MINIMAL LENGTH ELEMENTS OF THOMPSON'S GROUPS $F(p)$

S. BLAKE FORDHAM AND SEAN CLEARY

ABSTRACT. We describe a method for determining the minimal length of elements in the generalized Thompson's groups $F(p)$. We compute the length of an element by constructing a tree pair diagram for the element, classifying the nodes of the tree and summing associated weights from the pairs of node classifications. We use this method to effectively find minimal length representatives of an element.

INTRODUCTION

Thompson's group F is a perplexing example of a finitely-presented group which is the simplest known example of a wide variety of a number of unusual group-theoretic phenomena. Cannon, Floyd and Parry [2] give an excellent introduction to a wide range of the properties of F. Fordham [3] developed an effective method for measuring lengths of elements in F with respect to the standard finite generating set and for finding minimal length representatives. Thompson's group F can be seen as a group of piecewise-linear homeomorphisms with dyadic breakpoints or as a group of rooted binary tree pairs. There are generalizations of F to p-adic breakpoint sets with slopes integral powers of p, and equivalently, to groups of rooted p-ary tree pairs. These generalizations were introduced by Higman [5], and studied by Brown [1] and Stein [6]. Here we extend Fordham's method for computing the minimal lengths of elements and finding minimal length representatives to the groups $F(p)$.

In the following, $\bar{g}$ denotes the inverse of a group element g. The group $F(p)$ has infinite presentation:

$$\langle c_0, c_1, c_2, \dots | \bar{c}_i c_n c_i = c_{n+p-1}, \forall i < n \rangle .$$

There is a set of normal forms for elements of F given by:

$$c_{i_1}^{r_1} c_{i_2}^{r_2} \dots c_{i_k}^{r_k} c_{j_l}^{-s_l} \dots c_{j_2}^{-s_2} c_{j_1}^{-s_1}$$

Key words and phrases. Thompson's group.

The second author gratefully acknowledges support from the PSC-CUNY Research Awards program and the hospitality of the Centre de Recerca Matemàtica .

with $r_i, s_i > 0$, $i_1 < i_2 \ldots < i_k$ and $j_1 < j_2 \ldots < j_l$. This normal form is unique if we further require a reduction condition that when both c_i and $\bar{c}_i$ occur, so does at least one of $c_{i+1}, c_{i+2} \ldots c_{i+p-1}$ or their inverses. The relations give effective means for putting a word in the infinite generating set into normal form. The generator c_0 is also called a. It is clear from the relations that the set $\{a, c_1, \ldots, c_{p-1}\}$ are sufficient to generate the group. In all of the following, we refer to length of words in $F(p)$ with respect to this finite generating set.

1. Nodes and Trees

The elements of $F(p)$ can be represented graphically as equivalence classes of pairs of rooted p-ary trees, both having the same number of nodes. These equivalence classes tree pair diagrams form a group under a natural operation of composition. The binary case, when $p = 2$, is Thompson's group F and many of the properties of F occur in $F(p)$. Stein [6] gives an excellent description of $F(p)$.

We now extend the notion of $\wedge$-nodes and $\wedge$-trees as described for F in [3, 4] to $F(p)$.

1.1. **Nodes and Ordering.** We consider ordered, rooted p-ary trees where each interior node has exactly p children, which are each interior nodes or exterior nodes. These p children are ordered and are divided into two classes: left children and right children, and there is at least one left and right child for each interior node. We call exterior nodes *leaves*. An interior node together with its downward directed edges is called a *caret node* or $\wedge$-node. A tree pair (S, T) is two p-ary trees with the same number of nodes. We sometimes refer to the first tree in the pair as the *domain* tree and the second as the *range* tree, reflecting their roles in describing subdivisions for interpolation to get piece-wise linear homeomorphisms of the unit interval. Caret nodes are ordered recursively by a variation of the standard infix order traversal of the tree, where we order the left children of a $\wedge$-node before that $\wedge$-node, and the right children after.

For the remainder of this paper, unless specifically noted otherwise, we will restrict discussion to the group $F(p+1)$where $p \geq 2$. This small change will simplify much of the notation and makes the bookkeeping in the proofs easier to follow.

1.2. **The groups** $F(p+1)$**.** Just as Thompson's group F can be described as the group of rooted binary tree pair diagrams under a natural operation of composition, the groups $F(p+1)$ can be described with rooted $p+1$-ary tree pair diagrams.

Definition 1.1 (Generators of $F(p+1)$)**.** Figure 1 illustrates the $\wedge$-tree pairs for the standard $p+1$ generators of $F(p+1)$, $a = c_0, c_1, \ldots, c_p$.

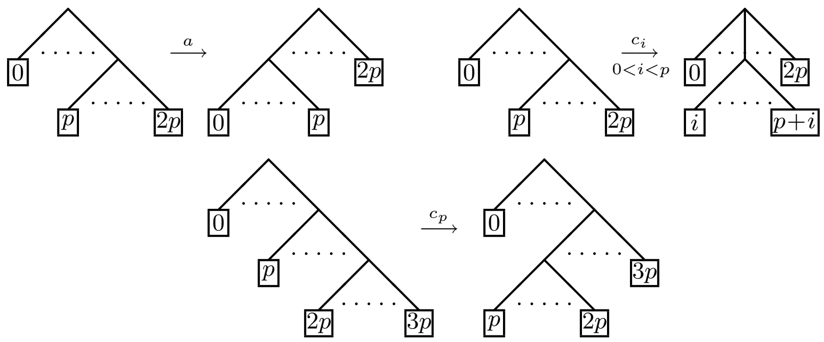


FIGURE 1. The Generators of $F(p+1)$.

Definition 1.2 (Primary $\wedge$-node types)**.** There are $p+2$ main types of $\wedge$-nodes in $F(p+1)$: left nodes ($\mathcal{L}$), right nodes ($\mathcal{R}$), and middle nodes ($\mathcal{M}^i$ for $1 \le i \le p$). We assign these types by the following recursive procedure. The root node is always type $\mathcal{L}$ and it has one left child node of type $\mathcal{L}$, then p right child nodes of types $\mathcal{M}^1, \mathcal{M}^2, \ldots \mathcal{M}^{p-1}$ and $\mathcal{R}$ ordered from left to right, as shown in Figure 2. A left node has one left child node of type $\mathcal{L}$, then p right child nodes of types $\mathcal{M}^1, \mathcal{M}^2, \ldots \mathcal{M}^p$. A right node has one left child node of type $\mathcal{M}^p$, then p right child nodes of types $\mathcal{M}^1, \mathcal{M}^2, \ldots \mathcal{M}^{p-1}$ followed by a right child node of type $\mathcal{R}$.

Nodes of type $\mathcal{M}^1$ have p left children of types $\mathcal{M}^1$ through $\mathcal{M}^p$ and a single right child of type $\mathcal{M}^1$. For nodes of type $\mathcal{M}^i$, there are $p+1-i$ left children of types $\mathcal{M}^i$ through $\mathcal{M}^p$ and i right children of types $\mathcal{M}^1$ through $\mathcal{M}^i$. Finally, nodes of type $\mathcal{M}^p$ each have a single left child of type $\mathcal{M}^p$ and right children of types $\mathcal{M}^1$ through $\mathcal{M}^p$. These $\wedge$-node types are illustrated in Figure 2.

The gaps drawn between the child nodes in Figure 2 are significant; the gaps indicate the division between left and right child nodes. All child or leaf nodes drawn to the horizontal left of the $\wedge$-node and before the gap are left children of the nodes and are said to be *to the left* of the parent node. Similarly, the remaining children are *right children*.

We note that in $F(2)$, the middle node type $\mathcal{M}^1$ is the same as the interior node type $\mathcal{I}$ defined in [3].

Definition 1.3 (Node order)**.** The *order* of the $\wedge$-nodes of a $\wedge$-tree are determined recursively, using Figure 2, as follows:

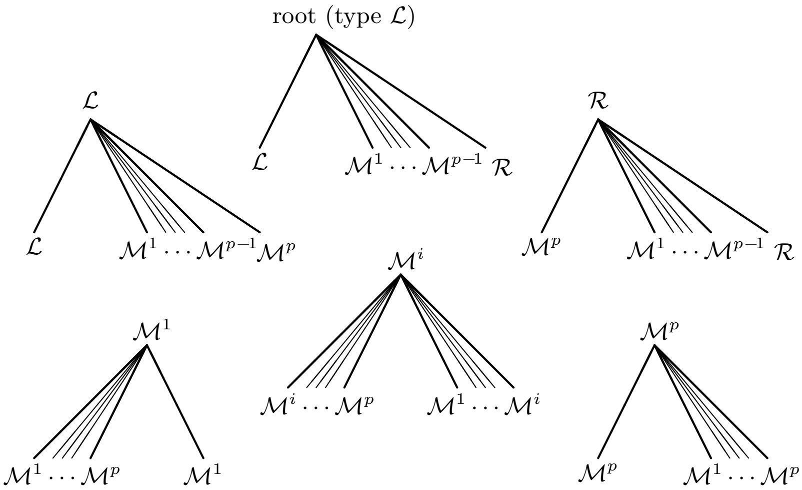


FIGURE 2. Types of the $\wedge$-nodes and the corresponding child nodes.

(i) If a $\wedge$-tree contains n $\wedge$-nodes, then the nodes are numbered $0, \dots, n-1$.
(ii) All left children of a $\wedge$-node are numbered less than that $\wedge$-node, and all right children are numbered greater than that $\wedge$-node.
(iii) The children of a node are ordered in accordance to their type, namely, if a left child of type $\mathcal{M}^j$ proceeds all left siblings of type $\mathcal{M}^k$ when $j < k$ and follows all left siblings of type $\mathcal{M}^i$ when $i < j$. Similarly, the right children of a node are ordered by their types.

Graphically, this node ordering means that nodes are numbered from 0 to $n-1$ from the left to the right as encountered by their vertical placement in accordance with the gaps separating left children from right children as shown in Figure 2.

Since this ordering of the $\wedge$-nodes of a tree is a total ordering, we can clearly identify the predecessor and successor nodes of any $\wedge_i$. Similarly, the *immediate predecessor* and *immediate successor* of $\wedge_i$ are, respectively, $\wedge_{i-1}$ and $\wedge_{i+1}$.

Although we will be primarily concerned with the ordering defined for the $\wedge$-nodes of a tree, we will occasionally need to identify the leaves of the trees. The leaves of a $\wedge$-tree inherit an ordering from the $\wedge$-nodes:

Definition 1.4 (Leaf order)**.** For a tree with n $\wedge$-nodes, there are $np + 1$ leaves and the leaves are numbered $0, 1, \dots, np$ by requiring that leaves of a node be numbered larger than all the leaves of any predecessor nodes and that the leaves of node are numbered in the obvious left-to-right order induced by the node order.

Definition 1.5 (Exposed caret)**.** A $\wedge$-node where all children are leaves is called an *exposed caret.*

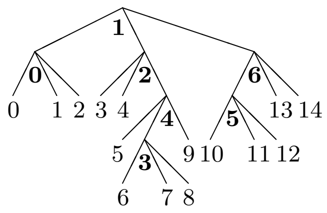


FIGURE 3. An example $\wedge$-tree from $F(3)$ with nodes and leaves labelled.

i	leftmost child of $\wedge_i$	leaf-index of $\wedge_i$	$\tau(\wedge_i)$
0	leaf 0	0	$\mathcal{L}$
1	$\wedge_0$	0	$\mathcal{L}$
2	leaf 3	3	$\mathcal{M}^1$
3	leaf 6	6	$\mathcal{M}^2$
4	leaf 5	5	$\mathcal{M}^1$
5	leaf 10	10	$\mathcal{M}^2$
6	$\wedge_5$	10	$\mathcal{R}$

TABLE 1. The node and leaf indices of the example $\wedge$-tree from Figure 3.

Definition 1.6 (Reduced and unreduced tree pairs)**.** A tree pair (S, T) is *unreduced* if there is an i such that the i-th through $(i+p)$th leaves of S are all children of a single $\wedge$-node and the corresponding i-th through $(i+p)$th leaves of T are children of a single $\wedge$-node. Equivalently, such a tree pair is unreduced if all of the leaves of an exposed $\wedge$-node in S are numbered the same as all of the leaves of an exposed $\wedge$-node in T. A *reduced* tree pair diagram is not unreduced.

Just as in F, there is a unique reduced tree pair for each element of $F(p+1)$. Similarly, the notion of leaf exponent described in [2] gives an easy mechanism for changing between the tree pair representation and algebraic normal forms, and the reduction condition on tree pairs corresponds exactly to the algebraic reduction condition described in the introduction.

Definition 1.7 (Leaf-index)**.** If $\wedge_n$ is a $\wedge$-node of a tree, then we define the *leaf-index* of $\wedge_n$ to be i if

(i) the leftmost child of $\wedge_n$ is the leaf labeled i, or
(ii) the leftmost child of $\wedge_n$ is a $\wedge$-node with leaf-index i.

For example, the $\wedge$-tree shown in Figure 3 has its $\wedge$-nodes and leaves both numbered and the leaf-index of each $\wedge$-node is listed in Table 1.

Theorem 1.8. *If $\wedge_n$ is a $\wedge$-node of type $\mathcal{M}^j$ in a $p+1$-ary tree T with leaf-index i then $i \simeq j \mod p$. For $\wedge$-nodes of type $\mathcal{L}$ or $\mathcal{R}$, $i \simeq 0 \mod p$.*

Proof. If we remove any exposed $\wedge$-node other than $\wedge_n$, the number of leaves in T is decreased by p. If the removed node was a predecessor of $\wedge_n$ then the original node of interest is now $\wedge_{n-1}$ in the new tree T' and it has leaf-index i or $i-p$. If the removed node is a successor of $\wedge_n$ in T then the index of the caret and its leaf-index are both unchanged in T'. Continuing this process, if we remove all exposed carets of T other than $\wedge_n$ then in the new tree T' the original $\wedge_n$ is now $\wedge_{n-m}$ with leaf-index of $i-mp$ for some $m \geq 0$, $\wedge_{n-m}$ is the only exposed caret in T' and all the $\wedge$-nodes of T' lie in the path from the root of the tree to the root of $\wedge_{n-m}$. Each caret on this path has exactly one child except for $\wedge_{n-m}$ which has no $\wedge$-node children and the leaf-index of the original $\wedge_n$ is congruent to the leaf-index of $\wedge_{n-m}$ modulo p.

We can now assume that T is a tree of $\wedge$-nodes where each node has exactly one child except for the exposed $\wedge$-node $\wedge_n$ which has a leaf-index i. If T consists of a single $\wedge$-node then the root node is type $\mathcal{L}$ with $n=0$ and $i=0$. If T has more than one $\wedge$-node, then we examine the relationship between the leaf-index of the exposed $\wedge$-node and the leaf-index of its parent.

If the exposed node is type $\mathcal{L}$, then all of its ancestors are also left nodes and the leaf-index is 0. If the exposed node is a right node then all its ancestors except the root are also type $\mathcal{R}$ and $\wedge_n$ is the last node of the tree so the leaf-index must be $np-p$ which is congruent to 0 modulo p.

If $\wedge_n$ is type $\mathcal{M}^j$ and its parent is type $\mathcal{L}$ then the parent node must have leaf-index 0 and so $j=i$. If the parent node of $\wedge_n$ is type $\mathcal{R}$ then the last leaf of the tree is the last leaf of the parent node of $\wedge_n$. In this case, $\wedge_n$ is either type $\mathcal{M}^p$ and the predecessor of its parent or $\wedge_n$ is type $\mathcal{M}^j$ $(j<p)$ and the rightmost $\wedge$-node of the tree. In the first case, $i=(n+1)p-2p \simeq p$ and, in the second case, $i=np-2p+j \simeq j$.

Lastly, using the type diagrams in Figure 2, if the parent of $\wedge_n$ is type $\mathcal{M}^k$, assuming the parents leaf-index is $k \mod p$, then $i \simeq k$ if $\wedge_n$ is the leftmost child of its parent, $i \simeq k+(p-k)+j \simeq j \mod p$ if $k \leq j$, and $i \simeq k+(j-k) \simeq j$ if $j>k$. Therefore, by induction $i \simeq j \mod p$. □

Note that in the example shown in Figure 3, that $\wedge_2$ and $\wedge_4$ have leaf-indices congruent to 1 modulo 2 and the rest are congruent to 0 modulo 2.

1.3. **Types and Weight.** Once the $\wedge$-nodes are ordered the specific types of the nodes can be determined using the following definition:

Definition 1.9 (Node Types)**.** The *type* of a $\wedge$-node, $\tau(\wedge_i)$, is one of the following:

- $\mathcal{L}_\emptyset$ – Left $\wedge$-node with no predecessor; $\wedge_0$ is always the only $\wedge$-node of this type.
- $\mathcal{L}_\mathrm{L}$ – Left $\wedge$-node with a predecessor; that is, all left $\wedge$-nodes other than $\wedge_0$.
- $\mathcal{R}_\emptyset$ – Right $\wedge$-node where all successors are right $\wedge$-nodes.
- $\mathcal{R}_\mathrm{R}$ – Right $\wedge$-node with an immediate successor that is a right $\wedge$-node, but not all successors are right nodes.
- $\mathcal{R}_j$ – Right $\wedge$-node with an immediate successor that is not a right $\wedge$-node and with a leftmost child successor of type $\mathcal{M}^j$ where $j < p$. If the leftmost child successor is type $\mathcal{R}$, we use $j = p$.
- $\mathcal{M}^i_\emptyset$ – Middle $\wedge$-node of type $\mathcal{M}^i$ with no child successor $\wedge$-nodes.
- $\mathcal{M}^i_j$ – Middle $\wedge$-node of type $\mathcal{M}^i$ that has a leftmost child successor of type $\mathcal{M}^j$ (the definition of the $\mathcal{M}^i$ type requires that $j \leq i$).

The type of the $\wedge_i$ in a reduced tree pair (S, T) is the ordered pair of types for $\wedge_i$ in the individual trees, i.e. $\tau_{(S,T)}(\wedge_i) = (\tau_S(\wedge_i), \tau_T(\wedge_i))$.

If the exact type of a $\wedge$-node is unknown or varies according to the circumstance, we will commonly use $\mathcal{L}_*$, $\mathcal{R}_*$ and $\mathcal{M}^i_*$ to represent the general node types. In most cases, $\wedge$-node of type $\mathcal{R}_\emptyset$ and $\mathcal{R}_\mathrm{R}$ have the same weight and behavior; when a node may be either of these two types but not type $\mathcal{R}_j$, we will use $\mathcal{R}_\mathrm{N}$ to represent the type of the node.

We note that in F, the caret type $\mathcal{R}_\mathrm{R}$ corresponds to R_{NI}, the caret type $\mathcal{M}^1_\emptyset$ corresponds to I_0, and $\mathcal{M}^1_1$ corresponds to I_R as described by Fordham [3, 4].

The definition of type $\mathcal{R}_\emptyset$ makes it impossible for a reduced pair of trees to have more than one $\wedge$-node pair of type $(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$. Also, any node pair of this type must be the last node pair of the trees.

Corollary 1.10. *(corollary of Theorem 1.8) If $\wedge_n$ is a reducible, exposed $\wedge$-node in the pair (S, T) then*

$$\tau_S(\wedge_n) = \tau_T(\wedge_n).$$

Proof. If $\wedge_n$ is reducible in (S, T), the leaves of $\wedge_n$ must be numbered $i, i+1, \ldots, i+p$ in both trees. By Theorem 1.8, if $i \not\equiv 0 \mod p$ and $\wedge_n$ is type $\mathcal{M}^j$ in S then it is also type $\mathcal{M}^j$ in T. If $i \simeq 0 \mod p$ then $\wedge_n$ must be type $\mathcal{L}$, $\mathcal{R}$, or $\mathcal{M}^p$. $\wedge_n$ is type $\mathcal{L}$ only if $i = 0$, in which case, $n = 0$ and the node is type $\mathcal{L}$ in both trees. If $\wedge_n$ is an exposed $\wedge$-node of type $\mathcal{R}$ in S then $\wedge_n$ is the last node of (and thus type $\mathcal{R}$ in) both trees and must have leaf-index $i = (n-1)p \simeq 0 \mod p$. Any other node where $i \simeq 0$ must be of type $\mathcal{M}^p$ in both trees. □

From the previous definition, the types of the child nodes of a $\wedge$-node of type $\mathcal{M}^i$ are quite restricted. The following two lemmas, which will be used frequently throughout this paper, describe two important type restrictions.

Lemma 1.11. *If $\wedge_n$ is a $\wedge$-node of type $\mathcal{M}^i$, the rightmost descendant (if such a node exists) of $\wedge_n$ must be of type $\mathcal{M}^j_\emptyset$ where $j \leq i$.*

Proof. By the definition of type $\mathcal{M}^i_k$, k must be less than or equal to i. Therefore, if $\wedge_n$ has a rightmost child then it must be type $\mathcal{M}^{i_1}$ with $i_1 \leq i$. Similarly, the rightmost child of this $\wedge$-node must be of type $\mathcal{M}^{i_2}$ with $i_2 \leq i_1 \leq i$. Continuing inductively, the rightmost descendant of $\wedge_n$ must be a $\wedge$-node with no successor and of type $\mathcal{M}^j_\emptyset$ where $j \leq \ldots \leq i_2 \leq i_1 \leq i$. □

Lemma 1.12. *If $\wedge_n$ is a $\wedge$-node of type $\mathcal{M}^i$, the leftmost descendant (if such a node exists) of $\wedge_n$ must be of type $\mathcal{M}^j_*$ where $j \geq i$.*

Proof. By the definition of type $\mathcal{M}^i$, if $\wedge_n$ has a predecessor child then the leftmost predecessor must be type $\mathcal{M}^{i_1}$ with $i_1 \geq i$. Similarly, the leftmost child of this $\wedge$-node must be of type $\mathcal{M}^{i_2}$ with $i_2 \geq i_1 \geq i$. Continuing inductively, the leftmost descendant of $\wedge_n$ must be a $\wedge$-node of type $\mathcal{M}^j$ where $j \geq \ldots \geq i_2 \geq i_1 \geq i$. This node may have successor children, so the type of $\wedge_n$ is $\mathcal{M}^j_*$. □

Definition 1.13 (Label sets)**.** A label set is a subset of the set of generators of $F(p+1)$ and their inverses. For every $\wedge$-node pair in a reduced pair of trees (S,T), we can assign a label set $\lambda_{(S,T)}(\wedge_i)$.

Table 2 gives the label sets for the various possible types of $\wedge$-node pairs, except for $(\mathcal{L}_\emptyset, \mathcal{L}_\emptyset)$ which has label set $\emptyset$.

Definition 1.14 (Weight)**.** We define the *weight* of the i^{th} $\wedge$-node pair of the reduced pair (S,T) to be the cardinality of the node pair's label set; that is,

$$\mu_{(S,T)}(\wedge_i) = \|\,\lambda_{(S,T)}(\wedge_i)\|.$$

The weight of the $\wedge$-tree pair, $\mu\,(S,T)$, is the sum of the weights of all the $\wedge$-nodes in the tree pair.

Domain Type ($j_1 \leq i < j_2$, $i_1 < j \leq i_2$)

		$\mathcal{L}_{\mathrm{L}}$	$\mathcal{R}_{\emptyset}$	$\mathcal{R}_{\mathrm{R}}$	$\mathcal{R}_j$	$\mathcal{M}^i_{\emptyset}$	$\mathcal{M}^i_j$
	$\mathcal{L}_{\mathrm{L}}$	$\{a, \bar{a}\}$	$\{a\}$	$\{a\}$	$\{a\}$	$\{\bar{c}_i, a\}$	$\{\bar{c}_i, a\}$
	$\mathcal{R}_{\emptyset}$	$\{\bar{a}\}$	$\emptyset$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{\bar{c}_i\}$	$\{\bar{c}_i, a, \bar{a}\}$
	$\mathcal{R}_{\mathrm{R}}$	$\{\bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{\bar{c}_i\}$	$\{\bar{c}_i, a, \bar{a}\}$
Range	$\mathcal{R}_{j_1}$	$\{\bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{\bar{c}_i, a, \bar{a}\}$	$\{\bar{c}_i, a, \bar{a}\}$
type	$\mathcal{M}^{i_1}_{\emptyset}$	$\{c_{i_1}, \bar{a}\}$	$\{c_{i_1}\}$	$\{c_{i_1}\}$	$\{c_{i_1}\}$	$\{\bar{c}_i, c_{i_1}\}$	$\{\bar{c}_i, c_{i_1}\}$
	$\mathcal{M}^{i_1}_{j_1}$	$\{c_{i_1}, \bar{a}\}$	$\{c_{i_1}, a, \bar{a}\}$	$\{c_{i_1}, a, \bar{a}\}$	$\{c_{i_1}, a, \bar{a}\}$	$\{\bar{c}_i, c_{i_1}, a, \bar{a}\}$	$\{\bar{c}_i, c_{i_1}, a, \bar{a}\}$
	$\mathcal{R}_{j_2}$	$\{\bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{a, \bar{a}\}$	$\{\bar{c}_i\}$	$\{\bar{c}_i, a, \bar{a}\}$
	$\mathcal{M}^{i_2}_{\emptyset}$	$\{c_{i_2}, \bar{a}\}$	$\{c_{i_2}\}$	$\{c_{i_2}\}$	$\{c_{i_2}, a, \bar{a}\}$	$\{\bar{c}_i, c_{i_2}\}$	$\{\bar{c}_i, c_{i_2}, a, \bar{a}\}$
	$\mathcal{M}^{i_2}_{j_2}$	$\{c_{i_2}, \bar{a}\}$	$\{c_{i_2}, a, \bar{a}\}$	$\{c_{i_2}, a, \bar{a}\}$	$\{c_{i_2}, a, \bar{a}\}$	$\{\bar{c}_i, c_{i_2}\}$	$\{\bar{c}_i, c_{i_2}, a, \bar{a}\}$

TABLE 2. Label sets for all $\wedge$-node types.

Even though we have required that a pair of trees must be reduced before calculating the weight, it will be occasionally useful to determine the weight of an unreduced tree pair. In such cases, we define the label set of a reducible caret to be $\emptyset$ and thus its weight is 0. Furthermore, we note that in such cases, it is important to consider only non-reducible carets in determining $\wedge$-node type for the non-reducible carets.

Example 1.15. Figure 4 and Table 3 show the tree pair diagram and listing of all the types and label sets for an example element of $F(4)$.

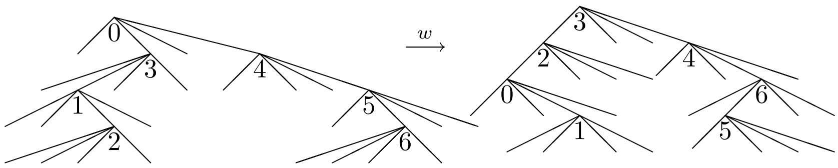


FIGURE 4. A tree pair diagram for element w in $F(4)$

i	$\tau(\wedge_i)$	$\lambda(\wedge_i)$	μ
0	$(\mathcal{L}_\emptyset, \mathcal{L}_\emptyset)$	$\emptyset$	0
1	$(\mathcal{M}_1^2, \mathcal{M}_\emptyset^2)$	$\{c_2, \bar{c}_2, a, \bar{a}\}$	4
2	$(\mathcal{M}_\emptyset^1, \mathcal{L}_\mathrm{L})$	$\{c_1, a\}$	2
3	$(\mathcal{M}_\emptyset^1, \mathcal{L}_\mathrm{L})$	$\{c_1, a\}$	2
4	$(\mathcal{R}_\mathrm{R}, \mathcal{R}_2)$	$\{a, \bar{a}\}$	2
5	$(\mathcal{R}_1, \mathcal{M}_\emptyset^3)$	$\{c_3, a, \bar{a}\}$	3
6	$(\mathcal{M}_\emptyset^1, \mathcal{M}_\emptyset^2)$	$\{\bar{c}_1, c_2\}$	2

TABLE 3. Caret pairings, label sets and weights for w from Figure 4

We now compute the weights of the identity and generators.

Theorem 1.16 (Weight of the identity)**.** *For an element $w \in F(p+1)$, $\mu(w) = 0$ iff $w = \mathrm{id}_{F(p+1)}$.*

Proof. The identity is represented by a pair of trees where each tree consists of a single $\wedge$-node, so by the definition of weight, $\mu(\mathrm{id}_{F(p+1)}) = \mu(\mathcal{L}_\emptyset, \mathcal{L}_\emptyset) = 0$. Conversely, if $w \in F(p+1)$ is an element of weight zero then, in the reduced pair of trees representing w, the $\wedge$-nodes must all have zero weight since weight is non-negative. The only types that have zero weight are $(\mathcal{L}_\emptyset, \mathcal{L}_\emptyset)$ and $(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$, so each tree must consist of a root node and n right $\wedge$-nodes for some $n \geq 0$. If the pair is reduced, then n must be zero and w is the identity. □

Using the $\wedge$-tree pairs of Figure 1 and adding up the appropriate weights, we can easily prove the following theorem:

Theorem 1.17. *For any generator $g \in F(p+1)$, $\mu(g) = 1$.* □

2. The Main Proof

We are interested in determining the minimal length of an element in a group based on a geometric representation of the element. Assuming we have a function φ from the words in the generators of a group to the nonnegative integers, we need to be able to determine whether or not φ is the same as the minimal length function ℓ with respect to that generating set. The following lemma from [3] gives a general criterion to characterize such functions.

Lemma 2.1 (Classifying minimal length)**.** *Given a generating set X of a group G and a function $\varphi : G \to \{0, 1, 2, \ldots\}$, if φ has the properties*

(1) $\varphi(\mathrm{id}_G) = 0$,
(2) *if* $\varphi(g) = 0$ *then* $g = \mathrm{id}_G$,
(3) *if* $g \in G$ *and* $x \in X$ *then* $\varphi(g) - 1 \leq \varphi(g\bar{x})$,
(4) *for any non-identity element* $g \in G$*, there is at least one generator* x *of* G *such that* $\varphi(g) - 1 = \varphi(g\bar{x})$

then $\varphi(g) = \ell(g)$ *for all* $g \in G$.

Proof. Assume that $x_n x_{n-1} \cdots x_2 x_1$ is a minimal length representative of g with respect to the generating set X of G. By definition of length, $\ell(gx_1^{-1} \cdots x_i^{-1}) = n - i$ for $1 \leq i \leq n$, and $\varphi(gx_1^{-1} \cdots x_i^{-1}) \geq \varphi(g) - i$ by property (iii). When we choose $i = n$, $\ell(gx_1^{-1} \cdots x_n^{-1}) = \ell(\mathrm{id}_G) = 0$ and $\varphi(gx_1^{-1} \cdots x_n^{-1}) = \varphi(\mathrm{id}_G) \geq \varphi(g) - n$. Therefore, by property (i) we have

$$\begin{aligned} \varphi(g) &\leq \varphi(\mathrm{id}_G) + n \\ &\leq n \\ &\leq \ell(g). \end{aligned}$$

Now assume that $\varphi(g) = n > 0$. By property (iv), there exist generators $x_1, \ldots, x_n$ such that $\varphi(gx_1^{-1} \cdots x_n^{-1}) = 0$. By property (ii), $gx_1^{-1} \cdots x_n^{-1} = \mathrm{id}_G$ so $x_n \cdots x_2 x_1$ is a representative of g with respect to the generating set X of G. Since this representation may not be minimal, we have

$$\ell(g) \leq n = \varphi(g).$$

Therefore, $\ell(g) = \varphi(g)$. □

Theorem 2.2 (Minimal length in $F(p+1)$)**.** *If $w \in F(p+1)$ is represented by the reduced pair (S, T), the length of the minimal representative of w is*

$$\ell(w) = \mu\,(S, T).$$

We have already established the first two conditions to apply Lemma 2.1, so to prove Theorem 2.2, we now prove the third and fourth conditions in the next two subsections.

2.1. **Condition 3: generators do not change weight by more than one.** Here, we consider the possible changes in weight from application of a generator. As in F, to apply a particular generator to a reduced tree pair, we may need to add carets to obtain an unreduced representative to which that generator can be applied. First we consider the case in which no such expansion is needed and then we consider the case when we need to add one or more carets to apply a generator.

Theorem 2.3. *If x is a generator that can be applied to S without the addition of any $\wedge$-nodes, and both (S,T) and (xS,T) form a reduced tree pair with n carets, then there is exactly one $\wedge$-node, $\wedge_i$ with $i < n$, such that*

$$\tau_{(S,T)}(\wedge_i) \neq \tau_{(xS,T)}(\wedge_i).$$

Proof. If (xS,T) is not reducible, T is fixed and none of its $\wedge$-nodes change. For S and xS, careful examination of the diagrams in Figure 1 reveals that only one $\wedge$-node changes type under the action of a generator. Specifically, for a, the rightmost child of the root node in S changes from a node of type $\mathcal{R}_*$ to a node of type $\mathcal{L}_{\mathrm{L}}$ in the tree representing aS. For c_p, the rightmost grandchild of the root changes from a type $\mathcal{R}_*$ to a type $\mathcal{M}_*^p$. And for $i < p$, the generator c_i changes the rightmost child of the root into a node of type $\mathcal{M}_*^i$. The inverse generators simply reverse these changes. □

The previous theorem allows us to compare the weights of (S,T) and (xS,T) by simply comparing the weights of single $\wedge$-node in each as long as the number of carets in (S,T) remains the same as in (xS,T).

Corollary 2.4. *If (S,T) is a reduced pair of trees and x is a generator of $F(p+1)$ such that (xS,T) is also a reduced pair of trees then for some node $\wedge_n$,*

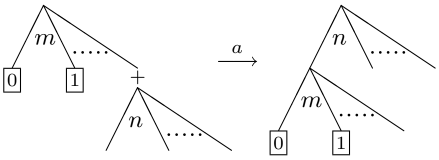
$$\Delta\omega = \mu\,(xS,T) - \mu\,(S,T) = \mu_{(xS,T)}(\wedge_n) - \mu_{(S,T)}(\wedge_n). \tag{2.1}$$

□

In cases where a $\wedge$-node of (S,T) becomes reducible after a generator is applied to S or where a $\wedge$-node needs to be added to S so that the generator can be applied, we need to take more care. The following two theorems describe the results of adding a $\wedge$-node to (S,T) or from reducing an exposed $\wedge$-node in (xS,T).

Theorem 2.5. *If x is a generator of $F(p+1)$ that cannot be applied to S without first adding a $\wedge$-node, then $\mu\,(xS,T) > \mu\,(S,T)$.*

Proof. In order to prove this theorem, we must choose S so that x cannot be applied to S, but can be applied to the tree S' formed by adding one or more $\wedge$-nodes to S. Any nodes added to (S, T) will be exposed $\wedge$-nodes in (S', T') and by definition 1.14, these nodes will have zero weight in (S', T'). We will examine each generator separately.

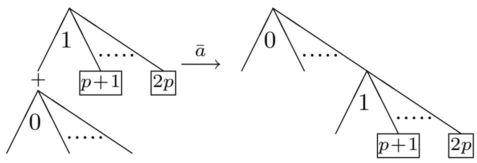


Case a:

If a cannot be applied to S, then the root of S must not have a child on its rightmost branch and we must add the $\wedge$-node, $\wedge_n$, to the last leaf of both trees in (S, T) (see the illustration above, where the + indicates the location of adding the necessary caret). If we apply a to S', all the $\wedge$-nodes of S have the same type in both S' and aS', and only $\wedge_n$ changes type so

$$\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, \mathcal{R}_{\emptyset}) - \mu_{(S',T')}(\wedge_n) = 1 - 0 = 1.$$

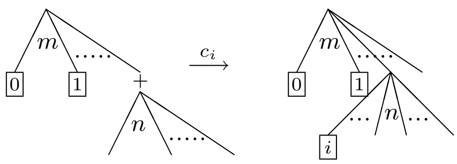
Case ā:



If $\bar{a}$ cannot be applied to S, then the root of S must not have a child on its leftmost branch and we must add the $\wedge$-node, $\wedge_0$, to the first leaf of both trees in (S, T). If we apply $\bar{a}$ to S', $\wedge_0$ changes from an exposed node in (S', T') to type $(\mathcal{L}_{\emptyset}, \mathcal{L}_{\emptyset})$ in $(\bar{a}S', T')$, and $\wedge_1$ changes from type $(\mathcal{L}_{\emptyset}, \mathcal{L}_{\emptyset})$ to type $(\mathcal{R}_*, \mathcal{L}_{\mathrm{L}})$ so

$$\Delta\omega = \mu(\mathcal{L}_{\emptyset}, \mathcal{L}_{\emptyset}) - 0 + \mu(\mathcal{R}_*, \mathcal{L}_{\mathrm{L}}) - \mu(\mathcal{L}_{\emptyset}, \mathcal{L}_{\emptyset}) = 1.$$

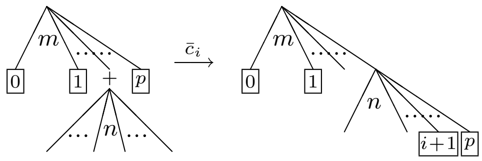
Case $\mathbf{c_i}$ (with $i < p$):



Similar to the a case above, adding $\wedge_n$ to the last leaf of the trees and applying c_i gives

$$\Delta\omega = \mu(\mathcal{M}^i_{\emptyset}, \mathcal{R}_{\emptyset}) - 0 = 1.$$

Case $\mathbf{\bar{c}_i}$ (with $i < p$): If the i^{th} leaf of the root of S is empty then adding the node, $\wedge_n$, to the trees allows us to apply $\bar{c}_i$ to the new tree S'.



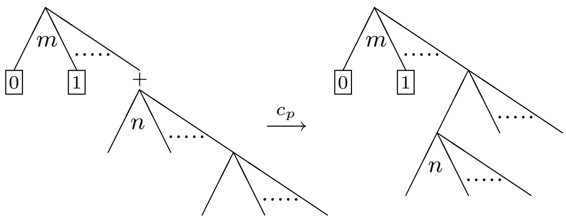
Depending on $\boxed{i+1}, \ldots, \boxed{p}$, the type of $\wedge_n$ in $\bar{c}_i S'$ may be $\mathcal{R}_\emptyset$, $\mathcal{R}_\mathrm{R}$ or $\mathcal{R}_j$ with $j \geq i+1$, so

$$\Delta\omega = \mu(\mathcal{R}_\mathrm{N}, \mathcal{M}^i_\emptyset) - 0 = 1$$

or

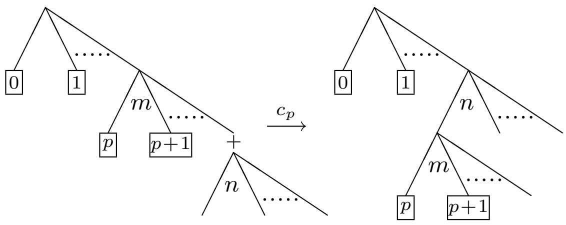
$$\Delta\omega = \mu(\mathcal{R}_j, \mathcal{M}^i_\emptyset) - 0 = 1.$$

Case $\mathbf{c_p}$: Two $\wedge$-nodes on the rightmost leaf of the root are needed to apply c_p to S. One of both of these nodes may be missing, so we need to check two cases.



If both the necessary $\wedge$-nodes are missing in S, then we need to add two nodes, $\wedge_n$ and $\wedge_{n+1}$, to the last leaf of the root of S (as shown above). Applying c_p to S' gives

$$\Delta\omega = \mu(\mathcal{M}^p_\emptyset, \mathcal{R}_\emptyset) - 0 + \mu(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) - 0 = 1.$$



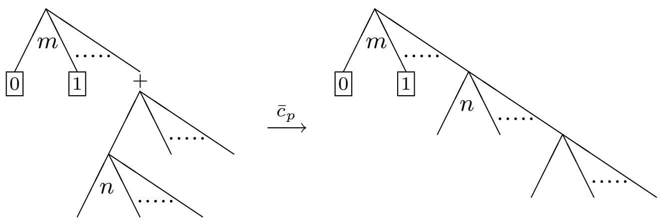
Assume that we only need to add one node, $\wedge_n$, before applying c_p to S. The type of $\wedge_m$ depends on the contents of $\boxed{p+1}, \ldots, \boxed{2p-1}$. If $\wedge_m$ is the last $\wedge$-node in (S,T) then $t_m = \tau_T(\wedge_m)$ must be either $\mathcal{L}_\mathrm{L}$ or $\mathcal{R}_\emptyset$, so

$$\Delta\omega = \mu(\mathcal{M}^p_\emptyset, t_m) - \mu(\mathcal{R}_\emptyset, t_m) = 1.$$

If $\wedge_m$ has a successor in (S,T), then for any choice of t_m,

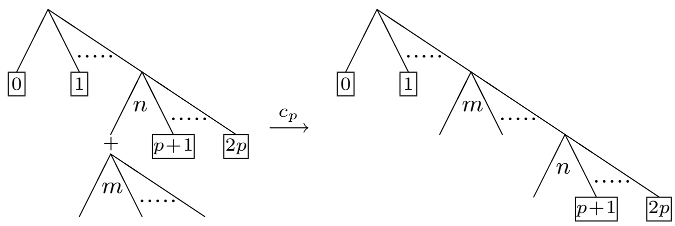
$$\Delta\omega = \mu(\mathcal{M}^p_j, t_m) - \mu(\mathcal{R}_j, t_m) = 1.$$

Case $\bar{\mathbf{c}}_\mathbf{p}$: Again, one or two $\wedge$-nodes may be needed to be added S before we can apply $\bar{c}_p$.



If we add two carets then

$$\Delta\omega = \mu(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) - 0 + \mu(\mathcal{R}_\emptyset, \mathcal{M}^p_\emptyset) - 0 = 1.$$



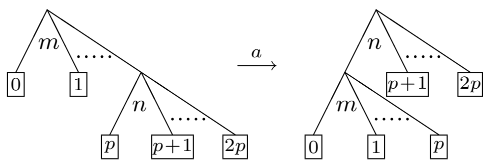
If we add one caret then

$$\Delta\omega = \mu(\mathcal{R}_\mathrm{N}, \mathcal{M}^p_\emptyset) - 0 = 1.$$

□

Theorem 2.6. *If (S,T) is a reduced pair of trees and $x \in F(p+1)$ is a generator such that the pair (xS,T) is not reduced, then $\mu(xS,T) = \mu(S,T) - 1$.*

Proof. We need to examine each generator separately:



Case a:

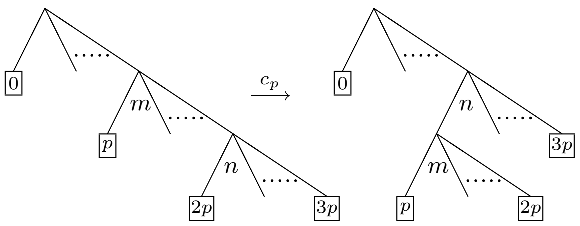
If $\wedge_m$ is reducible in (aS,T), then $\boxed{0}, \ldots, \boxed{p}$ are all empty sub-trees and $m = 0$ and $n = 1$. Since $\wedge_0$ is reducible, $\wedge_1$ must be the parent of $\wedge_0$ in

T. Therefore, only the first two carets will be affected by the removal of $\wedge_0$ and

$$\begin{aligned}\Delta\omega &= \mu\left(aS,T\right) - \mu\left(S,T\right)\\ &= \mu_{(aS,T)}(\wedge_0) - \mu_{(S,T)}(\wedge_0) + \mu_{(aS,T)}(\wedge_1) - \mu_{(S,T)}(\wedge_1)\\ &= 0 - \mu(\mathcal{L}_\emptyset,\mathcal{L}_\emptyset) + \mu(\mathcal{L}_\emptyset,\mathcal{L}_\emptyset) - \mu(\mathcal{R}_*,\mathcal{L}_\mathrm{L})\\ &= -1.\end{aligned}$$

Case $\bar{\mathbf{a}}$: Reversing the arrow in the previous figure, we have a representative of $\bar{a}$. If $\wedge_n$ is reducible in $(\bar{a}S,T)$, then $\boxed{p},\ldots,\boxed{2p}$ are all empty sub-trees and $\wedge_n$ is the rightmost $\wedge$-node in the trees. If $m=0$ then (S,T) represents $\bar{a}$ and $(\bar{a}S,T)$ is the identity so $\Delta\omega=-1$. If $m>0$, $t_m=\tau_T(\wedge_m)$ must be type $\mathcal{L}_\mathrm{L}$ or $\mathcal{R}_\emptyset$ since $\wedge_n$ is the only child of $\wedge_m$. Only the rightmost two carets will be affected by the removal of $\wedge_n$ and

$$\begin{aligned}\Delta\omega &= \mu_{(\bar{a}S,T)}(\wedge_n) - \mu_{(S,T)}(\wedge_n) + \mu_{(\bar{a}S,T)}(\wedge_m) - \mu_{(S,T)}(\wedge_m)\\ &= 0 - \mu(\mathcal{L}_\mathrm{L},\mathcal{R}_\emptyset) + \mu(\mathcal{L}_\mathrm{L},t_m) - \mu(\mathcal{L}_\mathrm{L},t_m)\\ &= -1.\end{aligned}$$



Case $\mathbf{c_p}$:

If $\wedge_m$ (in the above diagram) is reducible in (c_pS,T), then $\boxed{p},\ldots,\boxed{2p}$ are all empty sub-trees and $n=m+1$. The type of $\wedge_n$ depends only on it successor children, so its type is unchanged by the action of the generator. On the other hand, $\wedge_{m-1}$ may change type in T if it is the parent of $\wedge_m$. If $\wedge_{m-1}$ changes type, it must be a change from type $\mathcal{M}_p^i$ to type $\mathcal{M}_\emptyset^i$, with $i<p$, or from $\mathcal{R}_p$ to type $\mathcal{R}_*$. In S, $\wedge_{m-1}$ is either the root node or the leftmost node of one of the subtrees $\boxed{1},\ldots,\boxed{p-1}$. In either case, using Table 2 it is clear that $\mu_{(c_pS,T)}(\wedge_{m-1}) - \mu_{(S,T)}(\wedge_{m-1}) = 0$. Therefore,

$$\begin{aligned}\Delta\omega &= \mu_{(c_pS,T)}(\wedge_m) - \mu_{(S,T)}(\wedge_m) + \mu_{(c_pS,T)}(\wedge_{m-1}) - \mu_{(S,T)}(\wedge_{m-1})\\ &= 0 - \mu(\mathcal{R}_\emptyset,\mathcal{M}_\emptyset^p) + 0\\ &= -1.\end{aligned}$$

Case $\bar{\mathbf{c}}_\mathbf{p}$: Using the previous figure, if $\wedge_n$ is reducible in $(\bar{c}_pS,T)$, then the subtrees $\boxed{2p},\ldots,\boxed{3p}$ are all empty sub-trees and $\wedge_n$ is the rightmost $\wedge$-node in the tree pair so

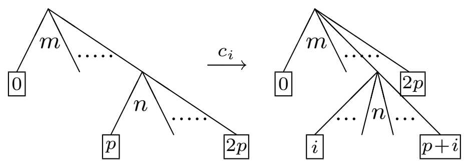
$$\mu_{(\bar{c}_p S,T)}(\wedge_n) - \mu_{(S,T)}(\wedge_n) = 0 - \mu(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) = 0.$$

If $\wedge_m$ has no child successors in S, then $m = n-1$ and $t_m = \tau_T(\wedge_m)$ must be either $\mathcal{L}_\mathrm{L}$ or $\mathcal{R}_\emptyset$, so

$$\begin{aligned}\Delta\omega &= \mu_{(\bar{c}_p S,T)}(\wedge_m) - \mu_{(S,T)}(\wedge_m) \\ &= \mu(\mathcal{R}_\emptyset, t_m) - \mu(\mathcal{M}^p_\emptyset, t_m) \\ &= -1.\end{aligned}$$

If $\wedge_m$ has a child successor, then $\wedge_{n-1}$ is unchanged by the action of the generator and $\wedge_m$ changes from type $\mathcal{M}^p_j$ to type $\mathcal{R}_j$ for some $j < p$. The type of $\wedge_m$ in T may be any type (other than $\mathcal{L}_\emptyset$) so using Table 2, for any type t_m we have

$$\begin{aligned}\Delta\omega &= \mu_{(\bar{c}_p S,T)}(\wedge_m) - \mu_{(S,T)}(\wedge_m) \\ &= \mu(\mathcal{R}_j, t_m) - \mu(\mathcal{M}^p_j, t_m) \\ &= -1.\end{aligned}$$



Case $\mathbf{c_i}$:

If $\wedge_n$ (in the figure above) is reducible in $(c_i S, T)$, then $\boxed{i}, \ldots, \boxed{p+i}$ are all empty sub-trees and $\wedge_n$ is any right type except $\mathcal{R}_j$ when $j \geq i$ so using Table 2,

$$\mu_{(c_i S,T)}(\wedge_n) - \mu_{(S,T)}(\wedge_n) = 0 - \mu(\mathcal{R}_*, \mathcal{M}^i_\emptyset) = -1.$$

The type of $\wedge_m$ is fixed, but the type of $\wedge_{n-1}$ may change in T. If the removal of $\wedge_n$ changes the type of $\wedge_{n-1}$, then $t_{n-1} = \tau(\wedge_{n-1})$ must change from type $\mathcal{M}^j_i$ to type $\mathcal{M}^j_{i_1}$ or $\mathcal{M}^j_\emptyset$, where $i < i_1$, or from $\mathcal{R}_i$ to $\mathcal{R}_*$ where if $t_{n-1} = \mathcal{R}_{i_1}$ then $i < i_1$. In S, $\wedge_{n-1}$ must be type $\mathcal{L}_\mathrm{L}$ (if $m = n-1$) or, by Lemma 1.11, type $\mathcal{M}^{j_1}_\emptyset$ with $j_1 < i$. In any of these cases, we need to show that $\Delta\omega_{n-1} = \mu_{(c_i S,T)}(\wedge_{n-1}) - \mu_{(S,T)}(\wedge_{n-1}) = 0$.

If $\tau_S(\wedge_{n-1}) = \mathcal{L}_\mathrm{L}$ then $\Delta\omega_{n-1} = 0$ since $\mu(\mathcal{L}_\mathrm{L}, t) = \mu(\mathcal{L}_\mathrm{L}, t')$ as long as t and t' are the same basic types. If $\tau_S(\wedge_{n-1}) = \mathcal{M}^{j_1}_\emptyset$ then, using the fact that $j_1 < i < i_1 \leq j$, the change in weight, $\Delta\omega_{n-1}$, must be

$$\mu(\mathcal{M}^{j_1}_\emptyset, \mathcal{M}^j_\emptyset) - \mu(\mathcal{M}^{j_1}_\emptyset, \mathcal{M}^j_i) = 0,$$

$$\mu(\mathcal{M}^{j_1}_\emptyset, \mathcal{M}^j_{i_1}) - \mu(\mathcal{M}^{j_1}_\emptyset, \mathcal{M}^j_i) = 0$$

or

$$\mu(\mathcal{M}_{\emptyset}^{j_1}, \mathcal{R}_*) - \mu(\mathcal{M}_{\emptyset}^{j_1}, \mathcal{R}_i) = 0.$$

Case $\bar{\mathbf{c}}_{\mathbf{i}}$: If $\wedge_n$ in the figure above is reducible in $(\bar{c}_i S, T)$, then $\boxed{p}, \ldots, \boxed{2p}$ are all empty sub-trees and $\wedge_n$ is the rightmost $\wedge$-node in the trees. In T, the parent of $\wedge_n$ is a right $\wedge$-node and removing $\wedge_n$ will not change its type. Similarly, $\wedge_m$ is unchanged in $\bar{c}_i S$ so

$$\begin{aligned}\Delta\omega &= \mu_{(\bar{c}_i S,T)}(\wedge_n) - \mu_{(S,T)}(\wedge_n) \\ &= 0 - \mu(\mathcal{M}_{\emptyset}^{i}, \mathcal{R}_{\emptyset}) \\ &= -1.\end{aligned}$$

□

Theorem 2.7. *For any reduced pair (S, T) and any generator x in $F(p+1)$,*

$$\mu\,(S, T) = \mu\,(xS, T) \pm 1.$$

Proof. It is clear from Theorem 2.5 and Theorem 2.6 that if a caret is added or exposed in the process of applying a generator to the domain tree of (S, T) that $\mu\,(S, T) = \mu\,(xS, T) \pm 1$. If (S, T) and (xS, T) are both reduced then by Theorem 2.3 we need only show that for the $\wedge$-node, $\wedge_i$, described in Theorem 2.3, $\mu_{(S,T)}(\wedge_i) = \mu_{(xS,T)}(\wedge_i) \pm 1$. In order to perform this calculation, we need only demonstrate that in Table 2 the label sets of the column representing $\mu_{(S,T)}(\wedge_i)$ differ by at most one element from the entries in the column representing $\mu_{(xS,T)}(\wedge_i)$. Each row entry of the $\mathcal{L}_{\mathrm{L}}$ column differs by one element from each if the entries in the same row of the $\mathcal{R}_{\emptyset}$, $\mathcal{R}_{\mathrm{R}}$ and $\mathcal{R}_j$ columns, and so on, for each $\wedge$-node type change described in Theorem 2.3. □

2.2. **The fourth condition: at least one generator reduces length.** In order to show that the weight of a pair of $\wedge$-trees is the same as the length of the corresponding element of $F(p+1)$, we need to show that there is always at least one generator x that gives $\mu\,(xS, T) - \mu\,(S, T) = -1$. For any such generator x we say that x *reduces the weight* of (S, T). From Theorems 2.5 and 2.6, we need only test situations where x can be applied to the $\wedge$-tree S without needing to add carets and where (xS, T) is not reducible.

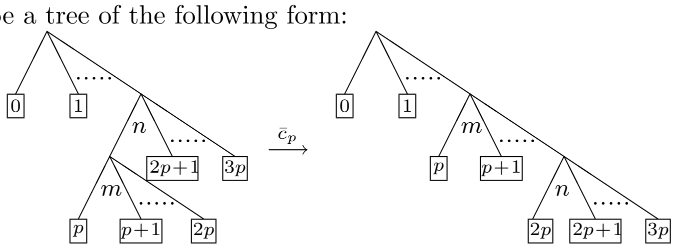
Theorem 2.8. *If (S, T) is a reduced pair of $\wedge$-trees representing an element of $F(p+1)$, and a generator x can be applied to S without adding carets, there is at least one generator x' where $\Delta\omega = \mu\,(x'S, T) - \mu\,(S, T) = -1$.*

To prove this theorem, we must either test each of the six major types of generators (a, c_p, c_i, and their inverses), or for any given pair (S, T), we must choose x based on the arrangement and types of the nodes in (S, T).

In all the following proofs, we will make extensive use of Theorem 2.3 in order to simplify the calculation of weight.

Lemma 2.9. *Assuming (S,T) is reduced, if $\bar{c}_p$ can be applied to S without adding carets, then one of the generators $\bar{c}_p$ or a reduces the weight of (S,T).*

Proof. S must be a tree of the following form:



If one of the subtrees $\boxed{p+1},\ldots,\boxed{2p}$ is not empty then it is clear from Table 2 that

$$\Delta\omega = \mu(\mathcal{R}_j, t_m) - \mu(\mathcal{M}_j^p, t_m) = -1$$

for any $t_m = \tau_T(\wedge_m)$.

If all the successor subtrees $\boxed{p+1},\ldots,\boxed{2p}$ are empty, then $\wedge_m$ changes from type $\mathcal{M}_\emptyset^p$ to type $\mathcal{R}_\emptyset$ or $\mathcal{R}_\mathrm{R}$. Again from Table 2,

$$\Delta\omega = \mu(\mathcal{R}_\mathrm{N}, t_m) - \mu(\mathcal{M}_\emptyset^p, t_m) = -1$$

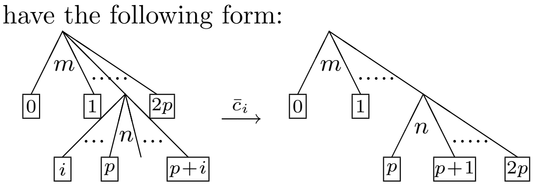
except in the cases where $t_m = \mathcal{R}_\mathrm{R}$ and when $t_m = \mathcal{R}_\emptyset$ but $\tau_{(S,T)}(\wedge_n) \neq (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$. In the case that $\bar{c}_p$ does not decrease the weight of (S,T), if we apply a to S then $\wedge_n$ changes to a node of type $\mathcal{L}_\mathrm{L}$ and when $\tau_{(S,T)}(\wedge_n) \neq (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$,

$$\Delta\omega = \mu(\mathcal{L}_\mathrm{L}, \mathcal{R}_\mathrm{N}) - \mu(\mathcal{R}_\mathrm{N}, \mathcal{R}_\mathrm{N}) = -1.$$

□

Lemma 2.10. *Assuming (S,T) is reduced, if for some $i < p$, $\bar{c}_i$ can be applied to S without adding carets, then there is at least one generator x where $\Delta\omega = \mu(xS,T) - \mu(S,T) = -1$.*

Proof. S must have the following form:



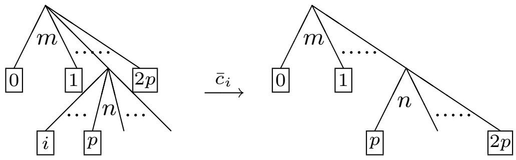
Assume one of the subtrees $\boxed{p+1},\ldots,\boxed{p+i}$ are not empty then, in S, $\wedge_n$ is type $\mathcal{M}_j^i$ for some $j \leq i$. Applying $\bar{c}_i$ to S gives

$$\Delta\omega = \mu(\mathcal{R}_j, t_n) - \mu(\mathcal{M}_j^i, t_n) = -1$$

for any $t_n = \tau_T(\wedge_n)$.

Assume now that none of the successor children, (except possibly the right child) of $\wedge_m$ have a successor child. If we choose i so that the i^{th} child of the root is the last successor child that is not of type $\mathcal{R}$, this child is type $\mathcal{M}^i_\emptyset$ and all the subtrees $\boxed{p+1}, \ldots, \boxed{2p-1}$ are empty. Applying $\bar{c}_i$ to S gives $\Delta\omega = \mu(\mathcal{R}_*, t_n) - \mu(\mathcal{M}^i_\emptyset, t_n) = -1$ unless $t_n = \tau_T(\wedge_n)$ is $\mathcal{R}_\emptyset$, $\mathcal{R}_{\mathrm{R}}$, $\mathcal{R}_{j_2}$ or $\mathcal{M}^{i_2}_{j_2}$ where $j_2 > i$. We now need to find a generator that reduces the weight of (S, T) in these cases.

By Lemma 2.9, we need only examine the cases where $\boxed{2p}$ is empty or where if $\boxed{2p}$ is not empty, the root $\wedge$-node of the $\boxed{2p}$ subtree has no left child.



Case $t_n = \mathcal{R}_\emptyset$: If $\boxed{2p}$ is empty, then $\wedge_n$ is the last $\wedge$-node of (S, T) and applying $\bar{c}_i$ gives

$$\Delta\omega = \mu(\mathcal{R}_\emptyset, \mathcal{R}_\emptyset) - \mu(\mathcal{M}^i_\emptyset, \mathcal{R}_\emptyset) = -1$$

if $\wedge_n$ is not exposed in $(\bar{c}_i S, T)$ (if $\wedge_n$ is removable then $\Delta\omega = -1$ by Theorem 2.6).

If $\boxed{2p}$ is not empty, then we can assume $\wedge_{n+1}$ is the root of $\boxed{2p}$ and since (S, T) is reduced, $s_{n+1} = \tau_S(\wedge_{n+1})$ must be type $\mathcal{R}_{\mathrm{R}}$ or $\mathcal{R}_j$ for some $j < p$. Applying a to S gives

$$\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, \mathcal{R}_\emptyset) - \mu(s_{n+1}, \mathcal{R}_\emptyset) = -1.$$

Case $t_n = \mathcal{R}_{\mathrm{R}}$: Since $\wedge_n$ is not the last caret of (S, T), $\boxed{2p}$ is not empty, and $t_{n+1} = \tau_T(\wedge_{n+1})$ must be $\mathcal{R}_\emptyset$ or $\mathcal{R}_{\mathrm{R}}$. Applying a to S,

$$\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, t_{n+1}) - \mu(\mathcal{R}_*, t_{n+1}) = -1.$$

Case $t_n = \mathcal{R}_{j_2}$: Again $\wedge_n$ is not the last caret of (S, T), $\boxed{2p}$ is not empty and by Lemma 1.12, $\tau_{(S,T)}(\wedge_{n+1}) = (\mathcal{R}_*, \mathcal{M}^{i_3}_*)$ where $i_3 \geq j_2 > i$.

When $\wedge_{n+1}$ has a right child in T, applying a gives

$$\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, \mathcal{M}^{i_3}_{j_3}) - \mu(\mathcal{R}_*, \mathcal{M}^{i_3}_{j_3}) = -1.$$

If $\wedge_{n+1}$ has no children in T, then $\wedge_{n+1}$ is removable in $(c_{i_3} S, T)$ (and $\Delta\omega = -1$), and if $\wedge_{n+1}$ only has a child of type $\mathcal{R}_*$ in S. On the other hand, if $\tau_S(\wedge_{n+1}) = \mathcal{R}_j$, the first child of $\wedge_{n+1}$ in S, $\wedge_q$, is a $\wedge$-node of type

$\mathcal{M}_*^j$. If $j > i_3$ then applying c_{i_3} to S will cause $\wedge_{n+1}$ to be removable (since $j > i$) in $(c_{i_3}S, T)$ giving $\Delta\omega = -1$. If $j \leq i_3$, applying a to S gives

$$\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, \mathcal{M}_\emptyset^{i_3}) - \mu(\mathcal{R}_j, \mathcal{M}_\emptyset^{i_3}) = -1.$$

Case $t_n = \mathcal{M}_{j_2}^{i_2}$: Similar to the previous case, $\tau_{(S,T)}(\wedge_{n+1}) = (\mathcal{R}_*, \mathcal{M}_*^{i_3})$ so by applying a or c_{i_3} in the appropriate situation will give $\Delta\omega = -1$.

Therefore, if $\bar{c}_i$ can be applied to S, either $\bar{c}_i$, a or c_{i_3}, for some $i_3 > i$, reduces the weight of (S, T). □

Based on the results of the previous two lemmas, in any pair (S, T) where $\bar{c}_i$ (for $0 < i \leq p$) can be applied to S, we can find a generator x that reduces the weight of (S, T). Therefore, for the remaining cases, we need only to examine pairs where S has one of the two forms shown in Figure 2.2.

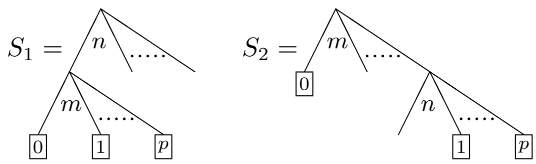


FIGURE 5. Domain trees not reduced by $\bar{c}_i$.

Lemma 2.11. *If* (S_1, T) *is reduced then* $\Delta\omega = \mu(S_1, T) - \mu(\bar{a}S_1, T) = -1$.

Proof. By the definition of S_1 in Figure 2.2, $\wedge_n$ is the last caret of (S_1, T) and must be type $t_n = \mathcal{L}_{\mathrm{L}}$, $\mathcal{R}_\emptyset$ or $\mathcal{M}_\emptyset^i$ in T. Therefore,

$$\Delta\omega = \mu(\mathcal{R}_\emptyset, t_n) - \mu(\mathcal{L}_{\mathrm{L}}, t_n) = -1$$

for these three choices for t_n. □

Lemma 2.12. *If* (S_2, T) *is reduced, there is at least one generator of* $F(p+1)$ *where* $\Delta\omega = -1$.

Proof. It is important to note that $m = n - 1$ in S_2. Applying a to S_2 gives $\Delta\omega = \mu(\mathcal{L}_{\mathrm{L}}, t_n) - \mu(\mathcal{R}_*, t_n)$ which is -1 unless $\tau_{(S_2,T)}(\wedge_n) = (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$ or $t_n = \tau(\wedge_n)$ is type $\mathcal{L}_{\mathrm{L}}$ or $\mathcal{M}_\emptyset^{i_1}$ where $i_1 < j$ when $\tau_{S_2}(\wedge_n) = \mathcal{R}_j$.

Case $t_n = \mathcal{L}_{\mathrm{L}}$: If $\boxed{0}$ is empty then $m = 0$ and $\wedge_0$ will be exposed in (aS_2, T) so $\Delta\omega = -1$ by Theorem 2.6. If $\boxed{0}$ is not empty, $t_m = \tau_T(\wedge_m)$ is either $\mathcal{L}_{\mathrm{L}}$ or $\mathcal{M}_\emptyset^i$. Applying $\bar{a}$ to S_2 gives

$$\Delta\omega = \mu(\mathcal{R}_{\mathrm{N}}, t_m) - \mu(\mathcal{L}_{\mathrm{L}}, t_m) = -1.$$

Case $\tau_{(S_2,T)}(\wedge_n) = (\mathcal{R}_\emptyset, \mathcal{R}_\emptyset)$: In this case, $\boxed{p}$ must be empty and $\wedge_m$ must be the left child of $\wedge_n$ in T, since (S,T) is reduced. Therefore, $m \neq 0$ so $\boxed{0}$ is not empty and $\tau_{(S_2,T)}(\wedge_m) = (\mathcal{L}_\mathrm{L}, \mathcal{M}_\emptyset^p)$. Applying $\bar{a}$ to S_2 gives

$$\Delta\omega = \mu(\mathcal{R}_\emptyset, \mathcal{M}_\emptyset^p) - \mu(\mathcal{L}_\mathrm{L}, \mathcal{M}_\emptyset^p) = -1.$$

Case $t_n = \mathcal{M}_\emptyset^{i_1}$: If $\tau_{S_2}(\wedge_n)$ is either $\mathcal{R}_\emptyset$ or $\mathcal{R}_\mathrm{R}$ then all the subtrees $\boxed{1}, \ldots, \boxed{p-1}$ must be empty. If $\wedge_n$ is childless in T, then $\wedge_n$ will be exposed in the pair $(c_{i_1}S_2, T)$. If $\wedge_n$ has a predecessor child in T, its immediate predecessor, $\wedge_m$, must be type $\mathcal{M}_\emptyset^{i_2}$ where $0 < i_2 \leq p$. Since $m > 0$ in T, the subtree $\boxed{0}$ must not be empty in S and applying $\bar{a}$ to S_2 gives

$$\Delta\omega = \mu(\mathcal{R}_\mathrm{N}, \mathcal{M}_\emptyset^{i_2}) - \mu(\mathcal{L}_\mathrm{L}, \mathcal{M}_\emptyset^{i_2}) = -1.$$

If $\tau_{S_2}(\wedge_n) = \mathcal{R}_j$ with $i_1 < j$, the first nonempty subtree of $\wedge_n$ in S must be $\boxed{j}$. As before, if $\wedge_n$ is childless in T then $\wedge_n$ will be exposed in $(c_{i_1}S_2, T)$; if $\wedge_n$ is not childless in T then $m > 0$ and $\tau(\wedge_m) = \mathcal{M}_\emptyset^{i_2}$. Applying $\bar{a}$ to S_2 gives

$$\Delta\omega = \mu(\mathcal{R}_\mathrm{R}, \mathcal{M}_\emptyset^{i_2}) - \mu(\mathcal{L}_\mathrm{L}, \mathcal{M}_\emptyset^{i_2}) = -1.$$

□

Proof of Theorem 2.8. Based on the results of Theorem 2.6 and Lemmas 2.9 – 2.12, it is clear that there is always one generator that makes the weight of (xS,T) one less than the weight of (S,T). □

2.3. **Conclusion.**

Proof of Theorem 2.2. We have shown that for any $w \in F(p+1)$ with reduced tree representation (S,T), $\mu\,(S,T)$ satisfies all the following properties:

$\mu(w) = 0$ iff $w = \mathrm{id}_{F(p+1)}$ by Theorem 1.16, and from Theorem 2.7, we know that for any generator x of $F(p+1)$, $\mu\,(xS,T) \geq \mu\,(S,T) - 1$. Finally, Theorem 2.8 proves that there is at least one generator where $\mu\,(xS,T) = \mu\,(S,T) - 1$. Therefore, by Lemma 2.1, $\ell(w) = \mu\,(S,T)$. □

Thus we measure word length with respect to the standard finite generating set effectively by summing the weights of the caret pairs. To find a minimal length representative of a group element w given by a tree pair diagram, we simply find successive generators $g_1, g_2, \ldots, g_n$ which reduce word length until we reach length 0 and then $w = \bar{g_n} \ldots \bar{g_2}\bar{g_1}$ will be a minimal length representative. For a word w given in terms of the infinite generating set in normal form, we use the process of leaf exponents from [2] to construct a tree pair representative and then use the above process on the tree pair. Similarly, for a word given in terms of the finite generating set, we can rewrite the word into a normal form in the infinite generating set using the relations and then construct the tree pair to measure the length.

References

1. Kenneth S. Brown, *Finiteness properties of groups*, Proceedings of the Northwestern conference on cohomology of groups (Evanston, Ill., 1985), vol. 44, J. Pure Appl. Algebra, no. 1-3, 1987, pp. 45–75. MR MR885095 (88m:20110)
2. J. W. Cannon, W. J. Floyd, and W. R. Parry, *Introductory notes on Richard Thompson's groups*, Enseign. Math. (2) **42** (1996), no. 3-4, 215–256. MR MR1426438 (98g:20058)
3. S. Blake Fordham, *Minimal length elements of Thompson's group F*, Ph.D. thesis, Brigham Young University, 1995.
4. ______, *Minimal length elements of Thompson's group F*, Geom. Dedicata **99** (2003), 179–220. MR MR1998934 (2004g:20045)
5. Graham Higman, *Finitely presented infinite simple groups*, Department of Pure Mathematics, Department of Mathematics, I.A.S. Australian National University, Canberra, 1974, Notes on Pure Mathematics, No. 8 (1974). MR MR0376874 (51 #13049)
6. Melanie Stein, *Groups of piecewise linear homeomorphisms*, Trans. Amer. Math. Soc. **332** (1992), no. 2, 477–514. MR MR1094555 (92k:20075)

1394 N. 770 W., Orem, Utah, 84057, USA
E-mail address: blake@math.byu.edu

Department of Mathematics, The City College of New York & The CUNY Graduate Center, New York, NY 10031
E-mail address: cleary@sci.ccny.cuny.edu