resulting from multiplication by x_i will still be positive.

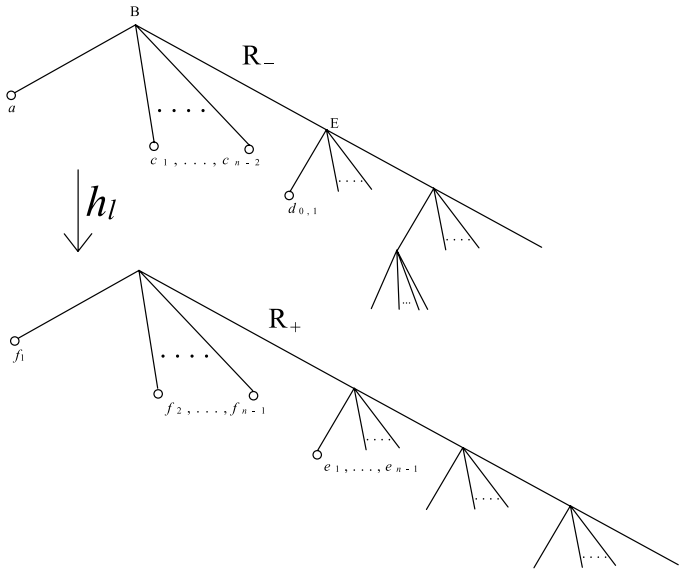

FIGURE 20. Minimal tree-pair diagram representative of h_l in $F(n)$.

(4) Multiplication by x_i^{-1} for some $i \in \{1, \ldots, n-2\}$: The negative critical leaf will become the rightmost leaf in subtree $d_{1,1}$, and the same consequences will hold as in the case of multiplication by x_0^{-1}.
(5) Multiplication by x_{n-1}: The negative critical leaf will become the rightmost leaf in subtree $d_{0,1}^1$, which clearly has index less than $c_-(x_{n-1})$, so the tree-pair diagram resulting from multiplication by x_{n-1} will still be positive.
(6) Multiplication by x_{n-1}^{-1}: The negative critical leaf will become the rightmost leaf of subtree $d_{1,1}$, and the same consequences will hold as in the case of multiplication by x_0^{-1}.

And finally, if any of the tree-pair diagrams resulting from the multiplication enumerated above are not minimal, the removal of exposed caret pairs will not make the resulting tree-pair diagram negative for the same reasons that the addition of caret pairs could not result in a negative tree-pair diagram. So multiplication of y_s by a generator (or its inverse) g on the right will never result in a negative product $y_s g$. □

Corollary 4.8. *h_l has a minimal tree-pair diagram representative of the form given in Figure* 20*.*

Proof. If y_j and s are as defined in the proof of Lemma 4.7, then by Lemma 4.7 and the definition of h_l, $y_s = h_l$. This means that there are no negative vertices on the subpath of the fixed arbitrary path γ from l to r in B_m between l and h_l, and by Lemma 4.6 this means that all vertices on the subpath of γ from l to h_l have the form given in Figure 18. But the only tree-pair diagram of the form which will be balanced is of the form given in Figure 20. □

Corollary 4.9. $h_l \neq h_r$.

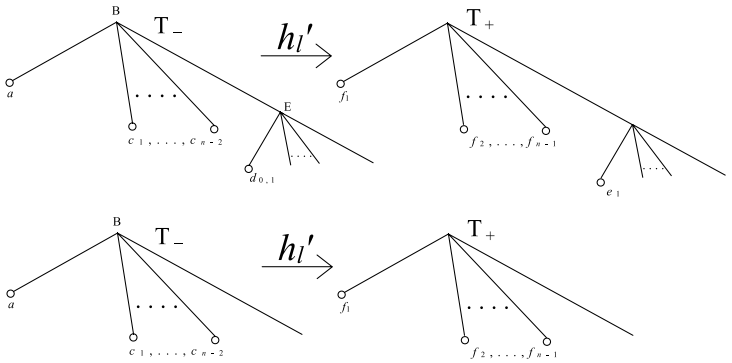

FIGURE 21. The two possible forms of the minimal tree-pair diagram representative of h_l' in $F(n)$: if subtrees $d_{0,1}$ and e_1 were both empty in the minimal tree-pair diagram representative of h_l (see Figure 20), then the minimal tree-pair diagram representative of h_l' will have the form given in the bottom figure; otherwise it will have the form given by the top figure. By Lemma 4.11, the total number of caret pairs in all the labeled subtrees of either of these tree-pair diagrams (i.e., not counting $\wedge_B$ or $\wedge_E$, if it exists) must be strictly less than q.

Proof. This is obvious by looking at the form of the minimal tree-pair diagram representative of h_l as given in Figure 20 and the form of the minimal tree-pair diagram representative of h_r as given in Figure 17. □

4.4. Finding $d(h_r, h_l)$. Now we calculate the distance between h_l and h_r; we show that $d_\Gamma(h_l, h_r) \geq m + 1$. In order to show that $d_\Gamma(h_l, h_r) \geq m + 1$, we will first define a new element, $h_l' = h_l x_0^{-1} x_{n-1}^{-1} x_0$ (which is a generalized version of the same definition which Belk and Bux chose for h_l' in [2]) and describe its minimal tree-pair diagram representative. Then we will show that $|h_r^{-1} h_l'| \geq m - 2$ and $|h_r^{-1} h_l| = |h_r^{-1} h_l'| + 3$, which will immediately yield $d_\Gamma(h_l, h_r) \geq m + 1$.

Definition 4.10 (h_l')**.** We define $h_l' = h_l x_0^{-1} x_{n-1}^{-1} x_0$. We can see the minimal tree-pair diagram representative for h_l' in Figure 21.

Lemma 4.11. *If we let v equal the number of caret pairs in the minimal tree-pair diagram representative for h_l', excluding $\wedge_B$ and $\wedge_E$ (if it exists), then $v < q$. The total number of caret pairs in the minimal tree-pair diagram representative for h_l is strictly less than $q + 4$. (See Figures* 20 *and* 21.)

Proof. $h_l \in B_m$, so we must have $|h_l| \leq m = 2q + 2$. When we go from h_l to h_l', only the last three caret pairs in the minimal tree-pair diagram representative change type or are canceled: if we supposed that the index of the last caret pair in the minimal tree-pair diagram of h_l is t, then the differences in the weight of the carets $\wedge_{t-2}, \wedge_{t-1}, \wedge_t$ is described in Table 5.

So clearly we will have $|h_l'| = |h_l| - 3 \leq 2q - 1$. All carets in the minimal tree-pair diagram representative of h_l' (except $\wedge_E$, if it exists — see Figure 21) will be of type $\mathcal{L}$ or $\mathcal{M}$. If we look at Table 1, we can see that any caret type pair except ($\mathcal{L}_\emptyset$,$\mathcal{L}_\emptyset$)

TABLE 5. Differences between the carets $\wedge_{t-2}, \wedge_{t-1}, \wedge_t$ in the minimal tree-pair diagram of h_l and the minimal tree-pair diagram of h'_l. Here $\tau_{h_l}(\wedge_i)$ denotes the type pair of $\wedge_i$ in the minimal tree-pair diagram representative for h_l, and $w_{h_l}(\wedge_i)$ denotes the weight of that caret pair in the minimal tree-pair diagram representative of h_l. Here "d.n.e." stands for "does not exist."

$\wedge_i$	$\tau_{h_l}(\wedge_i)$	$\tau_{h'_l}(\wedge_i)$	$w_{h_l}(\wedge_i)$	$w_{h'_l}(\wedge_i)$
$\wedge_{t-2}$	$(\mathcal{R}_{n-1},\mathcal{R}_\emptyset)$	$(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$	2	0
$\wedge_{t-1}$	$(\mathcal{M}_\emptyset^{n-1},\mathcal{R}_\emptyset)$	d.n.e.	1	d.n.e.
$\wedge_t$	$(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$	d.n.e.	0	d.n.e.

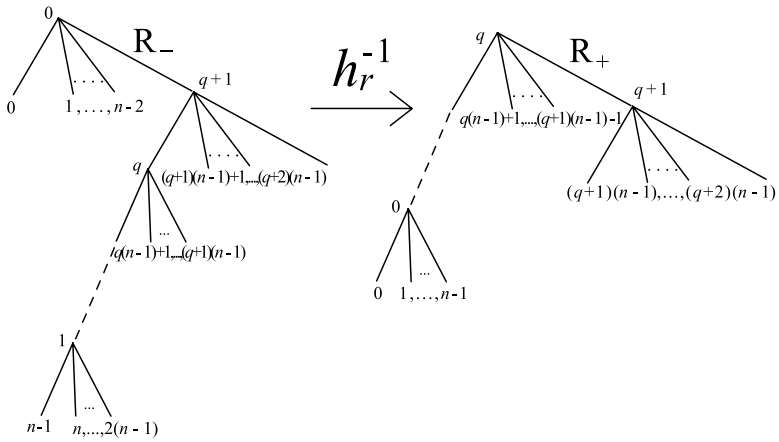

FIGURE 22. Minimal tree-pair diagram representative of h_r^{-1} in $F(n)$.

which does not contain any carets of type $\mathcal{R}$ must have weight 2 or more. So there will be one caret in the tree-pair diagram of h'_l which will be of type $(\mathcal{L}_\emptyset,\mathcal{L}_\emptyset)$ which will not contribute anything to the length, and the caret $\wedge_E$ if it exists will not contribute anything to the length, but all other carets must contribute a weight of at least 2 to the length. So $|h'_l| \geq 2v$ and therefore $v \leq q - \frac{1}{2}$, which implies that $v < q$. Since the number of caret pairs in the minimal tree-pair diagram representative of h_l is 4 more than v, we can conclude that the total number of caret pairs in the minimal tree-pair diagram representative for h_l is strictly less than $q + 4$. □

Now we show that $|h_r^{-1}h'_l| \geq m-2$. First we give the minimal tree-pair diagram representative for $h_r^{-1}h'_l$, and then we use this tree-pair diagram to calculate a bound for $|h_r^{-1}h'_l|$.

Lemma 4.12. *We let $h'_l = (T_-, T_+)$ and $h_r^{-1} = (R_-, R_+)$. We let (T'_-, T'_+) and (R'_-, R'_+) denote (T_-, T_+) and (R_-, R_+) respectively after all necessary carets have been added so that the product $h_r^{-1}h'_l$ can be computed. Then Figure* 23 *is a (possibly nonminimal) tree-pair diagram representative of $h_r^{-1}h'_l$ and:*

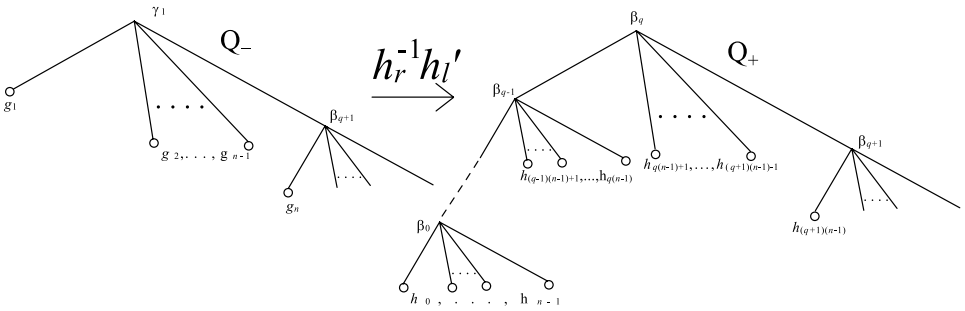

FIGURE 23. Possibly nonminimal (see Lemma 4.12) tree-pair diagram representative of $h_r^{-1}h_l'$ in $F(n)$. This will be minimal except when subtrees g_n and $h_{(q+1)(n-1)}$ are both empty, in which case $\wedge_{\beta_{q+1}}$ can be deleted from both trees and the resulting tree-pair diagram will be minimal.

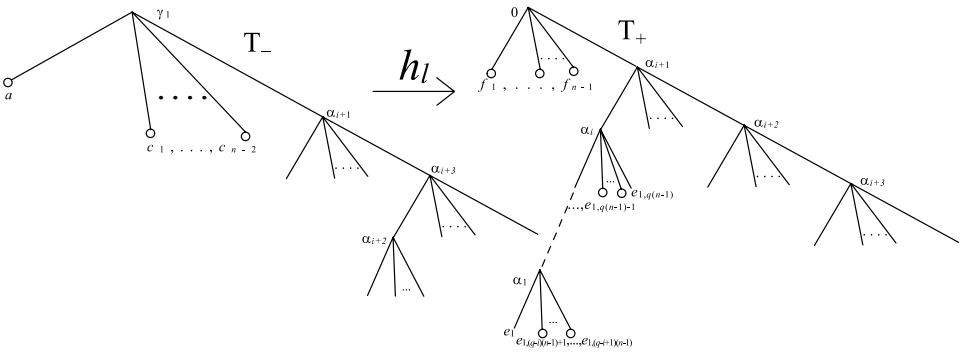

FIGURE 24. One possible form of the minimal tree-pair diagram representative of h_l in $F(n)$. Here $i \in \{1, \ldots, q-1\}$. (We cannot have $i = q$ because the number of caret pairs in this tree-pair diagram must be strictly less than $q + 4$ by Lemma 4.11. We note also that the total number of carets in all of the subtrees $e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)-1}$ must be strictly less than $q - i$.)

(1) *If the minimal tree-pair diagram representative of h_l can be written in the form given in Figure* 24*, then the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ is of the form given in Figure* 25*.*
(2) *Otherwise, Figure* 23 *is the minimal tree-pair diagram representative of $h_r^{-1}h_l'$.*

Proof. We begin by proving (1). Then we show that $h_r^{-1}h_l'$ must always have a (possibly nonminimal) tree-pair diagram representative which can be written in the form given in Figure 23. Then we prove (2).

The proof of (1) is a straightforward calculation. The minimal tree-pair diagram representative of h_l' will be the tree-pair diagram given in Figure 24 with the last two carets in each tree removed. Then in order to compose $h_r^{-1}h_l'$, we must add a string of $q - i$ carets of type $\mathcal{M}_{\emptyset}^{n-1}$ to the leaf with index number e_1 in T_+ and likewise to the leaf with index number e_1 in T_-. Then we must add the subtrees

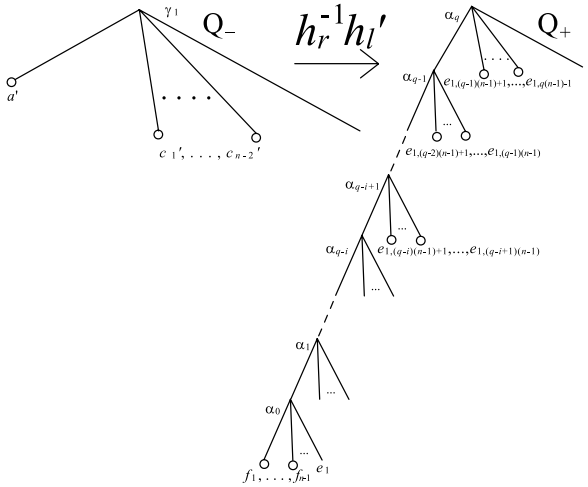

FIGURE 25. One possible form of the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ in $F(n)$. This form is obtained when the minimal tree-pair diagram of h_l has the form given in Figure 24. Here $a', c_1', \ldots, c_{n-2}'$ are obtained from the subtrees $q, c_1, \ldots, c_{n-2}$ of h_l in Figure 24 by possibly adding carets to those subtrees.

$f_1, \ldots, f_{n-1}$ in T_+ to the leaves with index numbers $0, \ldots, n-2$ respectively in R_- and the subtrees $e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)-1}$ in T_+ to the leaves with index numbers $(q-i+1)(n-1)+1, \ldots, (q+1)(n-1)-1$ respectively in R_-. This will make T_+' identical to R_-'. Taking (T_-', R_+') as the tree-pair diagram representative of $h_r^{-1}h_l'$, we see that the caret pair with index number α_{q+1} cancels, resulting in the minimal tree-pair diagram representative given in Figure 25. Because Figure 25 can be put into the form given in Figure 23 by adding one caret to the last leaf in each tree, we can see that in this case, Figure 23 is a (nonminimal) tree-pair diagram representative of $h_r^{-1}h_l'$.

Now we show that $h_r^{-1}h_l'$ must always have a (possibly nonminimal) tree-pair diagram representative which can be written in the form given in Figure 23. We begin by considering the possible forms of the minimal tree-pair diagram representative of h_l as depicted in Figure 20. Either the subtree e_1 is empty or it is not. If e_1 is empty, then h_l can be written in the form given in Figure 26. Then the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ is of the form given in Figure 27. This tree-pair diagram is clearly in the same form as Figure 23.

If subtree e_1 is not empty and the minimal tree-pair diagram of h_l cannot be written in the form given in Figure 24, then it can be written in the form given in Figure 28. Then the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ is of the form given in Figure 29. This tree-pair diagram is also clearly in the same form as Figure 23.

To prove (2), we show that when none of the conditions of (1) are met, there does not exist a pair of exposed carets in the tree-pair diagram given in Figure 23.

If we can consider all possible exposed carets in Q_- and show that none of them can possibly pair up with an exposed caret in Q_+, then this will be sufficient to

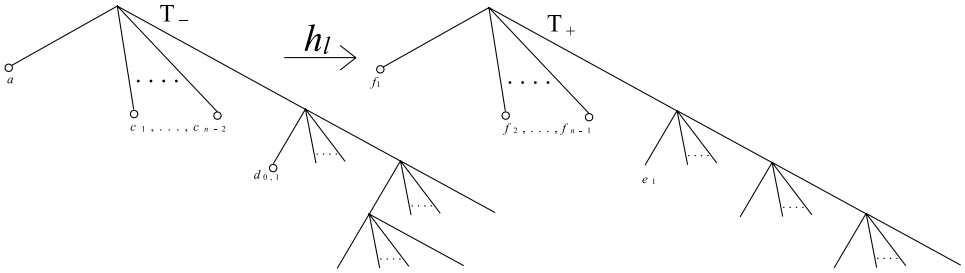

FIGURE 26. One possible form of the minimal tree-pair diagram representative of h_l in $F(n)$.

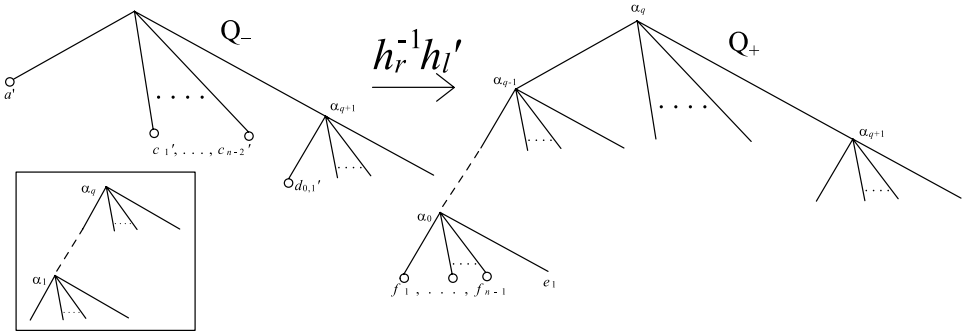

FIGURE 27. The minimal tree-pair diagram representative of $h_r^{-1}h_l'$ when the minimal tree-pair diagram representative of h_l is in the form given by Figure 26. The subtrees $f_1, \ldots, f_{n-1}$ are taken from Figure 26, and the subtrees $a', c_1', \ldots, c_{n-2}', d_{0,1}'$ are formed from the subtrees $a, c_1, \ldots, c_{n-2}, d_{0,1}$ in Figure 26 by possibly adding carets. If the subtree $d_{0,1}$ in Figure 26 was empty, then the subtree $d_{0,1}'$ has the form given in the box in the lower left corner.

show that the diagram is minimal. We note that the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ in (2) will always be able to be written in the form given in either Figure 27 or 29. For the duration of this proof, we use the convention that $h_l = (T_-, T_+)$ and $h_r^{-1}h_l' = (Q_-, Q_+)$.

So we begin by considering all possible exposed carets in Q_-:

(a) $\wedge_{\beta_{q+1}}$: In Q_-, $\wedge_{\beta_{q+1}}$ will only be exposed if the subtree g_n is empty, which implies that the subtree $d_{0,1}'$ in Figure 27 or 29 is empty; but this is possible only in Figure 29. So in order for g_n to be empty in Q_-, h_l must have the form given in Figure 28. In Q_+, $\wedge_{\beta_{q+1}}$ will be exposed only if the last caret in the positive tree of the tree-pair diagram given in Figure 27 or 29 is exposed, but since $e_{1,q(n-1)}$ will always be nonempty in Figure 29 if $d_{0,1}$ in T_- is empty and $d_{0,1}$ in T_- will always be empty whenever $d_{0,1}'$ in Q_- is empty, this is only possible in Figure 27. So h_l must have the form given in Figure 26. This is a contradiction, so in (2), the caret pair $\wedge\beta_{q+1}$ in Figure 23 will never cancel.

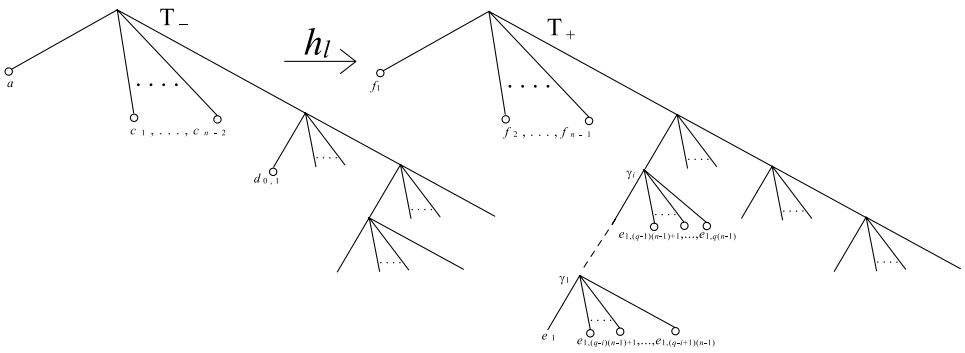

FIGURE 28. One possible form of the minimal tree-pair diagram representative of h_l in $F(n)$. Here the subtree $e_{1,q(n-1)}$ or $d_{0,1}$ is nonempty, and $i \in \{1, \ldots, q-1\}$. (We cannot have $i = q$ by Lemma 4.11. This lemma also tells us that the total number of all carets in all the subtrees $f_1, \ldots, f_{n-1}, e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)}$ must be strictly less than $q - i$.)

(b) $\wedge_{\beta_i}$ for $i = 1, \ldots, q$: The carets in Q_+ with index numbers $\beta_1, \ldots, \beta_q$ each have a child caret, so none of them can be exposed, and therefore none of the caret pairs $\wedge_{\beta_j}$ for $j \in \{1, \ldots, q\}$ will cancel unless one of their descendants cancels first.

(c) A caret descended from one of the carets $\wedge_{\beta_1}, \ldots, \wedge_{\beta_q}$ in Q_- which itself does not have index number β_i for any $i \in \{1, \ldots, q\}$:

The carets $\wedge_{\beta_0}, \ldots, \wedge_{\beta_q}$ correspond to the carets $\wedge_{\alpha_0}, \ldots, \wedge_{\alpha_q}$ respectively in Figure 27 or 29. First we consider whether $\wedge_{\alpha_0}$ in the positive tree of Figure 27 or 29, which is a descendent of $\wedge_{\alpha_i}$ for $i \in \{1, \ldots, q\}$, could be exposed. We note that the minimal tree-pair diagram representative of h'_l will be the same as Figure 26 or 28 with the last 2 carets removed from each tree. Then, in order to compose $h_r^{-1} h'_l$, it will be necessary to add at least one caret to the leaf numbered e_1 in both the positive and negative trees of the minimal tree-pair diagram representative of h'_l. Once all of the necessary carets are added, the leaves with index number e_1 in the minimal tree-pair diagram of h'_l will be the leftmost child vertex of an exposed caret. This exposed caret with leftmost leaf index e_1 in the negative tree of the minimal tree-pair diagram for h'_l will still be exposed in the negative tree of Figure 27 or 29, but we notice that the leaf with index e_1 is now the rightmost child vertex of $\wedge_{\alpha_0}$ in the positive tree of Figure 27 or 29. So we can see that it will not be possible for there to be an exposed caret in the negative tree of Figure 27 or 29 with rightmost leaf having index e_1, and therefore $\wedge_{\alpha_0}$ will not be exposed in the negative tree of 27 or 29.

Now we consider whether there can be any exposed carets in the subtrees $f_1, \ldots, f_{n-1}, e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)}$ that will cancel in Figure 27 or 29. Let $\wedge_j$ be an exposed caret in one of these subtrees. Then $\wedge_j$ in the negative tree will be in one of the subtrees $a', c'_1, \ldots, c'_{n-2}, d'_{0,1}$, and if it is exposed in one of these subtrees, then it must have been exposed in the corresponding subtree $a, c_1, \ldots, c_{n-1}, d_{0,1}$ in Figure 26 or 28; but since the tree-pair diagram

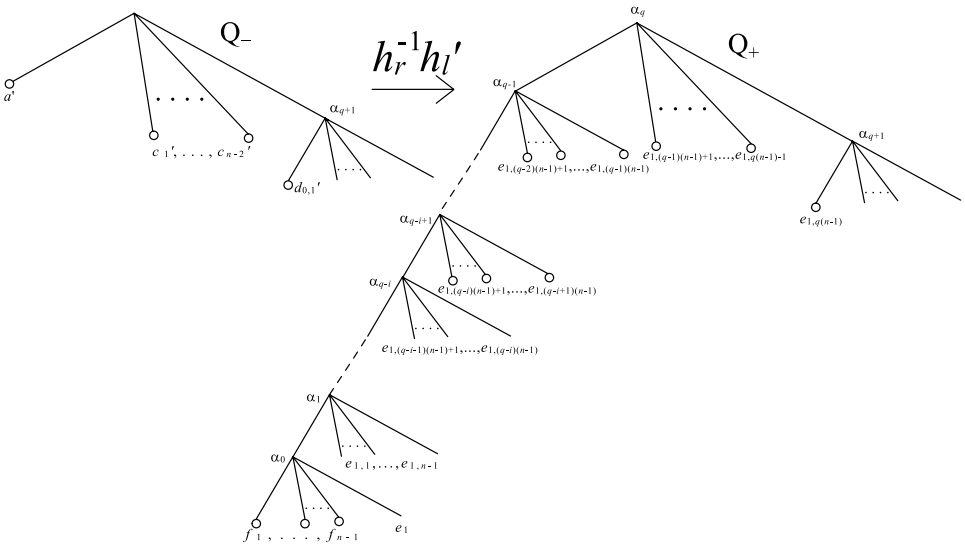

FIGURE 29. The minimal tree-pair diagram representative of $h_r^{-1}h_l'$ when the minimal tree-pair diagram representative of h_l is in the form given by Figure 28. Here the subtrees

$$f_1, \ldots, f_{n-1}, e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)}$$

are taken from Figure 28 (so $e_{1,q(n-1)}$ will always be nonempty if $d_{0,1}$ in T_- is empty and the total number of all carets in all the subtrees

$$f_1, \ldots, f_{n-1}, e_{1,(q-i)(n-1)+1}, \ldots, e_{1,q(n-1)}$$

must be strictly less than $q-i$), and and the subtrees

$$a', c_1', \ldots, c_{n-2}', d_{0,1}'$$

are formed from the subtrees $a, c_1, \ldots, c_{n-2}, d_{0,1}$ in Figure 28 by possibly adding carets.

given in Figure 26 or 28 is already minimal, this is not possible, so none of these caret pairs can cancel in Figure 27 or 29.

(d) Carets in the subtrees $g_1, \ldots, g_n$ in Q_- (other than $\wedge_{\beta_1}, \ldots, \wedge_{\beta_q}$ or descendants of these carets):

The subtrees $g_1, \ldots, g_n$ are exactly the subtrees $a', c_1', \ldots, c_{n-2}', d_{0,1}'$ in Figure 27 or 29, which are formed by taking the subtrees $a, c_1, \ldots, c_{n-2}$, $d_{0,1}$ in Figure 26 or 28 respectively and adding any carets which will have the same index numbers as the carets $\wedge_{\beta_0}, \ldots, \wedge_{\beta_q}$ in Figure 23. So any exposed carets in the subtrees $g_1, \ldots, g_n$ which are not $\wedge_{\beta_0}, \ldots, \wedge_{\beta_q}$ will have been present and exposed in $a, c_1, \ldots, c_{n-2}, d_{0,1}$ respectively in Figure 26 or 28. But all carets in $a, c_1, \ldots, c_{n-2}, d_{0,1}$ were paired with carets in at least one of the subtrees $f_1, \ldots, f_{n-1}, e_{1,1}, \ldots, e_{1,q(n-1)}$ in Figure 26 or 28, and none of these subtrees have been modified from their form in (T_-, T_+). So if a given caret pair with one caret in one of the subtrees $a_0', \ldots, a_{n-2}', d_{0,1}'$ of Figure 27 or 29 is exposed, then that same caret pair would have been

exposed in (T_-, T_+); but since (T_-, T_+) is minimal, this is not possible, and therefore none of the carets in the subtrees $a'_0, \ldots, a'_{n-2}, d'_{0,1}$ in Q_- can be canceled. □

The following lemma is also related to a lemma (Lemma 4.4 in [2]) of Belk and Bux (in Belk and Bux, this is specifically proven for what they define as "left sided elements"):

Lemma 4.13. *If we let $N(w)$ denote the number of carets in the minimal tree-pair diagram representative of an element w of $F(n)$, then $|h_r^{-1}h'_l| \geq 2(N(h_r^{-1}h'_l) - 2)$.*

Proof. We consider all the possible types of caret pairings in the minimal tree-pair diagram representative of $h_r^{-1}h'_l$ (see Figure 23):

(1) The first caret pair must have type pair ($\mathcal{L}_\emptyset$,$\mathcal{L}_\emptyset$), which has weight 0.
(2) The only caret pair which contains right carets, if it exists, may have type pair ($\mathcal{R}_\emptyset$,$\mathcal{R}_\emptyset$), which has weight 0.
(3) All other caret pairs in the tree-pair diagram must be one of the following types (note that the order in which the types appear in the caret pair does not change its weight, i.e., $w(\tau_1, \tau_2) = w(\tau_2, \tau_1)$ where τ_1 and τ_2 are caret types):
    ($\mathcal{L}_\mathrm{L}$,$\mathcal{L}_\mathrm{L}$) which has weight 2,
    ($\mathcal{M}_\emptyset^i$, $\mathcal{L}_\mathrm{L}$) which has weight 2,
    ($\mathcal{M}_\emptyset^i$, $\mathcal{M}_\emptyset^j$) which has weight 2,
    ($\mathcal{M}_\emptyset^i$, $\mathcal{M}_l^k$) which has weight 2 if $i \leq l$ and weight 4 if $i > l$,
    ($\mathcal{M}_k^i$, $\mathcal{L}_\mathrm{L}$) which has weight 2,
    ($\mathcal{M}_k^i$, $\mathcal{M}_\emptyset^j$) which has weight 2 if $k > j$ and weight 4 if $k \leq j$,
    ($\mathcal{M}_k^i$, $\mathcal{M}_l^k$) which has weight 2.

So the weight of each caret pair in the tree-pair diagram is greater than or equal to 2, with the exception of the first and last caret pair. Adding all of these together gives us the following length of the element $h_r^{-1}h'_l$:
$|h_r^{-1}h'_l| \geq 2(N(h_r^{-1}h'_l) - 2)$. □

Now we use $|h_r^{-1}h'_l|$ to compute $|h_r^{-1}h_l|$.

Corollary 4.14. $|h_r^{-1}h'_l| \geq m - 2$

Proof. By looking at the minimal tree-pair diagram representative of $h_r^{-1}h'_l$ in Figure 23, we can see that $N(h_r^{-1}h'_l) \geq q + 2$, because simply counting the labeled carets $\wedge_{\beta_0}, \ldots, \wedge_{\beta_{q+1}}$ in the positive tree yields $q + 2$ carets. By Lemma 4.13, $|h_r^{-1}h'_l| \geq 2(N(h_r^{-1}h'_l) - 2) \geq 2q = m - 2$. □

Lemma 4.15. $|h_r^{-1}h_l| = |h_r^{-1}h'_l| + 3$.

Proof. We can see by considering the tree-pair diagram representative for $h_r^{-1}h_l = h_r^{-1}h'_l x_0^{-1} x_p x_0$ in Figure 30 that $h_r^{-1}h_l$ will have exactly the same caret pairings as $h_r^{-1}h'_l$ did except for the caret pairs $\wedge_{\beta_{q+1}}, \wedge_{\beta+q+2}, \wedge_{\beta_{q+3}}$. If present in $h_r^{-1}h'_l$, the caret pair $\wedge_{\beta_{q+1}}$ produces the type pair ($\mathcal{R}_\emptyset$,$\mathcal{R}_\emptyset$) which has weight 0, but in $h_r^{-1}h_l$ the type of this caret pair is ($\mathcal{R}_{n-1}$,$\mathcal{R}_\emptyset$) which has weight 2. The caret pair $\wedge_{\beta_{q+2}}$ was added to the minimal tree-pair diagram representative of $h_r^{-1}h'_l$ in order for multiplication by $x_0^{-1}x_p x_0$ to take place, so it did not contribute any weight to the length of $h_r^{-1}h'_l$. In $h_r^{-1}h_l$ it has type pair ($\mathcal{M}_\emptyset^{n-1}$,$\mathcal{R}_\emptyset$), which has weight 1.

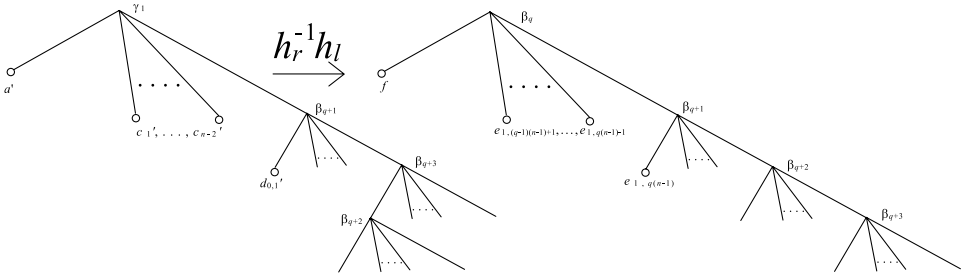

FIGURE 30. Minimal tree-pair diagram representative of $h_r^{-1}h_l$ ($h_r^{-1}h_l'x_0^{-1}x_px_0$). (Here the subtree f is exactly the subtree present in Q_+ in Figure 23 which has $\wedge_{\beta_{q-1}}$ as the root.) in $F(n)$

The caret pair $\wedge_{\beta_{q+3}}$ was added to the minimal tree-pair diagram representative of $h_r^{-1}h_l'$ in order for multiplication by $x_0^{-1}x_px_0$ to take place, so it did not contribute any weight to the length of $h_r^{-1}h_l'$ but in $h_r^{-1}h_l$ it has type $(\mathcal{R}_\emptyset,\mathcal{R}_\emptyset)$, which has weight 0. So

$$|h_r^{-1}h_l| = |h_r^{-1}h_l'| + 3 \qquad \square$$

Corollary 4.16. $|h_r^{-1}h_l| \geq m+1$.

Proof. By Lemma, 4.15, $|h_r^{-1}h_l| \geq |h_r^{-1}h_l'|+3$ and by Lemma 4.14, $|h_r^{-1}h_l'| \geq m-2$, so $|h_r^{-1}h_l| \geq m+1$. $\square$

References

[1] BELK, JAMES M.; BROWN, KENNETH S. Forest diagrams for elements of Thompson's group F. *Internat. J. Algebra Comput.* **15** (2005) 815–850. MR2197808 (2007d:20069), Zbl pre05017961.

[2] BELK, JAMES; BUX, KAI-UWE. Thompson's group F is maximally nonconvex. *Contemp. Math.* **372** (2005) 131–146. MR2139683 (2006f:20050), Zbl 1085.20021.

[3] BRIN, MATTHEW G.; GUZMÁN, FERNANDO. Automorphisms of generalized Thompson groups. *J. Algebra* **203** (1998) 285–348 MR1620674 (99d:20056), Zbl 0930.20039.

[4] BROWN, KENNETH S. Finiteness properties of groups. *J. Pure Appl. Algebra* **44** (1987) 45–75. MR0885095 (88m:20110), Zbl 0613.20033.

[5] BROWN, KENNETH S.; GEOGHEGAN, ROSS. An infinite-dimensional torsion-free FP_∞ group. *Invent. Math.* **77** (1984), no. 2, 367–381. MR0752825 (85m:20073), Zbl 0426.20039.

[6] CANNON, JAMES W. Almost convex groups. *Geom. Dedicata* **22** (1987) 197–210. MR0877210 (88a:20049), Zbl 0607.20020.

[7] CANNON, J. W.; FLOYD, W. J.; PARRY, W. R. Introductory notes on Richard Thompson's groups. *Enseign. Math.* **42** (1996) 215–256. MR1426438 (98g:20058), Zbl 0880.20027.

[8] CLEARY, SEAN; TABACK, JENNIFER. Dead end words in lamplighter groups and other wreath products. *Q. J. Math.* **56** (2005), no. 2, 165–178. MR2143495 (2006h:20055), Zbl pre02230593.

[9] CLEARY, SEAN; TABACK, JENNIFER. Thompson's group F is not almost convex. *J. Algebra* **270** (2003) 133–149. MR2016653 (2004m:20077), Zbl 1054.20022.

[10] ELDER, MURRAY; HERMILLER, SUSAN. Minimal almost convexity. *J. Group Theory* **8** (2005), no. 2, 239–266. MR2126733 (2005j:20048), Zbl pre02157197.

[11] FORDHAM, S. BLAKE. Minimal length elements of $F(p)$. Preprint.

[12] HIGMAN, GRAHAM. Finitely presented infinite simple groups. Notes on Pure Mathematics, No. 8 (1974). *Department of Pure Mathematics, Department of Mathematics, I.A.S. Australian National University, Canberra*, 1974. MR0376874 (51 #13049).

[13] KAPOVICH, ILYA. A note on the Ponaru condition. *J. Group Theory* **5** (2002), no. 1, 119–127. MR1879521 (2003d:20039), Zbl 0996.20018.

[14] MCKENZIE, RALPH; THOMPSON, RICHARD J. An elementary construction of unsolvable word problems in group theory. *Word problems: decision problems and the Burnside problem in group theory* (Conf., Univ. California, Irvine, Calif. 1969; dedicated to Hanna Neumann), Studies in Logic and the Foundations of Math., **71**, 457–478. *North-Holland, Amsterdam*, 1973. MR0396769 (53 #629), Zbl 0286.02047.

DEPARTMENT OF MATHEMATICS, BOROUGH OF MANHATTAN COMMUNITY COLLEGE/CITY UNIVERSITY OF NEW YORK, 199 CHAMBERS ST., NEW YORK, NY 10007
cwladis@gmail.com

This paper is available via http://nyjm.albany.edu/j/2007/13-19.html.