

GROWTH OF GROUPS DEFINED BY AUTOMATA

ASHLEY S. DOUGHERTY, LYDIA R. KINDELIN, AARON M. REAVES, ANDREW J. WALKER,
AND NATHANIEL F. ZAKAHI

ABSTRACT. Nekrashevych conjectured that iterated monodromy groups of quadratic polynomials with a pre-periodic kneading sequence have intermediate growth. In this paper we prove intermediate growth for two such groups. We also provide a new proof of intermediate growth for a class of groups containing the Grigorchuk group, previously known to have intermediate growth.

1. INTRODUCTION

Finitely generated groups of infinite order grow at many different rates. Growth functions are more commonly polynomial or exponential but there are rarer cases of intermediate growth. Grigorchuk found the first group of intermediate growth in the 1980's and these groups have been studied ever since.

Grigorchuk's group is generated by an automaton and these mathematical structures are convenient for finding groups of intermediate growth. Nekrashevych provides a process in his monograph *Self-Similar Groups* for producing an automaton from any given post-critically finite polynomial. He also conjectured that a specific kind of post-critically finite polynomial would produce an automaton which would generate a group of intermediate growth. We have considered this conjecture and found two more groups of intermediate growth using Nekrashevych's procedure.

The automata produced by Nekrashevych's process are of a specific form and the Grigorchuk group does not come from such an automaton. We looked at automata with similar properties to that of the Grigorchuk group and give a new proof that a known class of automata will produce groups of intermediate growth.

In order to prove intermediate growth of groups we modeled our arguments on those from the paper *On the Growth of Iterated Monodromy Groups* Bux and Pérez [2] with the use of a weight system to represent the length of words. This method can be used to prove intermediate growth for both, groups produced by polynomials in Nekrashevych's process, and groups generated by any automaton. In this paper we also discuss some of the obstacles we encountered with this method for proving intermediate growth.

2. BACKGROUND

2.1. Growth.

Definition 2.1. For a finite generating set Σ of a group G and a map $\tilde{\ell} : \Sigma \rightarrow \mathbb{R}^+$, the *ball of radius n* , $B_G(n)$ is the set of all elements $g \in G$ represented by a word $w = x_1^{\epsilon_1} x_2^{\epsilon_2} \cdots x_m^{\epsilon_m}$, $x_i \in \Sigma$, $\epsilon_i = \pm 1$, with $\sum_{i=1}^n \tilde{\ell}(i) \leq n$.

Definition 2.2. The *growth function* $g(n) : \mathbb{N} \rightarrow \mathbb{N}$ of G is defined by $g(n) = |B_G(n)|$.

Definition 2.3. Let $f, g \in \mathbb{N} \cup \{0\} \rightarrow \mathbb{N} \cup \{0\}$. We say that $f \preceq g$ if there are constants $A, B, C \in \mathbb{N}$ such that $f(n) \leq Ag(Bn) + C$, for all $n \in \mathbb{N} \cup \{0\}$.

Definition 2.4. Let G be a group with the finite generating set S and growth function $g(n)$. We say G has *polynomial growth* if there exists some polynomial $p(n)$ such that $g(n) \preceq p(n)$. We say G has *exponential growth* if there exists some $a > 1$ such that $a^n \preceq g(n)$. A group G has *intermediate growth* if $p(n) \preceq g(n) \preceq a^n$ for all polynomials p and some $a > 1$. Growth is not dependent on the generating set but is a property of the group in general.

2.2. Automata.

2.2.1. *Defining Automata.* Except for the definition of the infinite rooted n -ary tree, the following definitions come from Nekrashevych Nekt. The tree definition is general knowledge; our definition comes from Terpstra[3].

Definition 2.5. For an integer $n \geq 2$ define \mathcal{T}^n to be the *infinite rooted n -ary tree*. The vertices of \mathcal{T}^n are finite sequences of elements (or possibly \emptyset) from $\{0, 1, \dots, n-1\}$, and the edges of \mathcal{T}^n connect pairs of vertices whose sequences differ in length by 1, and in which the shorter sequence can be obtained by erasing the last term of the longer sequence. For example, vertices $v_i = (011)$ and $v_j = (0110)$ would be connected by an edge, while $v_l = (010)$ and $v_j = (0110)$ would not. The empty sequence defines the *root* vertex denoted by \emptyset .

Definition 2.6. Let Γ be a simple graph. An *automorphism* of Γ is a bijection of the vertices that preserves the adjacency and non-adjacency.

Definition 2.7. An *alphabet* is a set X . The *free monoid on X* , denoted X^* , is the set (possibly empty) of all n -tuples.

Definition 2.8. An *automaton* A over the alphabet X is given by

- (1) a *set of the states*, usually also denoted by A
- (2) a map $\tau : A \times X \rightarrow X \times A$

Definition 2.9. It is convenient to define automata using their *Moore diagrams*. A Moore diagram is a directed labeled graph with the vertices identified with the states of the automaton. If $\tau(q \cdot x) = y \cdot p$, then we write $q \cdot x = y \cdot p$, and we have an arrow starting in q , ending in p and labeled by (x, y) . So, the arrows show the state transitions, and their labels show the input and the output.

2.2.2. *Groups Generated by Automata.* The group generated by the automaton A is the group $\langle A \rangle$ generated by the transformations defined on all the states of A .

For example, consider the automaton A over the alphabet X containing state q . Then q is represented by an n -tuple where the first coordinate represents how the state acts with 0, the second coordinate is how the state acts with 1, and so on through $n-1$ actions. So, if $q \cdot 0 = 0 \cdot p$ and

$q \cdot 1 = 1 \cdot r$, then $q = (p, r)$. This bracket notation means that when the state q acts on the n -ary tree, it does nothing to the 0 but produces a p at the next vertex of the tree and does nothing to 1 but produces an r at the next vertex. This notation is also an easier way to multiply elements of the group. If $q = (p, r)$ and $p = (1, q)$ then $qp = (p, r)(1, q) = (p, rq)$. Another important aspect of the bracket notation is the ease in representing certain automorphisms of the tree. Consider an automorphism $\alpha := (\alpha_L, \alpha_R) \in \text{Aut}(T^{(2)})$ where α preserves each of the two subtrees and acts in $T_L^{(2)}$ as α_L and in $T_R^{(2)}$ as α_R . Now, consider $\tau \in \text{Aut}(T^{(2)})$, called the swap, that interchanges $T_L^{(2)}$ and $T_R^{(2)}$. So, $\tau(\alpha_L, \alpha_R) = (\alpha_R, \alpha_L)\tau$. For $\tau \in \text{Aut}(T^{(3)})$, $\tau(\alpha_L, \alpha_C, \alpha_R) = (\alpha_R, \alpha_L, \alpha_C)\tau$.

2.2.3. Nekrashevych's Process for Creating Groups from Polynomials.

Definition 2.10. An *active state* a of an automaton A is a state such that $a \times x \rightarrow y \times a$ for some $x \neq y \in X$. Because of this, a state in an automaton may be seen as a string transformation on strings of the alphabet, with an active state a transforming x to y and a non-active state as the identity transformation.

Definition 2.11. For an automaton A with only a single active state and a single arrow leading to and from each non-active state that does not go to the identity state, the *kneading sequence* of A is the sequence of $x_i \in X$ labeling the arrows not leading to the identity state. The first arrow, with corresponding x_1 is the arrow entering the active state, and each successive arrow is the arrow entering the state the previous arrow began at. A *periodic kneading sequence* is when the kneading sequence is a loop connecting all the labels of the arrows of the Moore diagram. A *pre-periodic kneading sequence* is when the kneading sequence has a straight path for at least one action and then is a loop connecting the rest of the labels.

Definition 2.12. Let $p : \mathbb{C} \rightarrow \mathbb{C}$ be a complex polynomial such that $p(z) = a_z^n + a_{n-1}z^{n-1} + \dots + a_1z + a_0$. A *critical point* of the polynomial is a solution of $p'(z) = 0$. A *post-critical point* is a point $p(p(p(\dots(c))))$ where c is a critical point. We say that p is *post-critically finite* if the set P of all post-critical points is finite.

In Nekrashevych's monograph, *Self-Similar Groups*, he provides a process for constructing automata from post-critically finite polynomials. These automata can be used to define iterated monodromy groups. We only describe the process for finding these groups, but a thorough discussion of them can be found in the monograph [1]. It is important to note that for any post-critically finite polynomial $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, if there exists an f -invariant spider as defined below, there will always exist an automaton which generates the action of $\text{IMG}(f)$.

Conjecture 2.13. *Nekrashevych conjectured that every second degree post-critically finite polynomial with a pre-periodic kneading sequence produces an iterated monodromy group of intermediate growth.*

Definition 2.14. A *spider* \mathcal{S} is a collection of disjoint simple curves in the complex plane connecting each of the post-critical points to infinity (since the number of post-critical points is finite, a spider will always exist). For each $z \in P$, we call the curve connecting z to infinity γ_z . Each curve γ_z may not pass through any post-critical points other than its initial point at z . A spider \mathcal{S} is said to be *f -invariant* if $f(\mathcal{S})$ is isotopic to a subset of \mathcal{S} through spiders.

Definition 2.15. For a critical point $z \in C$ of local degree d_z , the set $X_z = f^{-1}(\gamma_{f(z)})$ consists of d_z curves connecting z to infinity. The collection $\{X_z \mid z \in C\}$ of such sets of curves is called the *critical portrait* associated to the spider \mathcal{S} and the topological polynomial. The curves of the critical portrait will divide the plane into components, called *sectors*.

For a polynomial f with a finite post-critical set and corresponding f -invariant spider \mathcal{S} , there will exist some critical portrait, \mathcal{C} . Then let $\{S_x \mid x \in \mathbf{X}\}$ be the set of corresponding sectors, where \mathbf{X} , with $|\mathbf{X}| = d$, is the set of labels of the sectors.

We shall construct the automaton $K_{\mathcal{C},f}$ using \mathbf{X} as our alphabet. We label this automaton $K_{\mathcal{C},f}$, as it encodes both the critical portrait \mathcal{C} and the action of the polynomial f . The procedure is as follows:

Part A Take an arbitrary post-critical point $z \in P$. For each $y \in f^{-1}(z)$, if y is not critical, proceed to Part B. Otherwise, skip to Part C.

Part B As y is not critical, the local degree of f around y should be 1, and y should be an internal point of a sector S_x . This information will be encoded by the output and transition functions of $K_{\mathcal{C},f}$:

$$g_z \cdot x = x \cdot g_y,$$

where $g_y = id$ if $y \notin P$.

Part C Since y is critical, it will be on the boundaries of d' sectors. Let these sectors be $S_{x_1}, S_{x_2}, S_{x_3}, \dots, S_{x_{d'}}$, listed in counterclockwise order around y . The set \mathbf{X} , which we already defined as the set of labels of the sectors, will then be the set $\{x_1, x_2, x_3, \dots, x_{d'}\}$. If y is not post-critical, go to Part C-I. If it is post-critical, go to part C-II.

Part C-I Since y is not post critical, we encode the action of a small loop around y by setting:

$$g_z \cdot x_i = x_{i+1(\text{mod } d')} \cdot id$$

for $i = 1, \dots, d'$.

Part C-II Since y is post-critical, there is a path γ_y in the spider \mathcal{S} . We may assume the sectors are labeled in such a way that the curve γ_y is adjacent to S_{x_1} and $S_{x_{d'}}$. Let us then encode this by

$$g_z \cdot x_i = x_{i+1} \cdot id$$

for $i = 1, \dots, d' - 1$ and

$$g_z \cdot x_{d'} = x_1 \cdot g_y.$$

Once this process has been completed for every post-critical point in the set P , we can draw the corresponding Moore diagram.

3. COMPUTATIONS OF ITERATED MONODROMY GROUPS OF POLYNOMIALS

In this section, we compute the iterated monodromy groups for the polynomials $z^2 - 2$, $z^3 + 3z$, $z^3 - 3z$, and $z^3 + \frac{3}{2}z$. We then show they are all of polynomial growth in Proposition 3.1.

3.1. The Iterated Monodromy Group of $\mathbf{p}(z) = z^2 - 2$.

Here, we use a simple example to demonstrate Nekrashevych's procedure for producing iterated monodromy groups. Consider the polynomial $z^2 - 2$. The derivative of $z^2 - 2$ is $2z$, and when set equal to zero, z must equal zero. Therefore, the set of critical points, C , only contains 0. When we substitute 0 into $z^2 - 2$ for z , we get -2 . When -2 is similarly substituted back into the polynomial, we get 2, and 2 will always produce another 2 when substituted for z . Therefore, the

set of post-critical points is $P = \{-2, 2\}$ and $z^2 - 2$ is post-critically finite. Now we will construct the spider \mathcal{S} . Let

$$\gamma_{-2} = \{z \in \mathbb{C} \mid z = -2 + it, t \in \mathbb{R}^+ \cup \{0\}\},$$

$$\gamma_2 = \{z \in \mathbb{C} \mid z = 2 - it, t \in \mathbb{R}^+ \cup \{0\}\}$$

portrayed in the picture below where

$$f(\gamma_{-2}) = \{z \in \mathbb{C} \mid (2 - t^2 - 4it), t \in \mathbb{R}^+ \cup \{0\}\},$$

$$f(\gamma_2) = \{z \in \mathbb{C} \mid (2 - t^2 - 4it), t \in \mathbb{R}^+ \cup \{0\}\}$$



To obtain the line X_0 and complete the critical portrait, we set $X_0 = f^{-1}(\gamma_{f(0)})$. Then, $f^{-1}(\gamma_{f(0)}) = f^{-1}(\gamma_{-2}) = \{z \in \mathbb{C} \mid f(z) = -2 + it, t \in \mathbb{R}^+ \cup \{0\}\}$. Thus:

$$z^2 - 2 = -2 + it$$

$$z^2 = it$$

$$z^2 = te^{\frac{\pi}{2}i}$$

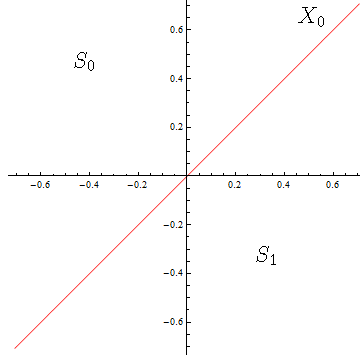
$$te^{\frac{\pi}{2}i} = r^2 e^{2\theta i}$$

$$r^2 = t$$

$$2\theta = \frac{\pi}{2} + 2\pi n, n \in \mathbb{Z}$$

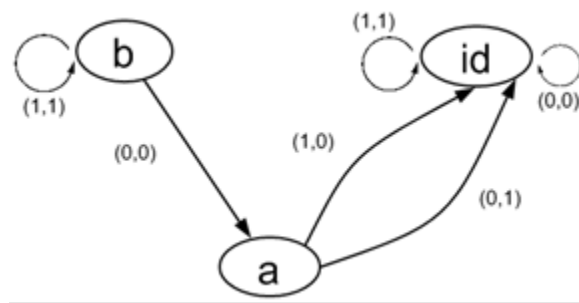
$$\theta = \frac{\pi}{4} + \pi n$$

The last step in labeling the critical portrait is to determine which sector is S_0 and which is S_1 . We want to label the sectors starting with S_0 and moving counterclockwise around X_0 to label S_1 . Since we only have two sectors, one valid labeling of the graph is as follows:



Let the post-critical points -2 and 2 be represented by a and b respectively. This gives our set of states $A = \{a, b, id\}$ and our alphabet $X = \{0, 1\}$.

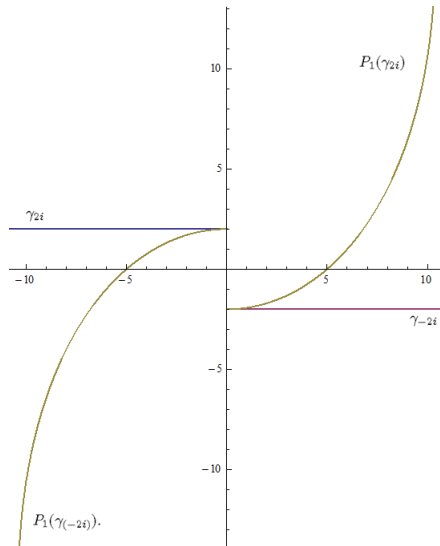
Now let z be the post-critical point -2 . Then $y = f^{-1}(z) = 0$. Note 0 is not a post-critical point, but since it is a critical point, it lies on X_0 . Thus, we set $a \cdot 0 = 1 \cdot id$ and $a \cdot 1 = 0 \cdot id$ as defined by $g_z \cdot x_i = x_{i+1(\text{mod } d')} \cdot 1$ for $i = 1, \dots, d'$ where $S_{x_{d'+1}} = S_{x_1}$. Consider next the other post-critical point 2 . Then $f^{-1}(z) = \{-2, 2\}$. Thus, we must finish the procedure for both $y = 2$ and $y = -2$. For $y = 2$, since 2 is not a critical point and lies in S_1 , we obtain $b \cdot 1 = 1 \cdot b$ by applying $g_z \cdot x = x \cdot g_y$ where $g_y = 1$ if $y \notin P$. For $y = -2$, we apply the same formula, with the exception that -2 lies in S_0 . This gives $b \cdot 0 = 0 \cdot a$. These actions give the following automaton:



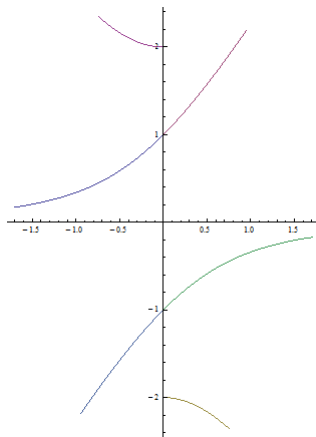
We may also represent the actions of each state of the automaton in bracket notation. The state a acts as the transposition (01) , which we commonly refer to as τ . The state b will act differently on 0 and 1 , but it does not change the number in either case. When it acts on a 0 , it returns the 0 and goes to a , and when it acts on a 1 , it returns the 1 and goes back to itself. Therefore $b = (a, b)$.

3.2. The Iterated Monodromy Group of $p_1(z) = z^3 + 3z$.

A more interesting example comes from the polynomial $p_1(z) = z^3 + 3z$. Carrying out the same procedure explained earlier, we find $C = \{i, -i\}$ and $P = \{2i, -2i\}$. We may then create the following spider with $\gamma_{2i} = \{z \in \mathbb{C} \mid z = 2i - t, t \in \mathbb{R}^+\}$ and $\gamma_{-2i} = \{z \in \mathbb{C} \mid z = t - 2i, t \in \mathbb{R}^+\}$.

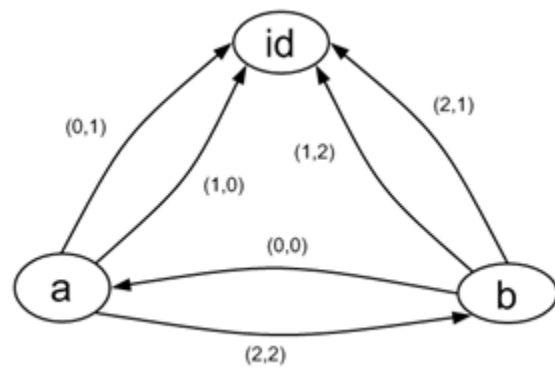


We may note it is a p_1 -invariant spider, with $p_1(\gamma_{2i})$ and $p_1(\gamma_{(-2i)})$ also depicted. We also give the critical portrait, with $X_{(i)}$ and $X_{(-i)}$ depicted.



We assign the post critical points $2i$ and $-2i$ to the states a and b , respectively, and we find the actions to be: $a \cdot 0 = 1 \cdot id, a \cdot 1 = 0 \cdot id, a \cdot 2 = 2 \cdot b$ and $b \cdot 0 = 0 \cdot a, b \cdot 1 = 2 \cdot id, b \cdot 2 = 1 \cdot id$.

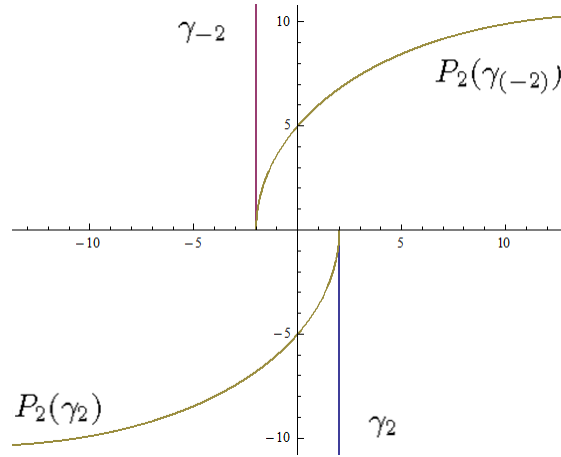
We then have our final automaton, with its Moore diagram depicted here:



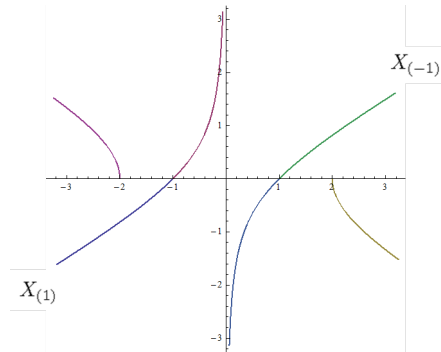
We may then convert the automaton into a group with the previously introduced bracket notation. This gives group elements $a = \alpha(1, 1, b)$ and $b = \beta(a, 1, 1)$, where $\alpha = (01)$ and $\beta = (12)$.

3.3. The Iterated Monodromy Group of $p_2(z) = z^3 - 3z$.

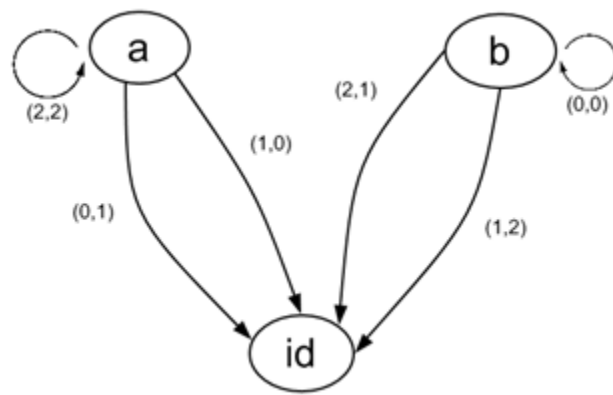
A similar example is the iterated monodromy group of $p_2(z) = z^3 - 3z$. Following the same procedure, we have $C = \{1, -1\}$ and $P = \{-2, 2\}$. We may graph $\gamma_2(t) = 2 + ti$, $\gamma_{-2}(t) = -2 - ti$, along with $p_2(\gamma_2)$ and $p_2(\gamma_{-2})$ to create a p -invariant spider, figure.



The curves $X_{(1)}$ and $X_{(-1)}$ are graphed to form the critical portrait shown below.



This critical portrait leads to the following actions of the group elements a and b . With a labeling post critical point 2 and b labeling -2 , the actions of the group are: $a \cdot 0 = 1 \cdot id$, $a \cdot 1 = 0 \cdot id$, $a \cdot 2 = 2 \cdot a$ and $b \cdot 0 = 0 \cdot b$, $b \cdot 1 = 2 \cdot id$, $b \cdot 2 = 1 \cdot id$. Here we show the automaton:

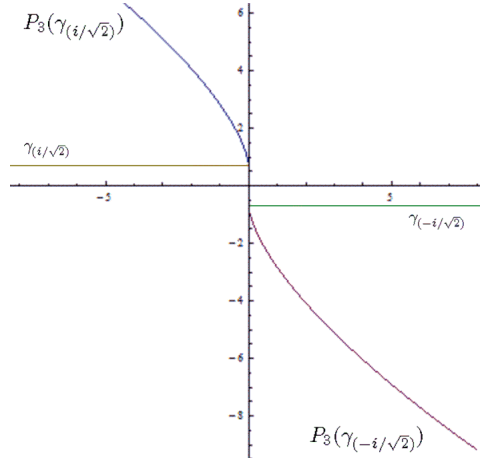


In bracket notation, these group elements are represented as $a = \alpha(1, 1, a)$ and $b = \beta(b, 1, 1)$, where α and β are cycles $\alpha = (01)$ and $\beta = (12)$.

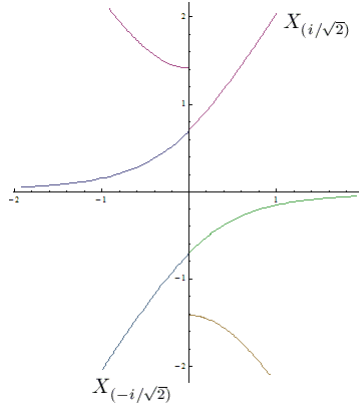
3.4. The Iterated Monodromy Group of $p_3(z) = z^3 + \frac{3}{2}z$.

For a final example, we use the post-critical polynomial $p_3(z) = z^3 + \frac{3}{2}z$. For this polynomial we have $C = \{\frac{i}{\sqrt{2}}, \frac{-i}{\sqrt{2}}\}$ and $P = \{p_3(\frac{i}{\sqrt{2}}), p_3(\frac{-i}{\sqrt{2}})\}$. We may note having p_3 act on either of the critical points simply returns that point, that is, $p_3(\frac{i}{\sqrt{2}}) = \frac{i}{\sqrt{2}}$ and $p_3(\frac{-i}{\sqrt{2}}) = \frac{-i}{\sqrt{2}}$.

As described earlier, we chose the curves, $\gamma_{(\frac{i}{\sqrt{2}})} = \{z \in \mathbb{C} \mid -t + \frac{i}{\sqrt{2}} : t \geq 0\}$ and $\gamma_{(\frac{-i}{\sqrt{2}})} = \{z \in \mathbb{C} \mid t - \frac{i}{\sqrt{2}} : t \geq 0\}$ to construct the f -invariant spider $p_3(\gamma_{(\frac{i}{\sqrt{2}})})$ and $p_3(\gamma_{(\frac{-i}{\sqrt{2}})})$.

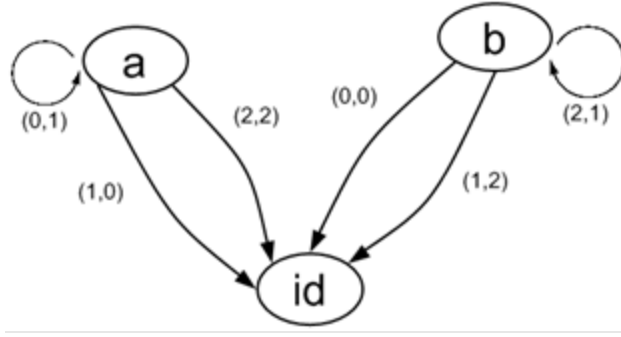


Again, we wish to create a critical portrait for the polynomial with the set of curves $X_z = \{p_3^{-1}(\gamma_{(P_3(z))}) \mid z \in C\}$. Therefore our critical portrait contains two curves $X_{(\frac{i}{\sqrt{2}})}$ and $X_{(\frac{-i}{\sqrt{2}})}$ and below is the critical portrait we constructed.



This critical portrait is different than those other polynomials because its critical points are the same as its post-critical points. This causes both post-critical points lie on the curves of the critical portrait.

State a corresponds to the post critical point $\frac{i}{\sqrt{2}}$ and state b corresponds to the post critical point $\frac{-i}{\sqrt{2}}$. The group actions are for a are $a \cdot 0 = 1 \cdot a, a \cdot 1 = 0 \cdot id, a \cdot 2 = 2 \cdot id$, and the actions for b are $b \cdot 0 = 0 \cdot id, b \cdot 1 = 2 \cdot b, b \cdot 2 = 1 \cdot id$. The following automaton represents the group of automorphisms from this polynomial:



Note $\alpha = (01)$ and $\beta = (12)$. Thus, writing a and b in bracket notation gives $a = \alpha(a, 1, 1)$ and $b = \beta(1, b, 1)$.

3.5. Isomorphism with the Infinite Dihedral Group.

Proposition 3.1. *The iterated monodromy groups of p, p_1, p_2 , and p_3 have polynomial growth.*

For the proof, we shall show the iterated monodromy group of $z^3 - 3z$ is isomorphic to D_∞ , which is a group known to have polynomial growth. The other groups proofs are similar.

Proof. First, note that both α and β have order two, as they are simple transpositions of two letters of our alphabet. Next, we show both a and b are also of order 2. Taking advantage of the bracket notation, we see $a^2 = \alpha(1, 1, b)\alpha(1, 1, b) = (1, 1, b^2)$.

However, we can also show this explicitly through acting on strings of the alphabet, considering $\alpha(1, 1, b)\alpha(1, 1, b)(x_1x_2x_3 \dots x_n)$ for some ternary string $x_1x_2 \dots x_n$. If x_1 is either 0 or 1, then, $(1, 1, b)$ simply acts as the identity, and since α acts on x_1 twice, it also does nothing, and, we simply find

$$\alpha(1, 1, b)\alpha(1, 1, b)(x_1x_2x_3 \dots x_n) = \alpha(1, 1, b)\alpha(1)(x_1x_2x_3 \dots x_n) = (x_1x_2x_3 \dots x_n).$$

On the other hand, if $x_1 = 2$, then $(1, 1, b)$ acts as the identity on x_1 and as b on $x_2x_3 \dots x_n$, but α acts as the identity on x_1 , as it does not permute 2. Thus,

$$\alpha(1, 1, b)\alpha(1, 1, b)(2x_2x_3 \dots x_n) = \alpha(1, 1, b)2b(x_2x_3 \dots x_n) = 2b^2(x_2x_3 \dots x_n).$$

Therefore, we again find $a^2 = (1, 1, b^2)$.

Using a similar process, we find $b^2 = \beta(a, 1, 1)\beta(a, 1, 1) = (a^2, 1, 1)$. Applying this inductively, we may note neither a^2 nor b^2 will ever contain an active state in any of their terms, and they will thus never act non-trivially on any string. Thus, they are both the identity, giving a and b each order 2.

Next, we will show the group element ab and its inverse ba are each of infinite order. To show this, we will assume the contrary, they are of finite order.

For this, note, $ab = \alpha(1, 1, b)\beta(a, 1, 1)$. Again taking advantage of bracket notation, this gives $ab = \alpha\beta(1, b, 1)(a, 1, 1) = \alpha\beta(a, b, 1)$. Similarly, $ba = \beta\alpha(1, a, b)$.

Note $\alpha\beta = (012)$ and $\beta\alpha = (210)$. Thus, both of these are of order 3, as they are cycles of three elements. Since $\alpha\beta$ is the only part of ab not fixing the top branch of the tree, we may then note the order of ab must also be a multiple of 3, else the top branch will have changed, and that multiple of ab is not the identity.

Then $|ab| = 3n$ for some $n \in \mathbb{N}$. Then $(ab)^{3n} = ((ab)^3)^n = 1$. Note, though, $(ab)^3 = \alpha\beta(a, b, 1)\alpha\beta(a, b, 1)\alpha\beta(a, b, 1) = (ba, ab, a)$. Thus, $(ba, ab, a)^n = 1 = (1, 1, 1)$. However, $(ba, ab, a)^n =$

$((ba)^n, (ab)^n, a^n)$, and so $(ab)^n = 1$. This contradicts the finding that $|ab| = 3n$, as then $(ab)^n = 1$, which requires $|ab| \leq n$. Thus, ab is of infinite order. A similar argument shows ba has infinite order.

With both generators having order two and their products having infinite order, we know that the iterated monodromy group of $p_1(z) = z^3 - 3z$ is isomorphic to the infinite dihedral group, which is known to have polynomial growth.

A similar argument shows the iterated monodromy groups of p, p_2 , and p_3 are also of polynomial growth. □

3.6. Bux-Pérez Method. Many of the techniques we use come from a paper by Bux and Pérez[2]. Most important of these is the use of weights to prove sub-exponential growth.

3.6.1. *Weights and Length Functions.*

For showing a group G has sub-exponential growth, we want a *length function*, a function $\ell : G \rightarrow \mathbb{R}_0^+$ giving each of the elements of G some non-negative weight, with $\ell(id) = 0$, $\ell(g) = \ell(g^{-1})$ for each $g \in G$, and $\ell(g_1) + \ell(g_2) \leq \ell(g_1g_2)$ for all $g_1, g_2 \in G$.

However, since we are defining a group G in terms of its generating set Σ , it is difficult to find such a length function. To find one, we will consider a map $\tilde{\ell} : \Sigma \rightarrow \mathbb{R}_0^+$ assigning a strictly positive weight to each non-identity generator. Then $\tilde{\ell}$ extends to a length function $\tilde{\ell}$ on the set Σ^* of all words in the alphabet $\Sigma \cup \Sigma^{-1}$, with $\tilde{\ell}(x_1^{\epsilon_1}x_2^{\epsilon_2} \cdots x_n^{\epsilon_n}) = \sum_{i=1}^n \tilde{\ell}(x_i)$ for any generic element $x_1^{\epsilon_1}x_2^{\epsilon_2} \cdots x_n^{\epsilon_n} \in \Sigma^*$, where $\epsilon_i \in \{0, 1\}$.

Then this induces a length function ℓ on G , where

$$\ell(g) = \min\{\ell(w) \mid w \text{ is a word representing } g\}$$

for any $g \in G$.

Groups with only a single active state will have only a single active generator τ . Among these, some have a reduced form in which the non- τ generators generate a finite set. We will write these groups in an alternating form, $\tau^{\epsilon_1}x_1\tau x_2 \cdots \tau x_n\tau^{\epsilon_2}$ where each x_i is some element of the set generated by the non- τ generators, and $\epsilon_0, \epsilon_1 \in \{0, 1\}$. Each minimum length word may be written in this form.

Definition 3.2. For a group in τ -alternating form, we say an x_i is *non-reducing* if $\ell(\tau) + \ell(x_i) = \ell(\phi_0) + \ell(\phi_1)$. We say an x_i is *reducing* if $\ell(\tau) + \ell(x_i) > \ell(\phi_0) + \ell(\phi_1)$. A length function is said to be *admissible* if every x_i is either reducing or non-reducing.

We say a word is ϵ -*bad* if it contains at most ϵn non-reducing x_i . A word is ϵ -*good* if it is not ϵ -bad.

The last key piece is this proposition proved by Bux and Pérez:

Proposition 3.3. (Bux-Pérez) [2]

Let H be a finite index subgroup of the finitely generate group G , and let ℓ be a length function on G as above.

Then suppose there exist numbers $\eta \in [0, 1)$, $p \in (0, 1]$, $K \geq 0$, and an injective homomorphism

$$\begin{aligned} \phi : H &\rightarrow \overbrace{G \times G \times \cdots \times G}^{n \text{ factors}} \\ \phi(h) &= (\phi_1(h), \dots, \phi_n(h)) \end{aligned}$$

such that for each $r \in \mathbb{N}$, the proportion of all elements in $\{h \in H \mid l(h) \leq r\}$ satisfying

$$\sum_{i=1}^n \ell(\phi_i(h)) \leq \eta r + K$$

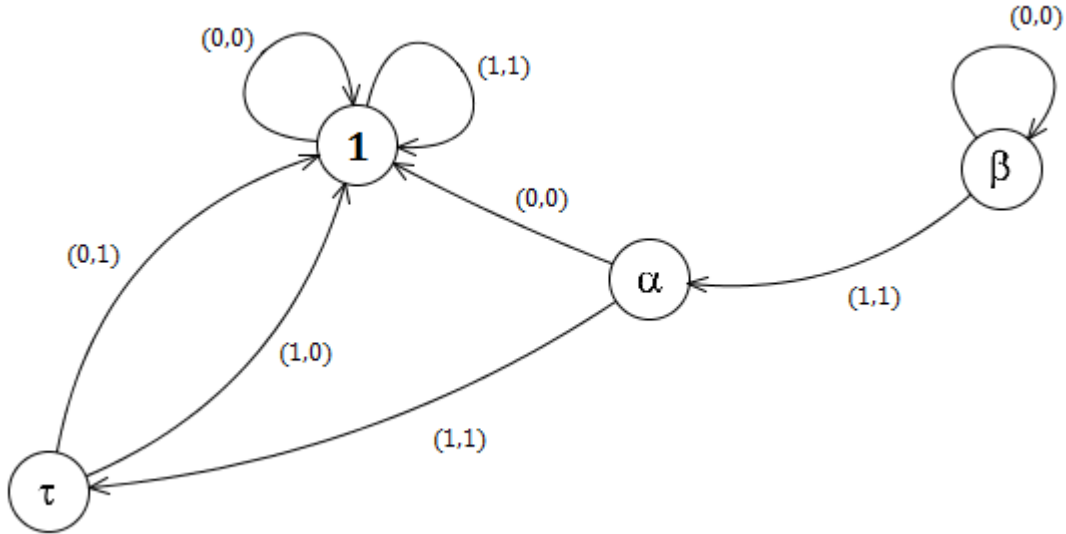
is at least p .

Then G has sub-exponential growth.

This theorem is convenient because it allows us to use properties of length reduction of words to show a group has sub-exponential growth without considering the actual elements in balls of increasing weights. Do note, however, that there is a significant difference between finding the proportion of words with the desired reduction and the proportion of elements of the group with the desired reduction.

4. INTERMEDIATE GROWTH FOR ITERATED MONODROMY GROUPS OF QUADRATIC POLYNOMIALS

4.1. Growth of \mathcal{A} . Consider the polynomial $z^2 + c$ where $c \approx -0.228155 + 1.11514i$ and is a root of $z^8 + 4z^7 + 6z^6 + 6z^5 + 4z^4$. Then $C = \{0\}$ and $P = \{-0.228155 + 1.11514i, -1.41964 + 0.606291i, 1.41964 - 0.606291i\}$. Using the same process as shown in the previous examples, the iterated monodromy group of this quadratic polynomial can be represented by the following automaton:



We then have that $\mathcal{A} = \langle \tau, \alpha, \beta \rangle$, where $\tau = (01)$, $\alpha = (1, \tau)$ and $\beta = (\beta, \alpha)$.

We first observe that since $\alpha^2 = 1$, that $\beta^2 = (\beta^2, 1)$. Thus each of our generators has order 2. It follows now that $\tau\alpha\tau\alpha = (\tau, \tau)$, and thus $(\tau\alpha)^4 = 1$. We can also see that since $\alpha\beta = (\beta, \tau\alpha)$,

we have that $(\alpha\beta)^4 = 1$. We also have that $\tau\beta\tau\beta = (\alpha\beta, \beta\alpha)$, and thus $(\tau\beta)^8 = 1$. Thus, any subgroup created by two of the generators in \mathcal{A} forms a quotient of one of the following dihedral groups:

$$\begin{aligned} D_4(\tau, \alpha) &:= \langle \tau, \alpha \mid \tau^2 = \alpha^2 = (\tau\alpha)^4 = 1 \rangle \\ D_8(\tau, \beta) &:= \langle \tau, \beta \mid \tau^2 = \beta^2 = (\tau\beta)^8 = 1 \rangle \\ D_4(\alpha, \beta) &:= \langle \alpha, \beta \mid \alpha^2 = \beta^2 = (\alpha\beta)^4 = 1 \rangle \end{aligned}$$

To represent any word w in \mathcal{A} , we can write $w = (\tau)x_1\tau x_2 \cdots \tau x_n(\tau)$, where $x_i \in D_4(\alpha, \beta)$. In order to write any word w as an alternating word of τ -letters and other generators, we can extend our generating set by making each of the elements in $D_4(\alpha, \beta)$ a new generator. If we do this, we should let the weight of each of these new generators just be the length according to the original generating set. This lets us not worry about having to find a minimal length representation of our word. We can then list out each of the unique elements that could go between τ -letters:

name	element	pair	name	element	pair
α_1	α	$(1, \tau)$	β_1	β	(β, α)
α_2	$\alpha\beta$	$(\beta, \tau\alpha)$	β_2	$\beta\alpha$	$(\beta, \alpha\tau)$
α_3	$\alpha\beta\alpha$	$(\beta, \tau\alpha\tau)$	β_3	$\beta\alpha\beta$	$(1, \alpha\tau\alpha)$
α_4	$\alpha\beta\alpha\beta$	$(1, \tau\alpha\tau\alpha)$			

With this extended generating set in hand, we can set up a new definition:

Definition 4.1. A word $(\tau)x_1\tau \cdots \tau x_n(\tau)$ is called a **reduced word** if each x_i is equal to a non- τ letter in the extended generating set. A generator is then said to **occur** or be **contained** in w if it appears as a subword of w . A word is also said to be **reducible** if it is not a reduced word.

Our goal now will be to apply Proposition 3.3 to prove subexponential growth. Our first step then should be to choose weights for our generators. It is important to recognize that we need to find an appropriate weight system that will minimize the amount of words that get no reduction when we split them (minimally, we should never increase the length of the word). To do this, we should try and set up some parameters for our weights so they minimize the amount of words that do not get any reduction when split into pair notation. We can do this by looking at some examples:

From $\tau\beta\tau\beta = (\alpha\beta, \beta\alpha)$, we need to have $\ell(\tau) \geq \ell(\alpha)$. This follows from the fact that the length of $\tau\beta\tau\beta$ is $2\ell(\tau) + 2\ell(\beta)$. The sum of the lengths of the words in the left and right coordinates of the pair notation is then $2\ell(\alpha) + 2\ell(\beta)$. We want to ensure that in the worst case scenario, we never increase the length of a word when we split it up. So as a minimal restriction, we need to have $2\ell(\tau) + 2\ell(\beta) \geq 2\ell(\alpha) + 2\ell(\beta)$. After cancellation, we then see that we at least need to satisfy $\ell(\tau) \geq \ell(\alpha)$.

In addition, $\tau\alpha\beta\alpha\tau\alpha\beta\alpha = (\tau\alpha\tau\beta, \beta\tau\alpha\tau)$, which forces us to have $\ell(\alpha) \geq \ell(\tau)$, and thus $\ell(\tau) = \ell(\alpha)$ from the first inequality. Lastly, if we look at $\tau\beta\alpha\beta\tau\beta\alpha\beta = (\alpha\tau\alpha, \alpha\tau\alpha)$, we have $2\ell(\beta) \geq \ell(\alpha)$. We now propose the weight system:

$$\ell(\tau) = 1, \ell(\alpha) = 1, \ell(\beta) = 1$$

It will not necessarily be the case that we can get all of our words to reduce in length when we split them. To prepare us for this case, we should try to at least have a global bound on the number of words of arbitrary length that get no reduction. Otherwise, it will be difficult to control the proportion of words that get less than a certain amount of reduction, a condition required to prove subexponential growth by Proposition 3.3. The following definitions will set our argument up so that we can find such a global bound.

Let w be a reduced word that contains an even number of τ -letters. When we split w into its pair notation, we will be looking for length reduction in two fundamental ways; we either create words in the pair notation that have a smaller length than w , or we create words in the pair notation that are reducible.

Definition 4.2. : Let w be a reduced word containing an even occurrence of τ -letters. We define the **first-line production of w** to be the product of the following algorithm:

- (1) Take a reduced word $w = (\tau)x_1\tau \cdots \tau x_n(\tau)$ that has an even occurrence of τ -letters and expand each x_i into its pair notation, denoted by (x_{i_L}, x_{i_R}) . If w represents an element in $D_4(\alpha, \beta)$, skip to (3), otherwise go to (2).
- (2) Take the leftmost string of the form $\tau(x_{i_L}, x_{i_R})$ inside w and use the property $\tau(x_{i_L}, x_{i_R}) = (x_{i_R}, x_{i_L})\tau$ to re-write $\tau(x_{i_L}, x_{i_R})$ as $(x_{i_R}, x_{i_L})\tau$. Since w has an even occurrence of τ -letters, every time we use this re-writing rule we get the cancellation of 2 τ -letters. This can be seen as follows for a string inside a word w :

$$\tau x_i \tau x_{i+1} = \tau(x_{i_L}, x_{i_R})\tau(x_{i+1_L}, x_{i+1_R}) = (x_{i_R}, x_{i_L})\tau\tau(x_{i+1_L}, x_{i+1_R}) = (x_{i_R}, x_{i_L})(x_{i+1_L}, x_{i+1_R}).$$

We can then see that we can perform (2) multiple times until there are no remaining τ -letters. Assuming we have done so, we can then proceed to (3).

- (3) Now that we have a word that can be expressed as a product of elements each of which is represented in the pair notation, we can multiply the elements in the left coordinate together and multiply elements in the right coordinate together. We should not perform any potential cancellation or reduction that could happen after we multiply elements together. We then call this final product the **first-line production of w** , which we will denote by $FLP(w)$.

It will be necessary to find a subgroup of finite index in \mathcal{A} to satisfy the conditions of Proposition 3.3. The simplest choice is to consider the stabilizer of the n th level of the tree, denoted by \mathcal{A}_n . A word representing an element in \mathcal{A}_n must contain an even number of τ -letters or else it acts directly on the first level of the tree (assuming $n \geq 1$). Thus it suffices to restrict our attention to reduced

words that contain an even occurrence of τ -letters as we did above.

Definition 4.3. Let $(\phi_L, \phi_R): \mathcal{A}_1 \rightarrow \mathcal{A} \times \mathcal{A}$ be the injective homomorphism that takes an element x containing an even number of τ -letters and writes it as an ordered pair, where $\phi_L(x)$ and $\phi_R(x)$ are the left and right coordinates of the pair notation respectively. An element $x \in D_4(\alpha, \beta)$ is **good by nature** if the following condition holds: $\ell(\phi_L(x)) + \ell(\phi_R(x)) < \ell(\tau) + \ell(x)$.

This definition allows us to identify the simplest cases of reduction we could have inside a word w , as just having the occurrence of a good by nature element guarantees some type of reduction after we split it. This follows from the fact that since a reduced word can be written as $(\tau)x_1\tau \cdots \tau x_n(\tau)$, except for a potential case at the beginning or end of a word, we can associate to each x_i a τ -letter. Then, since τ cannot be represented in pair notation, when we split the reduced word, we immediately lose the weight of one τ -letter for each x_i . The exception mentioned above is ultimately not problematic, as that case can only occur in at most one place in any given word.

We can now look at each of our non- τ generators then to see if they are good by nature:

x	$\ell(\tau) + \ell(x)$	$\ell(\phi_L(x)) + \ell(\phi_R(x))$	x	$\ell(\tau) + \ell(x)$	$\ell(\phi_L(x)) + \ell(\phi_R(x))$
α_1	2	1	β_1	2	2
α_2	3	3	β_2	3	3
α_3	4	4	β_3	4	3
α_4	5	4			

Our good by nature elements are thus: α_1 , α_4 , and β_3 . We should now try and distinguish between elements that lead to reduction and those that do not.

Definition 4.4. A **primitive set \mathcal{S} of good blocks** is a finite set of reduced words (the good blocks) that have an even occurrence of τ -letters such that each $w \in \mathcal{S}$ satisfies one of the following:

- (i) For $FLP(w) = (w_L, w_R)$, at least one of w_L or w_R is reducible.
- (ii) A good by nature element occurs between two τ -letters in one of the reduced coordinates of $FLP(w)$, and that coordinate is also in \mathcal{A}_1 .
- (iii) For $FLP(w) = (w_L, w_R)$, either $FLP(w_L)$ or $FLP(w_R)$ has a reduced coordinate in \mathcal{A}_1 that has an occurrence of a good by nature element in between two τ letters.
- (iv) The element represented by w is good by nature.
- (v) For $FLP(w_L) = (w_{LL}, w_{RR})$ and $FLP(w_R) = (w_{RL}, w_{RR})$, either $FLP(w_{LL})$, $FLP(w_{LR})$, $FLP(w_{RL})$, or $FLP(w_{RR})$ has a reduced coordinate that is in \mathcal{A}_1 and has an occurrence of a good by nature element in between two τ letters.

Definition 4.5. An \mathcal{S} -bad block is a reduced word $(\tau)x_1\tau \cdots \tau x_n(\tau)$ containing no occurrence of any $w \in \mathcal{S}$.

Our goal will now be the following:

(*) Find an appropriate primitive set \mathcal{S} of good blocks such that the number of \mathcal{S} -bad blocks containing i non- τ letters is globally bounded by some M , for any i .

Proposition 4.6. *There is a primitive set \mathcal{S} of good blocks for the subgroup \mathcal{A}_3 that satisfies (*).*

Proof. We first propose the following primitive set \mathcal{S} of good blocks:

$$\mathcal{S} = \left(\bigcup_{i=1}^8 P_i \right) \cup \{\alpha_1\} \cup \{\alpha_4\} \cup \{\beta_3\} \cup \{\alpha_3\}, \text{ where each } P_i \text{ is defined below.}$$

It is easy to see that α_1 , α_4 , and β_3 satisfy condition (iv) of Definition (4.4) to be in a primitive set of good blocks. Since $FLP(\alpha_3) = (\beta, \tau\alpha\tau)$, it follows that $\tau\alpha\tau$ contains an occurrence of a good by nature element isolated between two τ -letters, so α_3 satisfies condition (ii) of Definition (4.4). We will now define each P_i and verify that each set consists of blocks that are in \mathcal{S} .

First, observe that if we let \square be an arbitrary element in the set $\{\beta_1, \beta_2, \alpha_2\}$ and $FLP(\square) = (\square_L, \square_R)$, then $\square_L = \beta$. It follows then that for $w = \tau\square\tau$, we have $w_R = \beta$.

From here define the following sets that contain blocks in \mathcal{S} .

$\underline{P_1}$: For a block $b \in P_1$, b has the form $\tau\square\tau\beta_1\tau\square\tau\beta_1\tau\square\tau$ and creates the string $\beta\alpha\beta\alpha\beta$ in b_R . Thus, every word in P_1 meets condition (i) of Definition (4.4) as $\beta\alpha\beta\alpha\beta$ can be reduced to $\alpha\beta\alpha$.

$\underline{P_2}$: For a block $b \in P_2$, b has the form $\square^*\tau\square\tau\beta_1\tau\square\tau\beta_2$ and creates the string $\square_R^*\beta\alpha\beta\alpha\tau$ in b_R . Since \square_R^* can only contain α or τ -letters, it follows that if \square_R^* ends in τ , we meet condition (ii). If \square_R^* ends in α , then we meet condition (i). Thus, every word in P_2 meets either condition (i) or (ii).

$\underline{P_3}$: For a block $b \in P_3$, b has the form $\beta_2\tau\square\tau\beta_1\tau\square\tau\alpha_2$ or $\alpha_2\tau\square\tau\beta_1\tau\square\tau\alpha_2$ and creates $\alpha\tau\beta_3\tau\alpha$ or $\tau\alpha_4\tau\alpha$ respectively in b_R . Thus, in either case, every word in P_3 meets condition (ii).

$\underline{P_4}$: For a block $b \in P_4$, b has the form $\tau\square\tau\beta_1\tau\square\tau\beta_1\tau\square\tau\alpha_2$ and creates $\beta\alpha\beta\alpha\beta\tau\alpha$ in b_R . Thus, every word in P_4 meets condition (i).

$\underline{P_5}$: For a block $b \in P_5$, b has the form $\beta_2\tau\square\tau\beta_1\tau\square\tau\beta_2$ and creates $\alpha\tau\alpha_4\tau$ in b_R . Thus, in either case, every word in P_5 meets condition (ii).

$\underline{P_6}$: For a block $b \in P_6$, b has the form $\tau\square\tau\beta_2\tau\square\tau\beta_2\tau\square\tau\alpha_2\tau\square\tau\alpha_2$ and creates $\beta\alpha\tau\beta_2\tau\beta_1\tau\alpha_2\tau\alpha$ in b_R . It follows then that $FLP(b_R)$ creates $\alpha\tau\beta_3\tau$ in b_{RR} . Thus, every word in P_6 meets condition (iii).

$\underline{P_7}$: For a block $b \in P_7$, b has the form $\alpha_2\tau\square\tau\beta_1\tau\square\tau\square^*$ and creates $\tau\alpha\beta\alpha\beta\square_R^*$ in b_R . Since \square_R^* can only contain α or τ -letters, it follows that if \square_R^* begins with α , then we satisfy condition (i). If \square_R^* begins with τ , then we satisfy condition (ii). Thus, every word in P_7 satisfies condition (i) or (ii).

$\underline{P_8}$: For a block $b \in P_8$, b has the form $\alpha_2\tau\square\tau\beta_2$ and creates $\tau\alpha_3\tau$ in b_R . It follows then that $\overline{b_{RL}}$ contains $\tau\alpha_1\tau$. Thus, every word in P_8 meets condition (iii).

Now that we have our primitive set \mathcal{S} , we can prove the proposition. Since a reduced word w is of the form $\cdots\tau x_{i-1}\tau x_i\tau x_{i+1}\cdots$, if w is an \mathcal{S} -bad block, then it cannot contain an occurrence of $\alpha_1, \alpha_3, \alpha_4$, or β_3 . Thus, every \mathcal{S} -bad block is of the form $\cdots\tau\square\tau\square\tau\cdots$

We can then show that the number of ways to arrange the elements from $\{\alpha_2, \beta_1, \beta_2\}$ in an \mathcal{S} -bad block is finite, and hence we have a global bound.

Consider an arbitrary \mathcal{S} -bad word of the form: $\cdots\tau\square\tau\square\tau\square_1\tau\square_2\tau\square_3\tau\square_4\tau\cdots$. It is sufficient to only show that there are a finite number of choices for $\square_1, \square_3, \square_5, \dots$, since we must have the exact same restriction for $\square_2, \square_4, \square_6, \dots$, and thus there are only a finite number of choices for the whole word to remain an \mathcal{S} -bad block. Before we do a case analysis though, we should remark that we are allowed to assume our \mathcal{S} -bad block has sufficiently many non- τ letters in its reduced word to give the patterns described below. Otherwise we can say we have a finite bound since we would have only a finite number of cases that do not meet this condition.

(1) $\square_1 = \alpha_2$

Given this, we cannot have $\square_3 = \beta_2$ or else we create a block in P_8 . If $\square_3 = \beta_1$, then regardless of our choice for \square_5 , we will create a block from P_7 . Thus the only way a string $\alpha_2\tau\square\tau\beta_1$ can occur is at the end of a word in order to remain an \mathcal{S} -bad block. Essentially then, except for a small amount of variation at the beginning and end of the word, if $\square_i = \alpha_2$, then $\square_{i+2}, \square_{i+4}, \dots$ must all be α_2 .

(2) $\square_1 = \beta_1$

Given this, if $\square_3 = \beta_2$, then assuming there is at least $\square\tau\square\tau$ to the left of \square_1 , this creates a block from P_2 . If $\square_3 = \beta_1$, assuming there is at least $\tau\square\tau$ to the left of \square_1 and $\tau\square\tau$ to the right of \square_3 , this creates a block from P_1 . If $\square_3 = \alpha_2$, then assuming there is at least $\tau\square\tau\square\tau\square\tau$ to the left of \square_1 , we create a word from either P_3 or P_4 . Thus, the only place β_1 can appear is near the beginning or end of a word, otherwise we make a good block from \mathcal{S} .

(3) $\square_1 = \beta_2$

Given this, say $\square_3 = \beta_1$. If $\square_5 = \beta_2$, we create a block in P_5 . If $\square_5 = \alpha_2$, then we create a block in P_3 . If $\square_5 = \beta_1$, then assuming we have at least $\tau\square\tau$ to the right of \square_5 , we have made a block in P_1 . Thus, $\beta_2\tau\square\tau\beta_1$ can only appear near the beginning or end of a word. Say now that $\square_3 = \alpha_2$, then as a result from case (1), $\square_5, \square_7, \dots$ must all be α_2 , except for some variation at the end of the word.

We discovered in case (1) that if α_2 occurs in an \mathcal{S} -bad block at position i in a word w then every letter at position congruent to i modulo 4 to the right of i must also be α_2 . If β_2 occurs in a \mathcal{S} -bad block at position i then we have 2 subcases to consider.

The first subcase is we have a string of β_2 -letters at positions i modulo 4 up until position j , where α_2 is at position j . Thus at every subsequent position j modulo 4 we have α_2 . Every sufficiently long string of this form contains as a substring a block in P_6 . Hence this subcase cannot occur in an \mathcal{S} -bad block. The other subcase to consider is that if we have β_2 at position i , then at every subsequent position i modulo 4, we must also have β_2 . This subcase then never creates a substring in P_i , for arbitrary i .

Thus if we fix either α_2 or β_2 at position i in a word, for sufficiently long words we can guarantee that every position i modulo 4 to the right of i must be the same letter that occurs at position i except for some variation at the end of the word.

It follows then that there is a global bound M on the number of \mathcal{S} -bad blocks that contain i non- τ letters since after we pick the first few non- τ letters at the end and beginning of a word, the rest of the word is essentially a proper power of one of the following words: $\beta_2\tau$, $\beta_2\tau\alpha_2\tau$, or $\alpha_2\tau$. \square

Let \mathcal{A}_3 be the subgroup in \mathcal{A} that stabilizes the first three levels of the tree. Consider now the injective homomorphism $\phi: \mathcal{A}_3 \rightarrow \mathcal{A}^8$ that maps $g \in \mathcal{A}_3$ to the ordered 8-tuple $(\phi_{LLL}(g), \phi_{LLR}(g), \dots, \phi_{RRR}(g))$.

We will now want to count the number of alternating words that represent elements in \mathcal{A}_3 that contain a certain proportion of good letters. First, fix some small $\epsilon > 0$.

Definition 4.7. An alternating word w containing n non- τ letters is called ϵ -**bad** if at most ϵn of its non- τ letters occur in good blocks in \mathcal{S} .

Definition 4.8. A non- τ letter is called **good by position** if it belongs to a block in \mathcal{S} .

We now propose the following bound on the number of ϵ -bad elements in \mathcal{A}_3 containing n non- τ letters, which we will denote by $b_\epsilon(n)$:

$$b_\epsilon(n) \leq 2 \binom{n}{\lfloor \epsilon n \rfloor} 7^{\lfloor \epsilon n \rfloor} M^{1+\lfloor \epsilon n \rfloor}$$

- The 2 accounts for the choice of starting with a τ -letter or not.
- The $\binom{n}{\lfloor \epsilon n \rfloor}$ accounts for the number of ways to designate positions for good by position letters.
- The $7^{\lfloor \epsilon n \rfloor}$ accounts for the maximum number of ways to place good by position letters.
- The $M^{1+\lfloor \epsilon n \rfloor}$ accounts for the maximum number of choices to place \mathcal{S} -bad blocks. The $1 + \lfloor \epsilon n \rfloor$ is explained by the fact that once we place our good by position letters, we break up the word into at most $1 + \lfloor \epsilon n \rfloor$ compartments to place \mathcal{S} -bad blocks.

Since all elements in the original generating set have equal length, a word of length r containing the most non- τ letters would begin and end with a non- τ letter. We thus have that any word of length r contains at most $(r+1)/2$ non- τ letters. Denote by $B_\epsilon(r)$ the number of words in $\{g \in \mathcal{A}_3 \mid \ell(g) \leq r\}$ that contain at most ϵn non- τ letters. We then have the following inequality:

$$B_\epsilon(r) \leq \sum_{n=0}^{(r+1)/2} b_\epsilon(n) \leq (r+1) \binom{(r+1)/2}{\lfloor \epsilon(r+1)/2 \rfloor} 7^{\lfloor \epsilon(r+1)/2 \rfloor} M^{1+\lfloor \epsilon(r+1)/2 \rfloor}$$

Proposition 4.9. *\mathcal{A} has subexponential growth.*

Proof. Assume now that \mathcal{A} has exponential growth. Then \mathcal{A}_3 has exponential growth as well [2]. Let $\lambda > 1$ be the growth rate of \mathcal{A}_3 . By Stirling's formula, choose $\epsilon > 0$ so that λ^r grows faster than the number of elements in $B_\epsilon(r)$. We then observe then that for any $g \in \mathcal{A}_3$ represented by a word w that contains a subword $v \in \mathcal{S}$, after splitting w at most two times, we either create a reducible word or create a good by nature element between two τ letters. Thus, after splitting w at most three times we are guaranteed to have reduction by at least some fixed amount. This is true since we either get immediate reduction in the former case, or get reduction after splitting again in the latter case. Since \mathcal{S} is a finite set, there is a word that gets the worst reduction such that we can guarantee any other word in \mathcal{S} reduces by at least that much. Thus there is a number η depending on ϵ that satisfies Proposition 3.3. This in turn implies that \mathcal{A}_3 has subexponential growth. \square

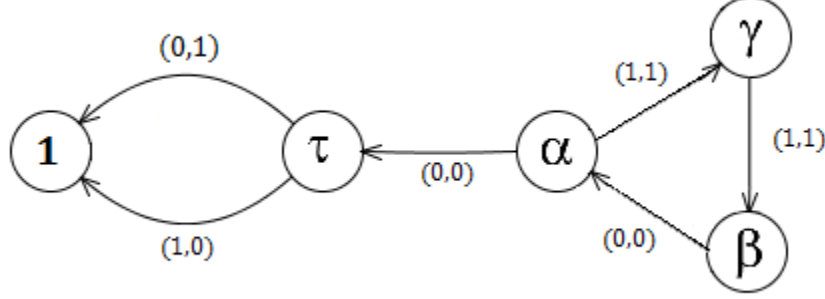
Proposition 4.10. *\mathcal{A} has superpolynomial growth.*

Proof. Consider $\mathcal{A}_1 \trianglelefteq \mathcal{A}$ and let $\phi: \mathcal{A}_1 \rightarrow \mathcal{A} \times \mathcal{A}$ be the injective homomorphism used before. It follows then that $\mathcal{A}_1 \cong \phi(\mathcal{A}_1)$. Let N_τ be the normal closure of τ . It follows that the factor group \mathcal{A}/N_τ has order at most $|\langle \alpha, \beta \rangle| = 8$ since the coset $\tau N_\tau = N_\tau$ in \mathcal{A}/N_τ . Thus $[\mathcal{A}: N_\tau] \leq 8$.

Since α and $\tau\alpha\tau$ are in \mathcal{A}_1 , it follows that $\langle \phi(\alpha), \phi(\tau\alpha\tau) \rangle = \langle (1, \tau), (\tau, 1) \rangle \leq \phi(\mathcal{A}_1)$. It is easy to see that $N_\tau \times 1 \subset \phi(\mathcal{A}_1)$ since given any $x \in \mathcal{A}_1$ such that $\phi(x) = (x_0, x_1)$, we have that $\phi(x\tau\alpha\tau x^{-1}) = (x_0\tau x_0^{-1}, 1)$. Thus, by definition, the left coordinate of $\phi(x)$ is in N_τ . Similarly, $1 \times N_\tau \subset \phi(\mathcal{A}_1)$, and thus $N_\tau \times N_\tau \subset \phi(\mathcal{A}_1)$.

Then $[\mathcal{A} \times \mathcal{A} : \phi(\mathcal{A}_1)] \leq [\mathcal{A} \times \mathcal{A} : N_\tau \times N_\tau] = [\mathcal{A} : N_\tau]^2 \leq 8^2 = 64$. This conclusion proves that \mathcal{A} and $\mathcal{A} \times \mathcal{A}$ are commensurable. Using a known theorem [3], if a finitely generated infinite group G is commensurable with G^m then G cannot have polynomial growth. Thus, \mathcal{A} has super-polynomial growth. \square

4.2. **Growth of \mathcal{B} .** Consider another quadratic polynomial $z^2 + c$ where c is a root of $z^{16} + 8z^{15} + 28z^{14} + 60z^{13} + 94z^{12} + 116z^{11} + 114z^{10} + 94z^9 + 69z^8 + 44z^7 + 26z^6 + 14z^5 + 5z^4 + 2z^3$ such that the iterated monodromy group of this quadratic polynomial can be represented by the following automaton:



We then have $\mathcal{B} = \langle \tau, \alpha, \beta, \gamma \rangle$, where $\tau = (01)$, $\alpha = (\tau, \gamma)$, $\beta = (\alpha, 1)$, and $\gamma = (1, \beta)$.

We first observe that $\alpha^2 = (\tau^2, \gamma^2) = (1, \gamma^2)$. Thus, since $\beta^2 = (\alpha^2, 1)$ and $\gamma^2 = (1, \beta^2)$, we have that $\alpha^2 = \beta^2 = \gamma^2 = 1$. We first observe that $\tau\gamma\tau\gamma = (\beta, \beta)$, and thus $(\tau\gamma)^4 = 1$. From here we can see that since $\tau\alpha\tau\alpha = (\gamma\tau, \tau\gamma)$, we must have $(\tau\alpha)^8 = 1$. It now follows that $\alpha\beta = (\tau\alpha, \gamma)$ must have order 8 as well. We can also see that $\beta\gamma = (\alpha, \beta)$ has order 2, and so $\alpha\gamma = (\tau, \gamma\beta)$ must have order 2 as well. Lastly, we can see that $\tau\beta\tau\beta = (\alpha, \alpha)$, and thus $(\tau\beta)^4 = 1$. We thus have that any two of the generators in \mathcal{B} forms a quotient of one of the following dihedral groups:

$$\begin{aligned} D_8(\tau, \alpha) &:= \langle \tau, \alpha \mid \tau^2 = \alpha^2 = (\tau\alpha)^8 = 1 \rangle \\ D_4(\tau, \beta) &:= \langle \tau, \beta \mid \tau^2 = \beta^2 = (\tau\beta)^4 = 1 \rangle \\ D_4(\tau, \gamma) &:= \langle \tau, \gamma \mid \tau^2 = \gamma^2 = (\tau\gamma)^4 = 1 \rangle \\ D_8(\alpha, \beta) &:= \langle \alpha, \beta \mid \alpha^2 = \beta^2 = (\alpha\beta)^8 = 1 \rangle \\ D_2(\alpha, \gamma) &:= \langle \alpha, \gamma \mid \alpha^2 = \gamma^2 = (\alpha\gamma)^2 = 1 \rangle \\ D_2(\beta, \gamma) &:= \langle \beta, \gamma \mid \beta^2 = \gamma^2 = (\beta\gamma)^2 = 1 \rangle \end{aligned}$$

Since we want to represent any word w in \mathcal{B} as $w = (\tau)x_1\tau x_2 \cdots \tau x_n(\tau)$, where $x_i \in \langle \alpha, \beta, \gamma \rangle$, we should first verify that $\langle \alpha, \beta, \gamma \rangle$ has finite order.

Using the homomorphism ϕ from the previous section, we can see that given any $x \in \langle \alpha, \beta, \gamma \rangle$, we have that $\phi_L(x) \subset D_8(\tau, \alpha)$ which has order 16 and $\phi_R(x) \subset D_2(\beta, \gamma)$ which has order 4. It follows then that $|\langle \alpha, \beta, \gamma \rangle| \leq 64$. Consider now the coset space $\langle \alpha, \beta, \gamma \rangle / D_8(\alpha, \beta)$. We first realize that since $\alpha\gamma = \gamma\alpha$ and $\beta\gamma = \gamma\beta$, any $x \in \langle \alpha, \beta, \gamma \rangle$ contains at most one γ or else we can move γ -letters through x until we get cancellation. Observing that γ commutes with α and β , we can then write any $x \in \langle \alpha, \beta, \gamma \rangle$ that contains one γ -letter as $x = \gamma y$, where $y \in D_8(\alpha, \beta)$. If we then consider the coset $x D_8(\alpha, \beta)$, we have $x D_8(\alpha, \beta) = \gamma y D_8(\alpha, \beta) = \gamma D_8(\alpha, \beta)$. It follows that $[\langle \alpha, \beta, \gamma \rangle : D_8(\alpha, \beta)] = 2$, and hence $D_8(\alpha, \beta) \trianglelefteq \langle \alpha, \beta, \gamma \rangle$. We then have that $|\langle \alpha, \beta, \gamma \rangle| = 2 \cdot |D_8(\alpha, \beta)| = 2 \cdot 16 = 32$.

We will again want to write any word w as an alternating word of τ -letters and other generators, so we can extend our generating set by making each of the elements in $\langle \alpha, \beta, \gamma \rangle$ a new generator:

name	element	pair	name	element	pair
s_1	γ	$(1, \beta)$	s_{17}	$\beta\alpha\beta\alpha\beta\alpha\beta$	$(\alpha\tau\alpha\tau\alpha, \gamma)$
s_2	β	$(\alpha, 1)$	s_{18}	$\gamma\alpha$	$(\tau, \beta\gamma)$
s_3	$\gamma\beta$	(α, β)	s_{19}	$\gamma\alpha\beta$	$(\tau\alpha, \beta\gamma)$
s_4	α	(τ, γ)	s_{20}	$\gamma\alpha\beta\alpha$	$(\tau\alpha\tau, \beta)$
s_5	$\alpha\beta$	$(\tau\alpha, \gamma)$	s_{21}	$\gamma\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha, \beta)$
s_6	$\alpha\beta\alpha$	$(\tau\alpha\tau, 1)$	s_{22}	$\gamma\alpha\beta\alpha\beta\alpha$	$(\tau\alpha\tau\alpha\tau, \beta\gamma)$
s_7	$\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha, 1)$	s_{23}	$\gamma\alpha\beta\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha\tau\alpha, \beta\gamma)$
s_8	$\alpha\beta\alpha\beta\alpha$	$(\tau\alpha\tau\alpha\tau, \gamma)$	s_{24}	$\gamma\alpha\beta\alpha\beta\alpha\beta\alpha$	$(\tau\alpha\tau\alpha\tau\alpha\tau, \beta)$
s_9	$\alpha\beta\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha\tau\alpha, \gamma)$	s_{25}	$\gamma\alpha\beta\alpha\beta\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha\tau\alpha\tau\alpha, \beta)$
s_{10}	$\alpha\beta\alpha\beta\alpha\beta\alpha$	$(\tau\alpha\tau\alpha\tau\alpha\tau, 1)$	s_{26}	$\gamma\beta\alpha$	$(\alpha\tau, \beta\gamma)$
s_{11}	$\alpha\beta\alpha\beta\alpha\beta\alpha\beta$	$(\tau\alpha\tau\alpha\tau\alpha\tau\alpha, 1)$	s_{27}	$\gamma\beta\alpha\beta$	$(\alpha\tau\alpha, \beta\gamma)$
s_{12}	$\beta\alpha$	$(\alpha\tau, \gamma)$	s_{28}	$\gamma\beta\alpha\beta\alpha$	$(\alpha\tau\alpha\tau, \beta)$
s_{13}	$\beta\alpha\beta$	$(\alpha\tau\alpha, \gamma)$	s_{29}	$\gamma\beta\alpha\beta\alpha\beta$	$(\alpha\tau\alpha\tau\alpha, \beta)$
s_{14}	$\beta\alpha\beta\alpha$	$(\alpha\tau\alpha\tau, 1)$	s_{30}	$\gamma\beta\alpha\beta\alpha\beta\alpha$	$(\alpha\tau\alpha\tau\alpha\tau, \beta\gamma)$
s_{15}	$\beta\alpha\beta\alpha\beta$	$(\alpha\tau\alpha\tau\alpha, 1)$	s_{31}	$\gamma\beta\alpha\beta\alpha\beta\alpha\beta$	$(\alpha\tau\alpha\tau\alpha\tau\alpha, \beta\gamma)$
s_{16}	$\beta\alpha\beta\alpha\beta\alpha$	$(\alpha\tau\alpha\tau\alpha\tau, \gamma)$			

We should again determine what types of restrictions we can have on our lengths:

$$\begin{aligned} \tau\alpha\tau\alpha &= (\gamma\tau, \tau\gamma) \Rightarrow \ell(\alpha) \geq \ell(\gamma). \\ \tau\alpha\beta\tau\alpha\beta &= (\gamma\tau\alpha, \tau\alpha\gamma) \Rightarrow \ell(\beta) \geq \ell(\gamma). \\ \tau\gamma\alpha\tau\gamma\alpha &= (\beta\gamma\tau, \tau\beta\gamma) \Rightarrow \ell(\alpha) \geq \ell(\beta). \\ \tau\gamma\beta\alpha\beta\tau\gamma\beta\alpha\beta &= (\gamma\beta\alpha\tau\alpha, \alpha\tau\alpha\tau\gamma\beta) \Rightarrow \ell(\beta) \geq \ell(\alpha). \end{aligned}$$

From the last two inequalities that we must have $\ell(\alpha) = \ell(\beta)$. We now propose the weight system:

$$\ell(\tau) = 3, \ell(\gamma) = 6, \ell(\alpha) = 7, \ell(\beta) = 7$$

We can look at each of our non- τ letters then to see which elements are good by nature:

x	$\ell(\tau) + \ell(x)$	$\ell(\phi_L(x)) + \ell(\phi_R(x))$	x	$\ell(\tau) + \ell(x)$	$\ell(\phi_L(x)) + \ell(\phi_R(x))$
s_1	9	7	s_{17}	52	43
s_2	10	7	s_{18}	16	16
s_3	16	14	s_{19}	23	23
s_4	10	9	s_{20}	30	20
s_5	17	16	s_{21}	37	27
s_6	24	13	s_{22}	44	36
s_7	31	20	s_{23}	51	43
s_8	38	29	s_{24}	58	40
s_9	45	36	s_{25}	65	47
s_{10}	52	33	s_{26}	23	23
s_{11}	59	40	s_{27}	30	30
s_{12}	17	16	s_{28}	37	27
s_{13}	24	23	s_{29}	44	34
s_{14}	31	20	s_{30}	51	43
s_{15}	38	27	s_{31}	58	50
s_{16}	45	36			

Denote by \mathcal{G} the set of good by nature elements. It follows then that the only non- τ generators not in \mathcal{G} are s_{18} , s_{19} , s_{26} , and s_{27} . Let \square denote any of the elements in $\{s_{18}, s_{19}, s_{26}, s_{27}\}$, and we notice that for $FLP(\square) = (\square_L, \square_R)$, we have that $\square_R = \beta\gamma$.

Proposition 4.11. *There is a primitive set \mathcal{S} of good blocks for the subgroup \mathcal{B}_4 that satisfies (*).*

Proof. We first propose the primitive set $\mathcal{S} = \left(\bigcup_{i=1}^{12} P_i \right) \cup \mathcal{G}$, where each P_i is defined below.

It is easily seen that every element of \mathcal{G} satisfies condition (iv) of Definition (4.4) to be in a primitive set of good blocks.

P_1 : For a block $b \in P_1$, b has the form $s_{18}\tau\square\tau s_{18}$ and creates the string $\tau s_3\tau$ in b_L . Thus, every word in P_1 meets condition (ii) of Definition (4.4).

P_2 : For a block $b \in P_2$, b has the form $s_{18}\tau\square\tau s_{19}$ and creates the string $\tau s_3\tau\alpha$ in b_L . Thus, every word in P_2 meets condition (ii).

P_3 : For a block $b \in P_3$, b has the form $\tau\square\tau s_{18}\tau\square\tau s_{26}\tau\square\tau$ and creates the string $\beta\gamma\tau s_{26}\tau\beta\gamma$ in b_L . It follows then that $FLP(b_L)$ creates the string $\alpha\beta\gamma\alpha$ in b_{LL} , and $FLP(b_{LL})$ creates the string $\tau s_4\tau$ in b_{LLL} . Thus, every word in P_3 meets condition (v).

P_4 : For a block $b \in P_4$, b has the form $\tau\square\tau s_{19}\tau\square\tau s_{18}\tau\square\tau$ and creates the string $\beta\gamma\tau s_{19}\tau\beta\gamma$ in b_L . It follows then that $FLP(b_L)$ creates the string $\alpha\beta\gamma\alpha$ in b_{LL} , and $FLP(b_{LL})$ creates the string $\tau s_4\tau$ in b_{LLL} . Thus, every word in P_4 meets condition (v).

P_5 : For a block $b \in P_5$, b has the form $s_{19}\tau\square\tau s_{26}$ and creates the string $\tau s_{20}\tau$ in b_L . Thus, every word in P_5 meets condition (ii).

P_6 : For a block $b \in P_6$, b has the form $s_{19}\tau\square\tau s_{27}$ and creates the string $\tau s_{20}\tau\alpha$ in b_L . Thus, every word in P_6 meets condition (ii).

P_7 : For a block $b \in P_7$, b has the form $s_{26}\tau\square\tau s_{18}$ and creates the string $\alpha\tau s_3\tau$ in b_L . Thus, every word in P_7 meets condition (ii).

P_8 : For a block $b \in P_8$, b has the form $s_{26}\tau\square\tau s_{19}$ and creates the string $\alpha\tau s_3\tau\alpha$ in b_L . Thus, every word in P_8 meets condition (ii).

P_9 : For a block $b \in P_9$, b has the form $s_{26}\tau\square\tau s_{27}$ and creates the string $\alpha\tau s_{26}\tau\alpha$ in b_L . It follows then that $FLP(b_L)$ creates $\tau s_3\tau$ in b_{LL} . Thus, every word in P_9 meets condition (iii).

P_{10} : For a block $b \in P_{10}$, b has the form $s_{27}\tau\square\tau s_{19}$ and creates the string $\alpha\tau s_{19}\tau\alpha$ in b_L . It follows then that $FLP(b_L)$ creates $\tau s_3\tau$ in b_{LL} . Thus, every word in P_{10} meets condition (iii).

P_{11} : For a block $b \in P_{11}$, b has the form $s_{27}\tau\square\tau s_{26}$ and creates the string $\alpha\tau s_{20}\tau$ in b_L . Thus, every word in P_{11} meets condition (ii).

$\underline{P_{12}}$: For a block $b \in P_{12}$, b has the form $s_{27}\tau\Box\tau s_{27}$ and creates the string $\alpha\tau s_{20}\tau\alpha$ in b_L . Thus, every word in P_{12} meets condition (ii).

Now that we have our primitive set \mathcal{S} , we can prove the proposition. Since a reduced word w is of the form $\cdots\tau x_{i-1}\tau x_i\tau x_{i+1}\cdots$, if w is an \mathcal{S} -bad block, then it cannot contain an occurrence of any element in \mathcal{G} . Thus, every \mathcal{S} -bad block is of the form $\cdots\tau\Box\tau\Box\tau\cdots$, where $\Box \in \{s_{18}, s_{19}, s_{26}, s_{27}\}$. Once again if we can then show that the number of ways to arrange the elements from $\{s_{18}, s_{19}, s_{26}, s_{27}\}$ in an \mathcal{S} -bad block is finite, we will have the global bound.

Consider an arbitrary \mathcal{S} -bad word of the form: $\cdots\tau\Box\tau\Box\tau\Box_1\tau\Box_2\tau\Box_3\tau\Box_4\tau\cdots$. It is sufficient to only show that there are a finite number of choices for $\Box_1, \Box_3, \Box_5, \dots$, since we must have the exact same restriction for $\Box_2, \Box_4, \Box_6, \dots$, and thus there are only a finite number of choices for the whole word to remain an \mathcal{S} -bad block. Before we do a case analysis though, we should once again assume our \mathcal{S} -bad block has sufficiently many non- τ letters to give the patterns described below.

(1) : $\Box_1 = s_{18}$.
If $\Box_3 = s_{18}$ or s_{19} , then we create either a block in either P_1 or P_2 . If $\Box_3 = s_{26}$, then assuming we have a $\tau\Box\tau$ before \Box_1 and after \Box_3 , we create a block in P_3 . Thus, if $\Box_1 = s_{18}$, for sufficiently long words the only option to remain an \mathcal{S} -bad block is to have $\Box_3 = s_{27}$.

(2) : $\Box_1 = s_{19}$.
If $\Box_3 = s_{18}$, then assuming we have a $\tau\Box\tau$ before \Box_1 and after \Box_3 , we create a block in P_4 . If $\Box_3 = s_{26}$ or s_{27} , we create a block in either P_5 or P_6 . Thus, if $\Box_1 = s_{19}$, for sufficiently long words the only option to remain an \mathcal{S} -bad block is to have $\Box_3 = s_{19}$.

(3) : $\Box_1 = s_{26}$.
If $\Box_3 = s_{18}, s_{19}$, or s_{27} then we create a block in either P_7, P_8 , or P_9 . Thus, if $\Box_1 = s_{26}$, for sufficiently long words the only option to remain an \mathcal{S} -bad block is to have $\Box_3 = s_{26}$.

(4) : $\Box_1 = s_{27}$.
If $\Box_3 = s_{19}, s_{26}$, or s_{27} then we create a block in either P_{10}, P_{11} , or P_{12} . Thus, if $\Box_1 = s_{27}$, for sufficiently long words the only option to remain an \mathcal{S} -bad block is to have $\Box_3 = s_{18}$.

Consider an arbitrary \mathcal{S} -bad block, call it w . Fix now the element s_{19} at position i in w . It follows then that every position i modulo 4 must also feature s_{19} . If we fix s_{26} at position i in w , then every position i modulo 4 must also feature s_{26} . If w features s_{18} at position i , then position i modulo 8 must also feature s_{18} , and position $i + 4$ modulo 8 must feature s_{27} . We thus have the desired global bound M , since besides a finite amount of variation at the beginnings and ends of the word, every \mathcal{S} -bad block is essentially a proper power of one of the following words: $s_{18}\tau s_{19}\tau s_{27}\tau s_{19}\tau$, $s_{18}\tau s_{26}\tau s_{27}\tau s_{26}\tau$, $s_{18}\tau s_{18}\tau s_{27}\tau s_{27}\tau$, $s_{19}\tau$, $s_{19}\tau s_{27}\tau$, or $s_{27}\tau$.

□

Let \mathcal{B}_4 be the subgroup in \mathcal{B} that stabilizes the first four levels of the tree. Consider now the injective homomorphism:

$\phi: \mathcal{B}_4 \rightarrow \mathcal{B}^{16}$ that maps $g \in \mathcal{B}_4$ to the ordered 16-tuple $(\phi_{LLLL}(g), \phi_{LLLR}(g), \dots, \phi_{RRRR}(g))$.

We now propose the following bound on the number of ϵ -bad elements in \mathcal{B}_4 containing n non- τ letters, which we will denote by $b_\epsilon(n)$:

$$b_\epsilon(n) \leq 2 \binom{n}{\lfloor \epsilon n \rfloor} 31^{\lfloor \epsilon n \rfloor} M^{1+\lfloor \epsilon n \rfloor}$$

- The 2 accounts for the choice of starting with a τ -letter or not.
- The $\binom{n}{\lfloor \epsilon n \rfloor}$ accounts for the number of ways to designate positions for good by position letters.
- The $31^{\lfloor \epsilon n \rfloor}$ accounts for the maximum number of ways to place good by position letters.
- The $M^{1+\lfloor \epsilon n \rfloor}$ accounts for the maximum number of choices to place \mathcal{S} -bad blocks. The $1 + \lfloor \epsilon n \rfloor$ is explained by the fact that once we place our good by position letters, we break up the word into at most $1 + \lfloor \epsilon n \rfloor$ compartments to place \mathcal{S} -bad blocks.

Since γ is the lowest weight non- τ generator, a word of length r containing the most non- τ letters would alternate between γ -letters and τ -letters. We thus have that any word of length r contains at most $(r+3)/9$ non- τ letters. Denote by $B_\epsilon(r)$ the number of words in $\{g \in \mathcal{B}_4 \mid \ell(g) \leq r\}$ that contain at most ϵn non- τ letters. We then have the following inequality:

$$B_\epsilon(r) \leq \sum_{n=0}^{(r+3)/9} b_\epsilon(n) \leq \frac{2(r+3)}{9} \binom{(r+3)/9}{\lfloor \epsilon(r+3)/9 \rfloor} 31^{\lfloor \epsilon(r+3)/9 \rfloor} M^{1+\lfloor \epsilon(r+3)/9 \rfloor}$$

Proposition 4.12. \mathcal{B} has subexponential growth.

Proof. Assume now that \mathcal{B} has exponential growth. Then \mathcal{B}_4 has exponential growth as well [2]. Let $\lambda > 1$ be the growth rate of \mathcal{B}_4 . By Stirling's formula, choose $\epsilon > 0$ so that λ^r grows faster than the number of elements in $B_\epsilon(r)$. We then observe then that for any $g \in \mathcal{B}_4$ represented by a word w that contains a subword $v \in \mathcal{S}$, after splitting w at most three times, we either create a reducible word or create a good by nature element between two τ letters. Thus, after splitting w at most four times we are guaranteed to have reduction by at least some fixed amount. This is true since we either get immediate reduction in the former case, or get reduction after splitting again in the latter case. Since \mathcal{S} is a finite set, there is a word that gets the worst reduction such that we can guarantee any other word in \mathcal{S} reduces by at least that much. Thus there is a number η depending on ϵ that satisfies Proposition 3.3. This in turn implies that \mathcal{B}_4 has subexponential growth. □

Proposition 4.13. *\mathcal{B} has superpolynomial growth.*

Proof. Similar to the proof for \mathcal{A} . □

5. GROUPS NOT FROM POLYNOMIALS

Although many the most interesting groups we studied were the iterated monodromy groups of polynomials, not all of these groups had easy to identify growth. However, slight modifications to the group could result in a group of such simpler structure. No polynomial yields the groups \mathcal{Z}_k below, but as groups of intermediate growth on a binary tree, they are still somewhat enlightening.

5.1. Groups \mathcal{Z}_k have Intermediate Growth.

For $k \geq 3$, we define the group $\mathcal{Z}_k = \langle \tau, L_1, L_2, \dots, L_{n-1}, L_n \rangle$ by

$$\begin{aligned} L_1 &= (\tau, L_2) \\ L_k &= (1, L_1) \\ L_i &= (1, L_{i+1}) \quad \text{for } 1 < i < n, \end{aligned}$$

where τ is the transition element and the elements in parentheses are simply elements in bracket notation, mapped into the homomorphism of the direct product $\mathcal{Z}_k \times \mathcal{Z}_k$.

These groups have been previously shown to be of intermediate growth. In particular, the group \mathcal{Z}_3 is known as the Grigorchuk overgroup. Regardless, we offer an original proof here.

First, let us show each element L_i in any \mathcal{Z}_k is order 2 for all $1 \leq i \leq k$. Consider $(L_i)^2$ for any i . Taking advantage of the bracket notation's term-by-term multiplication, we may note $(L_i)^2 = (1, (L_j)^2)$ for some $1 \leq j \leq k$. However, in a similar manner, $(L_{i+1})^2 = (1, (L_{i+2})^2)$, so applying this repeatedly, we find $(L_i)^2 = (1, (1, (1, \dots)))$, which is the identity element.

Next, let us consider any element of the set

$$P = \{L_{j_1}L_{j_2}\dots L_{j_k} \mid j_i \text{ are unique integers from } 1 \text{ to } k\}$$

of products of all L_1, L_2, \dots, L_k in any order. Any element $p \in P$ will have $p = (\tau, p')$, for some $p' \in P$. Again, if we apply this repeatedly, we find $p = (\tau, (\tau, (\tau, \dots)))$ for all $p \in P$, and all $p \in P$ are the same element.

We may now apply this to show all L_i generating \mathcal{Z}_k are commutative. For any L_i, L_j , where without loss of generality, $i < j$, consider the elements

$$L_iL_jL_1L_2\dots L_{i-1}L_{i+1}\dots L_{j-1}L_{j+1}\dots L_{k-1}L_k$$

and

$$L_jL_iL_1L_2\dots L_{i-1}L_{i+1}\dots L_{j-1}L_{j+1}\dots L_{k-1}L_k.$$

Note these are both in P , so they are equal. Thus, multiplying each of them by

$$L_1L_2\dots L_{i-1}L_{i+1}\dots L_{j-1}L_{j+1}\dots L_{k-1}L_k$$

on the right, we find the two reduce to L_iL_j and L_jL_i , respectively, and so $L_iL_j = L_jL_i$, as desired.

Since the L_i are commutative and of order 2, we may now choose an alternating form for elements of \mathcal{Z}_k . Specifically, note we may write any element $z \in \mathcal{Z}_k$ in the minimum length form

$$z = \tau^{\epsilon_0}x_1\tau x_2\tau x_3\tau \dots x_{n-1}\tau x_n\tau^{\epsilon_1}$$

where $\epsilon_0, \epsilon_1 \in \{0, 1\}$ and $x_i \in \langle L_1, L_2, \dots, L_k \rangle$.

To show \mathcal{Z}_k has intermediate growth, we will apply Proposition 3.3 and a process of Bux and Perez to show it has sub-exponential growth and a technique given by Terpstra to show it has super-polynomial growth.

Theorem 5.1. *The group \mathcal{Z}_k has sub-exponential growth for all $k \in \mathbb{N}$, $k \geq 3$.*

To prove theorem 5.1, we shall apply a process given by Bux and Perez, first bounding the number of ϵ -bad elements in any ball of radius r , then finding a contradiction by assuming the group has exponential growth and noting this will require the group to meet the hypothesis for Proposition 2.0.1, which gives it sub-exponential growth.

Proof.

For our length function, we shall choose the weights, for $1 < i \leq k$:

$$\begin{aligned}\ell(\tau) &= k^2 \\ \ell(L_1) &= k \\ \ell(L_i) &= i - 1\end{aligned}$$

Let us now show that there is only a single non-reducing element in $C = \langle L_1, L_2, \dots, L_k \rangle$, which forces any non-reducing x_i to be that element. First, note since all L_i commute and are of order 2, an element c of C is of the form $c = (L_1)^{\epsilon_1} (L_2)^{\epsilon_2} \dots (L_k)^{\epsilon_k}$, where $\epsilon_i \in \{0, 1\}$.

Next, note τ has a length greater than the remaining generators combined. Thus, since L_1 is the only generator of C that gives a τ when put in bracket notation, any elements of C with $\epsilon_1 = 0$ will not contain a τ , and they will thus reduce.

For an element $c \in C$ with $\epsilon_1 = 1$, note $L_1 = (\tau, L_2)$. Thus, the reduction in length from putting L_1 in bracket notation is exactly $\ell(L_1) - \ell(L_2) = n - 1$. Furthermore, each L_i we include gives an increase in length of only 1 when put in bracket notation. Thus, if even a single ϵ_i is zero, the reduction from L_1 will be greater than the increase from all the L_i combined, giving c a net length reduction.

Thus, the only $c \in C$ that might not reduce is the element $L_1 L_2 \dots L_k$. Indeed, as $L_1 L_2 \dots L_k = (\tau, L_1 L_2 \dots L_k)$, we find it has exactly the length it began with.

For a given $r \in \mathbb{N}$, using the chosen weights, there exists an integer K_r such that an alternating word w of length $\ell(w) \leq r$ contains at most $K_r = \lfloor \frac{k^2+r}{k^2+1} \rfloor$ total x_i , where $k^2 + 1$ is the combined weight of a τ and the least weighted x_i , L_2 . Let $b_\epsilon(n)$ be the number of ϵ -bad alternating words that have exactly n x_i and represent an even element of \mathcal{Z}_k . Then,

$$b_\epsilon(n) \leq 2 \binom{n}{\lfloor \epsilon n \rfloor} (2^k - 1)^{\lfloor \epsilon n \rfloor},$$

where the leading 2 accounts for the choice of starting with τ or not, the binomial coefficient counts the number of way to allocate the $\lfloor \epsilon n \rfloor$ positions where reducing x_i may be placed, and the factor $(2^k - 1)^{\lfloor \epsilon n \rfloor}$ counts the ways we may choose which reducing x_i are placed in the allocated positions.

We can estimate the number $B_\epsilon(r)$ of ϵ -bad group elements of length at most r by noting:

$$B_\epsilon(r) \leq \sum_{n=0}^{K_r} b_\epsilon(n) \leq 2K_r \binom{K_r}{\epsilon K_r} (2^k - 1)^{\lfloor \epsilon K_r \rfloor}.$$

This is a direct application of the previous equation, using $\frac{k^2+r}{k^2+1}$ as the greatest possible n for the given r , then summing all the b_ϵ for that and lower n .

Now suppose \mathcal{Z}_k has exponential growth. Then the index 2 subgroup H_k of elements of \mathcal{Z}_k with an even number of τ has exponential growth with respect to the restricted length $\ell|_H$. Let $\lambda > 1$ be the growth rate of H_k . Using Stirling's formula, we can choose $\epsilon > 0$ small enough so that λ^r grows faster than $B_\epsilon(r)$. In particular, for r big enough, at least half of the elements in $\{z \in H_k \mid \ell(z) \leq r\}$ are ϵ -good (smaller r 's are covered by the constant K).

We finish the proof by observing that an ϵ -good element has either $n, n+1$, or $n-1$ copies of τ . If it has either n or $n+1$ such τ , then the ϵ -good element reduces by at least:

$$\eta = \frac{\epsilon n \left(\frac{k^2+k}{2} - 2 \right) + (1-\epsilon)n \frac{k^2+k}{2} + nk^2}{\epsilon n \left(\frac{k^2+k}{2} - 1 \right) + (1-\epsilon)n \frac{k^2+k}{2} + nk^2} < 1,$$

where the top of the fraction is the weight of the reduced element, and the bottom the pre-reduced. This is the worst case scenario, where each of the ϵn reducing x_i are the element $L_1 L_3 \dots L_k$, which has a weight of $\frac{k^2+k}{2} - 1$ but only reduces in length by 1.

If our ϵ -good element has $n-1$ copies of τ , then we will have an identical η if we simply increase the constant K by $\frac{k^2+k}{2}$. The worst case is attained when the only x_i of the ϵ -good element is the element $L_1 L_2 \dots L_n$.

We have thus verified the hypothesis for proposition 2.0.1 with $\eta < 1$, $p = \frac{1}{2}$, and $K = \frac{k^2+k}{2}$. Thus, \mathcal{Z}_k is found to have sub-exponential growth, contradicting our assumption that it has exponential growth. Therefore, \mathcal{Z}_k has sub-exponential growth. □

Next, we wish to show \mathcal{Z}_k has super-polynomial growth. For this, we shall use a technique similar to that used by Terpstra. To show \mathcal{Z}_k is commensurable with its direct product, and thus of super-polynomial growth, we will demonstrate a group \mathbb{B} is of finite index of \mathcal{Z}_k and is a subgroup of H_k . From there, we will conclude H_k is of finite index in \mathcal{Z}_k and its isomorphic copy is of finite index in $\mathcal{Z}_k \times \mathcal{Z}_k$, which is sufficient to show super-polynomial growth.

Theorem 5.2. *The group \mathcal{Z}_k has super-polynomial growth for all $k \in \mathbb{N}$, $k \geq 3$.*

Proof.

Choose $\mathbb{B} = \{zLz^{-1} \mid L \in \langle L_1, \dots, L_{k-1} \rangle; z \in \mathcal{Z}_k\}$, that is, the normal closure of $\langle L_1, \dots, L_{k-1} \rangle$. Note, then, \mathcal{Z}_k/\mathbb{B} is a group generated by the images of L_n and τ , so $[\mathcal{Z}_k : \mathbb{B}] \leq |\langle L_n, \tau \rangle| = 8$.

Since we have an injective homomorphism $\phi : H_k \rightarrow \phi(H_k)$, we know H_k is isomorphic to $\phi(H_k)$. Furthermore, note for each $L \in \langle L_1, \dots, L_{k-1} \rangle$, we have $(1, L')$ and $(L', 1)$ in $\phi(H_k)$, where if $L = L_{a1} L_{a2} \dots L_{am}$, then $L' = L_{a1+1} L_{a2+1} \dots L_{am+1}$.

Let $z_0 \in \mathcal{Z}_k$. Then there exists $z_1 \in \mathcal{Z}_k$ such that $(z_0, z_1) \in \phi(H_k)$. Thus, $(z_0, z_1)^{-1} = (z_0^{-1}, z_1^{-1}) \in \phi(H_k)$. Putting these together, we find $(z_0, z_1)(L, 1)(z_0^{-1}, z_1^{-1}) = (z_0 L z_0^{-1}, 1) \in \phi(H_k)$. Similarly, $(1, z_0 L z_0^{-1}) \in \phi(H_k)$. Note, the respective sets of all such $(z_0 L z_0^{-1}, 1)$ and $(1, z_0 L z_0^{-1})$ are exactly $\mathbb{B} \times 1$ and $1 \times \mathbb{B}$. However, if $\mathbb{B} \times 1$ and $1 \times \mathbb{B}$ are both subgroups of $\phi(H_k)$, then their product $\mathbb{B} \times \mathbb{B}$ is also a subgroup of $\phi(H_k)$.

Since \mathbb{B} has finite index in \mathcal{Z}_k , we know $\mathbb{B} \times \mathbb{B}$ has finite index in $\mathcal{Z}_k \times \mathcal{Z}_k$. Thus, since $\mathbb{B} \times \mathbb{B}$ is a subgroup of $\phi(H_k)$, we find $\phi(H_k)$ must also have finite index in $\mathcal{Z} \times \mathcal{Z}$. Therefore, $[\mathcal{Z} : H_k]$ and

$[\mathcal{Z} : \phi(H_k)]$ are both finite, where H_k is isomorphic to $\phi(H_k)$, and so \mathcal{Z}_k is commensurable with $\mathcal{Z}_k \times \mathcal{Z}_k$. Thus, \mathcal{Z}_k has super-polynomial growth, as desired. \square

We have found each \mathcal{Z}_k for $k \geq 3$ to have both sub-exponential and super-polynomial growth. Thus, each such \mathcal{Z}_k has intermediate growth.

6. POSSIBLE LIMITATIONS OF THE BUX-PÉREZ METHOD

6.1. Problems Assigning Weights. When attempting to find groups of intermediate growth using the Bux-Pérez method of assigning weights, we find that it is not always possible to assign weights to the standard generators to all groups. One such group is $\mathcal{D} = \langle \tau, a, b, c \rangle$ where

$$\tau = (01) \qquad a = (\tau, 1) \qquad b = (c, a) \qquad c = (1, b)$$

We found the following identities for this group:

$$\begin{aligned} \tau^2 &= a^2 = b^2 = c^2 = (ca)^2 = id \\ (a\tau)^4 &= (\tau a)^4 = (b\tau)^4 = (\tau b)^4 = (c\tau)^4 = (\tau c)^4 = id \\ ac &= ca \\ (abc)^4 &= (bac)^4 = id \end{aligned}$$

Problems arise when defining weights for the generators.

Proposition 6.1. *There is no assignment of weights to the generating set a, b, c, τ which extends to an admissible weight function.*

Proof. As with \mathcal{Z} we attempted to use the weight system [2] to try and prove subexponential growth. Using this method we want to ensure that every group element is represented by minimum length words that alternate between τ 's and the other generators. We also need that our elements do not increase in length when split into pair notation. Let $x = (x_0, x_1)$ represent any even element in \mathcal{D} . In order to analyze the length of elements in \mathcal{D} in pair form we will use the homomorphism

$$(\phi_0, \phi_1) : \mathcal{D}_1 \rightarrow \mathcal{D} \times \mathcal{D}$$

where $\phi_0(x) = x_0$ and $\phi_1(x) = x_1$. Letting $\gamma \in \langle a, b, c \rangle$, we see that in order for the length of any group element to not increase when split into pair notation, the following inequality must hold:

$$\ell(\tau) + \ell(\gamma) \geq \ell(\phi_0(\gamma)) + \ell(\phi_1(\gamma))$$

The elements of $\langle a, b, c \rangle$ expressed as minimum length words are as follows:

element	pair	element	pair	element	pair
a	$(\tau, 1)$	b	(c, a)	c	$(1, b)$
ac	(τ, b)	ab	$(\tau c, a)$	ba	$(c\tau, a)$
bc	(c, ab)	cb	(c, ba)	aba	$(\tau c\tau, a)$
bab	$(c\tau c, 1)$	bcb	$(1, aba)$	cbc	(c, bab)
abc	$(\tau c, ab)$	cba	$(c\tau, ba)$	bca	$(c\tau, ab)$
cab	$(\tau c, ba)$	$(ab)^2$	$((\tau c)^2, 1)$	$(bc)^2$	$(1, (ab)^2)$
$abca$	$(\tau c\tau, ab)$	$cbac$	$(c\tau, bab)$	$acba$	$(\tau c\tau, ba)$
$cabc$	$(\tau c, bab)$	$bcab$	$(c\tau c, aba)$	$babc$	$(c\tau c, b)$
$abcb$	(τ, aba)	$acbac$	$(\tau c\tau, bab)$	$abcab$	$((\tau c)^2, aba)$
$cbacb$	$(c\tau c, (ba)^2)$	$abcbc$	$(\tau, (ab)^2)$	$cbaba$	$((c\tau)^2, b)$
$(abc)^2$	$((\tau c)^2, (ab)^2)$	id	$(1, 1)$		

Assume $\ell : \mathcal{D} \rightarrow \mathbb{R}^+$ is an admissible weight function. We consider the weights of ab and cab :

$$\begin{aligned}
\ell(\tau) + \ell(ab) &\geq \ell(\tau c) + \ell(a) \Leftrightarrow \\
\ell(\tau) + \ell(a) + \ell(b) &\geq \ell(\tau) + \ell(c) + \ell(a) \Leftrightarrow \\
\ell(b) &\geq \ell(c) \\
\ell(\tau) + \ell(cabc) &\geq \ell(\tau c) + \ell(bab) \Leftrightarrow \\
\ell(\tau) + \ell(c) + \ell(a) + \ell(b) + \ell(c) &\geq \ell(\tau) + \ell(c) + \ell(b) + \ell(a) + \ell(b) \Leftrightarrow \\
\ell(c) &\geq \ell(b)
\end{aligned}$$

We see that $\ell(c) = \ell(b)$. Now considering the weights of b and aba :

$$\begin{aligned}
\ell(\tau) + \ell(b) &\geq \ell(c) + \ell(a) \Leftrightarrow \\
\ell(\tau) &\geq \ell(a) \\
\ell(\tau) + \ell(aba) &\geq \ell(\tau c\tau) + \ell(a) \Leftrightarrow \\
\ell(\tau) + \ell(a) + \ell(b) + \ell(a) &\geq \ell(\tau) + \ell(c) + \ell(\tau) + \ell(a) \Leftrightarrow \\
\ell(a) &\geq \ell(\tau)
\end{aligned}$$

We see that $\ell(a) = \ell(\tau)$. A problem occurs when analyzing the element $cbacb$. Considering the weight of $cbacb$ we get:

$$\begin{aligned}
\ell(\tau) + \ell(cbacb) &\geq \ell(c\tau c) + \ell(baba) \Leftrightarrow \\
\ell(\tau) + \ell(c) + \ell(b) + \ell(a) + \ell(c) + \ell(b) &\geq \ell(\tau) + \ell(c) + \ell(\tau) + \ell(b) + \ell(a) + \ell(b) + \ell(a) \Leftrightarrow \\
0 &\geq \ell(a)
\end{aligned}$$

This makes no sense however, therefore there is no applicable weight system for the group \mathcal{D} . \square

6.2. Difficulties Finding a Finite Primitive Set for the Iterated Monodromy Group of $z^3 + C$. Another such group is the group formed from the third degree polynomial $z^3 + C$, where $C \approx 0.635 - 0.645i$. Unlike \mathcal{D} , we are able to find an admissible set of weights for its generators. The problem with this group comes from the complexities in proving its growth. We find that unlike the groups \mathcal{A} and \mathcal{B} , we cannot find a finite primitive set \mathcal{S} such that the number of \mathcal{S} -bad blocks of a given length is bounded by some constant that is independent of the length. In this section we will form the automaton for this group and demonstrate the complexities when using the Bux-Pérez method.

The critical set for this polynomial is $C = \{0\}$. By approximating the value of C at $0.635 - 0.645i$, we calculate the post-critical set to be

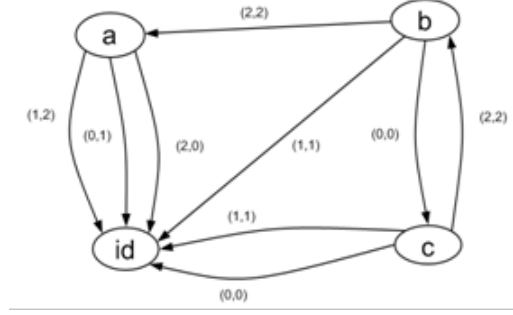
$$P = \{0.635 - 0.645i, 0.098 - 1.158i, 0.241 + 0.872i\}$$

which we will denote as $\{a, b, c\}$ respectively. We will label the curves of our spider to be γ_a , γ_b , and γ_c . We obtain the following critical portrait from this spider:

Using our critical portrait and the method illustrated by [1] we see the relations:

$$\begin{array}{lll}
a \cdot 0 = 1 \cdot id & b \cdot 0 = 0 \cdot c & c \cdot 0 = 0 \cdot id \\
a \cdot 1 = 2 \cdot id & b \cdot 1 = 1 \cdot id & c \cdot 1 = 1 \cdot id \\
a \cdot 2 = 0 \cdot id & b \cdot 2 = 2 \cdot a & c \cdot 2 = 2 \cdot b
\end{array}$$

These relations form the following automaton.



We will denote this group \mathcal{Z} . $\mathcal{Z} = \langle a, b, c \rangle$, where a , b , and c are defined by the equations:

$$\begin{aligned} a &= (012) \\ b &= (c, 1, a) \\ c &= (1, 1, b) \end{aligned}$$

The following identities stem from the generating set of \mathcal{Z} :

$$\begin{aligned} a^3 &= b^3 = c^3 = id \\ |ac| &= |ca| = 9 \\ |ab| &= |ba| = 27 \\ |bc| &= |cb| = 27 \end{aligned}$$

We will use the weight system defined by Bux-Pérez to try and prove \mathcal{Z} has sub exponential growth. Using this method we want to ensure that every group element is represented by minimum length words that alternate between a 's and/or a^{-1} 's and the other generators. We also need that our elements do not increase in length when split into triplet notation. Let $\xi = (x_0, x_1, x_2)$ represent any element in \mathcal{Z}_1 . In order to analyze the length of elements in \mathcal{Z}_1 in triplet form we will use the homomorphism

$$(\phi_0, \phi_1, \phi_2) : \mathcal{Z}_1 \rightarrow \mathcal{Z} \times \mathcal{Z} \times \mathcal{Z}$$

where $\phi_0(\xi) = x_0$, $\phi_1(\xi) = x_1$, and $\phi_2(\xi) = x_2$. Letting ℓ represent our length function, $\epsilon \in \{-1, 1\}$, and $\sigma \in \langle b, c \rangle$, we see that in order for the length of any group element to not increase when split into triplet notation, the following inequality must hold:

$$\ell(a^\epsilon) + \ell(\sigma) \geq \ell(\phi_0(\sigma)) + \ell(\phi_1(\sigma)) + \ell(\phi_2(\sigma))$$

The weight of any word is the sum of the weights of the generators that define it. In order to find weights for a , b , and c consider the elements b , cbc , and $(bc)^2$. Applying the above inequality to b and cbc we get:

$$\begin{aligned} \ell(a^\epsilon) + \ell(b) &\geq \ell(c) + \ell(1) + \ell(a) \Leftrightarrow \\ \ell(b) &\geq \ell(c) \\ \ell(a^\epsilon) + \ell(cbc) &\geq \ell(c) + \ell(1) + \ell(bab) \Leftrightarrow \\ \ell(a^\epsilon) + \ell(c) + \ell(b) + \ell(c) &\geq \ell(c) + \ell(b) + \ell(a) + \ell(b) \Leftrightarrow \\ \ell(c) &\geq \ell(b) \end{aligned}$$

So we see that the weights of b and c must be equal. Applying the above inequality to $(bc)^2$ we get:

$$\begin{aligned} \ell(a^\epsilon) + \ell(bc bc) &\geq \ell(c) + \ell(1) + \ell(abab) \Leftrightarrow \\ \ell(a^\epsilon) + \ell(b) + \ell(c) + \ell(b) + \ell(c) &\geq \ell(c) + \ell(1) + \ell(a) + \ell(b) + \ell(a) + \ell(b) \Leftrightarrow \\ \ell(b) &\geq \ell(a) \end{aligned}$$

Therefore $a \leq b = c$. We will choose the weights:

$$\ell(a) = q, \quad \ell(b) = p, \quad \ell(c) = p$$

Assume $p > q$. We will choose to define words $\sigma_b \in \langle b, c \rangle$ as bad if they follow the equality

$$\ell(a^\epsilon) + \ell(\sigma) = \ell(\phi_0(\sigma)) + \ell(\phi_1(\sigma)) + \ell(\phi_2(\sigma))$$

Claim 1. Any word with only one b^{ϵ_n} will be bad.

Proof. Notice that by our definition of a bad word, the ratio of the weight of some any word σ_b plus some a^{ϵ_n} and the weight of its triple form is going to be 1. So any good word σ_g must satisfy the following inequality:

$$\frac{\ell(\phi_0(\sigma_g)) + \ell(\phi_1(\sigma_g)) + \ell(\phi_2(\sigma_g))}{\ell(a^\epsilon) + \ell(\sigma_g)} < 1$$

It can easily be checked that c^{ϵ_n} is a good word. Let us now analyze words of the form $b^{\epsilon_1}c^{\epsilon_2}$. This word will have the triple form $(c^{\epsilon_1}, 1, a^{\epsilon_1}b^{\epsilon_2})$ Where $\epsilon_n \in \{-1, 1\}$. It can easily be seen that any word of this form will be bad. By adding another b^{ϵ_3} to this word we are adding p to its weight, thus $\ell(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}) = \ell(b^{\epsilon_1}c^{\epsilon_2}) + p$. Looking at $\phi_0(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})$ we get $c^{\epsilon_1}c^{\epsilon_3}$, which is simply c^{θ_1} , where $\theta_n \in \{-1, 0, 1\}$. This means that either $\ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})) = \ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2}))$ or $\ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})) = \ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2})) - p$, depending on what c^{ϵ_1} and c^{ϵ_3} are. Since ϕ_1 of any element of $\langle b, c \rangle$ is always the identity, whenever any b^{ϵ_n} or c^{ϵ_n} is added to any of these words $\ell(\phi_1)$ of that word is always 0. Now, looking at $\phi_2(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})$ we get $a^{\epsilon_1}b^{\epsilon_2}a^{\epsilon_3}$. We see that $\ell(\phi_2(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})) = \ell(\phi_2(b^{\epsilon_1}c^{\epsilon_2})) + q$. Therefore the length of the new word will increase by p while the length of the new word's triple will increase by q or decrease by $p - q$ depending on c^{θ_1} 's. If this new word's triple increases by q then we will have that $\theta_1 \in \{-1, 1\}$, but if it decreases by $p - q$ we will have that $\theta_1 = 0$ or, in other words, the identity. We see that the ratio of the weight of $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}$ and its triple is $\frac{2q+p}{3p+q}$ or $\frac{2q+2p}{3p+q}$ depending on c^{θ_1} . In either case $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}$ is good.

Now we have a word of the form $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}$. By adding a c^{ϵ_4} to this word we see that we once again add p to the weight of $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}$, thus $\ell(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4}) = \ell(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}) + p$. Since $c^{\epsilon_4} = (1, 1, b^{\epsilon_4})$, we see that $\ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4})) = \ell(\phi_0(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}))$ and $\ell(\phi_1(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4})) = \ell(\phi_1(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}))$. In $\phi_2(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4})$ we will now have $a^{\epsilon_1}b^{\epsilon_2}a^{\epsilon_3}b^{\epsilon_4}$ so $\ell(\phi_2(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4})) = \ell(\phi_2(b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3})) + p$, therefore the total weight added to the new word's triplet form is just p . We see that the ratio of the weight of $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4}$ and its triple is $\frac{2q+2p}{4p+q}$ or $\frac{3p+2q}{4p+q}$ depending on c^{θ_1} . In either case $b^{\epsilon_1}c^{\epsilon_2}b^{\epsilon_3}c^{\epsilon_4}$ is good. Notice that when adding a c^{ϵ_n} to any string of alternating b^{ϵ_n} 's and c^{ϵ_n} 's ending with a b^{ϵ_n} it adds the same amount of weight to the word and its triple. This is due to c^{ϵ_n} only adding weight to $\ell(\phi_2)$ by adding a b^{ϵ_n} to a string of alternating b^{ϵ_n} 's and a^{ϵ_n} 's. We see that because $\ell(c^{\epsilon_n}) = \ell(b^{\epsilon_n})$ and b^{ϵ_n} is the only element in the triplet form of c^{ϵ_n} , adding a c^{ϵ_n} to any word σ of alternating b^{ϵ_n} 's and c^{ϵ_n} 's will only add p to $\ell(a^\epsilon) + \ell(\sigma)$ and p to $\ell(\phi_0(\sigma)) + \ell(\phi_1(\sigma)) + \ell(\phi_2(\sigma))$. This means that c^{ϵ_n} never changes a word from good to bad or bad to good since it can never make the fraction:

$$\frac{\ell(\phi_0(\sigma)) + \ell(\phi_1(\sigma)) + \ell(\phi_2(\sigma))}{\ell(a^\epsilon) + \ell(\sigma)} = R(\sigma)$$

change from being less than one to greater than one or from being greater than one to less than one.

We now have a word of the form $b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4} = (c^{\theta_1}, 1, a^{\epsilon_1} b^{\epsilon_2} a^{\epsilon_3} b^{\epsilon_4})$. Adding some $b^{\epsilon_5} = (c^{\epsilon_5}, 1, a^{\epsilon_5})$ to this string will once again add q to $\ell(\phi_2(b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4}))$, do nothing to $\ell(\phi_1(b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4}))$, and, depending on c^{θ_1} and c^{ϵ_5} , either increase $\ell(\phi_3(b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4}))$ by q , decrease it by p , or keep it the same. If $\epsilon_5 = \theta_1 \neq 0$ or $\epsilon_5 \neq \theta_1 = 0$ then the ratio of the weight of $b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4} b^{\epsilon_5}$ plus the weight of a^c and its triple is $\frac{3q+3p}{5p+q}$. If $\epsilon_5 \neq \theta_1 \neq 0$ then the ratio of the weight of $b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4} b^{\epsilon_5}$ plus the weight of a^c and its triple is $\frac{2p+3q}{5p+q}$. In any of these cases, $b^{\epsilon_1} c^{\epsilon_2} b^{\epsilon_3} c^{\epsilon_4} b^{\epsilon_5}$ is good. We will consider the worst case scenario of $R(\sigma)$ to be when the c^{θ_n} in $\phi_0(\sigma)$ is not the identity. Following this pattern, we see that the worst case scenario of $R(\sigma)$ for some k number of $b^{\epsilon_n} c^{\epsilon_n}$'s will have the form $\frac{p+k(p+q)}{q+k(2p)}$. By adding a c^{ϵ_n} to the beginning or a b^{ϵ_n} to the end of those k number of $b^{\epsilon_n} c^{\epsilon_n}$'s, $R(\sigma)$ will have the form $\frac{2p+k(p+q)}{p+q+k(2p)}$ or $\frac{p+q+k(p+q)}{p+q+k(2p)}$ respectively. Thus, we have the equations:

$$\begin{aligned} R((b^{\epsilon_n} c^{\epsilon_n})^k) &= \frac{p+k(p+q)}{q+k(2p)} \\ R(c^{\epsilon_n} (b^{\epsilon_n} c^{\epsilon_n})^k) &= \frac{2p+k(p+q)}{p+q+k(2p)} \\ R((b^{\epsilon_n} c^{\epsilon_n})^k b^{\epsilon_n}) &= \frac{p+q+k(p+q)}{p+q+k(2p)} \end{aligned}$$

As k approaches infinity we see that all three of these equations approach $\frac{(p+q)}{(2p)}$, which is clearly less than 1. Therefore, for any $k \geq 2$ we see that $R((b^{\epsilon_n} c^{\epsilon_n})^k) < 1$, $R(c^{\epsilon_n} (b^{\epsilon_n} c^{\epsilon_n})^k) < 1$, and $R((b^{\epsilon_n} c^{\epsilon_n})^k b^{\epsilon_n}) < 1$. Therefore any words of the form $(b^{\epsilon_n} c^{\epsilon_n})^k$, $c^{\epsilon_n} ((b^{\epsilon_n} c^{\epsilon_n})^k)$, or $(b^{\epsilon_n} c^{\epsilon_n}) b^{\epsilon_n}$ will be good. For $k \in \{0, 1\}$, the only words that will be bad are words of the form $b^{\epsilon_n} c^{\epsilon_n}$, $c^{\epsilon_n} b^{\epsilon_n} c^{\epsilon_n}$, or b^{ϵ_n} . It is trivial to check that words of the form $c^{\epsilon_n} b^{\epsilon_n}$ are also bad. Therefore words with only one b^{ϵ_n} are bad. If $p = q$ we see that every worst case scenario for $R(\sigma)$ will be equal to 1 thus making all of the worst case schenario words bad, vastly increasing our number of bad words. We need a minimal case of bad words so we will only look at the case where $p > q$. \square

This means our list of bad words is:

Word	triple	Word	triple
b	$(c, 1, a)$	b^{-1}	$(c^{-1}, 1, a^{-1})$
bc	$(c, 1, ab)$	$b^{-1}c$	$(c^{-1}, 1, a^{-1}b)$
bc^{-1}	$(c, 1, ab^{-1})$	$b^{-1}c^{-1}$	$(c^{-1}, 1, a^{-1}b^{-1})$
$c^{-1}b$	$(c, 1, b^{-1}a)$	$c^{-1}b^{-1}$	$(c^{-1}, 1, b^{-1}a^{-1})$
cb	$(c, 1, ba)$	cb^{-1}	$(c^{-1}, 1, ba^{-1})$
cbc	$(c, 1, bab)$	$cb^{-1}c$	$(c^{-1}, 1, ba^{-1}b)$
$c^{-1}bc$	$(c, 1, b^{-1}ab)$	$c^{-1}b^{-1}c$	$(c^{-1}, 1, b^{-1}a^{-1}b)$
$c^{-1}bc^{-1}$	$(c, 1, b^{-1}ab^{-1})$	$cb^{-1}c^{-1}$	$(c^{-1}, 1, ba^{-1}b^{-1})$
cbc^{-1}	$(c, 1, bab^{-1})$	$c^{-1}b^{-1}c^{-1}$	$(c^{-1}, 1, b^{-1}a^{-1}b^{-1})$

We see that any bad word has the form $\Psi_\alpha = (c^{\epsilon_n}, 1, \psi_\alpha)$, Where Ψ_α is any bad element of \mathcal{Z} , $\epsilon_n \in \{-1, 1\}$, and $\psi_\alpha \in \langle a, b \rangle$ is the word of alternating a^{ϵ_n} 's and b^{ϵ_n} 's that correspond to Ψ_α .

Conjecture 6.2. *For this generating set, there is no finite primitive set \mathcal{S} for the group \mathcal{Z} such that the number of \mathcal{S} -bad blocks of a given length is bounded by some constant independent of the length.*

To support this conjecture we will illustrate the process of finding the primitive set, \mathcal{S} , of \mathcal{Z} . We will use this general form to analyze these bad words. In attempting to find our primitive set we must find patterns of bad words that either reduce or give good words when split. This is

analogous to the process for finding the primitive sets for \mathcal{A} and \mathcal{B} . We want to look at repeated and alternating strings of $a\Psi_\alpha$ and $a^{-1}\Psi_\beta$ and, where $\Psi_\alpha, \Psi_\beta \in \mathcal{Z}_b$.

Case I:

We will first analyze the case where we have a switch from a^{-1} 's to a 's in some string of alternating a^ϵ and bad words. Notice that in the area of a string where such a switch occurs we will have $a^\epsilon\Psi_1a^{-1}\Psi_2a\Psi_3$. This $a^\epsilon\Psi_1a^{-1}\Psi_2a\Psi_3 = a^\epsilon(c^{\epsilon_1}, 1, \psi_1)(1, \psi_2, c^{\epsilon_2})(c^{\epsilon_3}, 1, \psi_3)$ and when multiplying the triples out we get $a^\epsilon(c^{\theta_1}, \psi_2, \psi_1\psi_3)$. Since $c^{\epsilon_1}c^{\epsilon_3}$ multiplies to form a single c^{θ_1} , we see that we have reduction on the second level. Therefore any string of alternating a^ϵ 's and bad words that contains a switch from an a^{-1} and a bad word to an a and a bad word will have reduction on the second level.

Case II:

We will now analyze the case where there are no switches from a to a^{-1} and vice versa. We are able to see that for some $m \in \{0, 1, 2, \dots\}$ that any string of $a\Pi_{\alpha_m}$ will have one of the following forms:

$$\begin{aligned} \prod_{i=1}^{3m} (a\Psi_{\alpha_i}) &= [U, V, W] \\ \prod_{i=1}^{3m+1} (a\Psi_{\alpha_i}) &= \left[U\psi_{\alpha_{(3m+1)}}, Vc^{\epsilon_{(3m+1)}}, W \right] a \\ \prod_{i=1}^{3m+2} (a\Psi_{\alpha_i}) &= \left[U\psi_{\alpha_{(3m+1)}}, Vc^{\epsilon_{(3m+1)}}\psi_{\alpha_{(3m+2)}}, Wc^{\epsilon_{(3m+2)}} \right] a^{-1} \end{aligned}$$

where

$$\begin{aligned} U &= \prod_{i=1}^m (\psi_{\alpha_{(3i-2)}} c^{\epsilon_{(3i)}}) \\ V &= \prod_{i=1}^m (c^{\epsilon_{(3i-2)}} \psi_{\alpha_{(3i-1)}}) \\ W &= \prod_{i=1}^m (c^{\epsilon_{(3i-1)}} \psi_{\alpha_{(3i)}}) \end{aligned}$$

We are also able to see that for some $n \in \{0, 1, 2, \dots\}$ that any string of $a^{-1}\Pi_{\beta_n}$ will have one of the following forms:

$$\begin{aligned} \prod_{j=1}^{3n} (a^{-1}\Psi_{\beta_j}) &= [U', V', W'] \\ \prod_{j=1}^{3n+1} (a^{-1}\Psi_{\beta_j}) &= \left[U', V'\psi_{\beta_{(3m+1)}}, W'c^{\epsilon_{(3m+1)}} \right] a^{-1} \\ \prod_{j=1}^{3n+2} (a^{-1}\Psi_{\beta_j}) &= \left[U'\psi_{\beta_{(3m+2)}}, V'\psi_{\beta_{(3m+1)}}c^{\epsilon_{(3m+2)}}, W'c^{\epsilon_{(3m+1)}} \right] a \end{aligned}$$

where

$$\begin{aligned}
U' &= \prod_{j=1}^n (\psi_{\beta_{(3j-1)}} c^{\epsilon(3j)}) \\
V' &= \prod_{j=1}^n (\psi_{\beta_{(3j-2)}} c^{\epsilon(3j-1)}) \\
W' &= \prod_{j=1}^n (c^{\epsilon(3j-2)} \psi_{\beta_{(3j)}})
\end{aligned}$$

For either case, there is no way to get reduction on the second level. There are, however, ways to produce good generators on the second level. It can be seen that the only way to produce good generators in either case is if adjacent bad words make an $ab^\epsilon c^\epsilon b^\epsilon a$ or $ac^\epsilon a$ on the second level.

Case III:

We will now look at the cases where we have a string of alternating a^ϵ 's and bad words that contains a switch from an a and a bad word to an a^{-1} and a bad word. There are nine possible combinations that can occur depending on where the switch takes place in the string. These combinations are as follows:

$$\begin{aligned}
\prod_{i=1}^{3m} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n} (a^{-1}\Psi_{\beta_j}) &= [UU', VV', WW'] \\
\prod_{i=1}^{3m} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+1} (a^{-1}\Psi_{\beta_j}) &= [UU', VV'\psi_{\beta_{(3m+1)}}, WW'c^{\epsilon(3m+1)}] a^{-1} \\
\prod_{i=1}^{3m} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+2} (a^{-1}\Psi_{\beta_j}) &= [UU'\psi_{\beta_{(3m+2)}}, VV'\psi_{\beta_{(3m+1)}}c^{\epsilon(3m+2)}, WW'c^{\epsilon(3m+1)}] a \\
\prod_{i=1}^{3m+1} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}W', Vc^{\epsilon(3m+1)}U', WW'] a \\
\prod_{i=1}^{3m+1} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+1} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}W'c^{\epsilon(3m+1)}, Vc^{\epsilon(3m+1)}U', WW'\psi_{\beta_{(3m+1)}}] \\
\prod_{i=1}^{3m+1} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+2} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}W'c^{\epsilon(3m+1)}, Vc^{\epsilon(3m+1)}U'\psi_{\beta_{(3m+2)}}, WW'\psi_{\beta_{(3m+1)}}c^{\epsilon(3m+2)}] a^{-1} \\
\prod_{i=1}^{3m+2} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}V', Vc^{\epsilon(3m+1)}\psi_{\alpha_{(3m+2)}}W'\psi_{\beta_{(3m+1)}}, Wc^{\epsilon(3m+2)}U'] a^{-1} \\
\prod_{i=1}^{3m+2} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+1} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}V'\psi_{\beta_{(3m+1)}}, Vc^{\epsilon(3m+1)}\psi_{\alpha_{(3m+2)}}W'c^{\epsilon(3m+1)}, Wc^{\epsilon(3m+2)}U'] a \\
\prod_{i=1}^{3m+2} (a\Psi_{\alpha_i}) \prod_{j=1}^{3n+2} (a^{-1}\Psi_{\beta_j}) &= [U\psi_{\alpha_{(3m+1)}}V'\psi_{\beta_{(3m+1)}}c^{\epsilon(3m+2)}, Vc^{\epsilon(3m+1)}\psi_{\alpha_{(3m+2)}}W'c^{\epsilon(3m+1)}, Wc^{\epsilon(3m+2)}U'\psi_{\beta_{(3m+2)}}]
\end{aligned}$$

We can see that there are possibilities for reduction and the production of good generators in the second level of all nine combinations. This is still only possible through adjacent bad words however. We see that when defining our primitive set we only need to look at alternating strings of a^ϵ 's and bad words that contain only two or less bad words. There are only a finite number of ways to keep pairs of $a^\epsilon\Psi$'s bad. A problem arises when attempting to find all the ways to keep these pairs bad when attaching them to other pairs. There appears to be an infinite number of ways to attach $a^{-1}\Psi$ pairs to other $a^{-1}\Psi$ pairs without getting reduction or producing a good generator. Similarly, there also appears to be an infinite number of ways to attach $a\Psi$ pairs to other $a\Psi$ pairs without getting reduction or producing any good generators. Even still there are an infinite number of places to switch from $a\Psi$ pairs to $a^{-1}\Psi$ pairs. This means that it seems to be impossible to find a finite number of ways to make bad words in order to define our primitive set.

7. ACKNOWLEDGEMENTS

We would like to thank our seminar director, Dr. Daniel Farley, for his guidance and expertise throughout our research. We would also like to thank our graduate assistant, Rachel Karpman, for her support through many long hours and late nights. Our gratitude goes out to all the faculty at Miami University involved in SUMSRI for their contributions and commitment to helping undergraduates. We also thank the National Science Foundation, the National Security Agency, and Miami University for making the research program possible.

REFERENCES

- [1] V. Nekrashevych, *Self-Similar Groups*, American Mathematical Society, Providence, RI, 2005.
- [2] K. Bux, R. Pérez *On the Growth of Iterated Monodromy Groups*, preprint, 2004.
- [3] A. Terpstra *The Grigorchuk Group, the Burnside Problem, and Milnor's Question*, unpublished.
- [4] L. Bartholdi, V. Nekrashevych *Iterated Monodromy Groups of Quadratic Polynomials, I*, preprint 2006.
- [5] R. Grigorchuk, I Pak, *Groups of Intermediate Growth: An Introduction for Beginners*, Enseign. Math. (2) **54**(2008), no. 3-4, 251-272

(DOUGHERTY) DEPARTMENT OF EDUCATION, DEPARTMENT OF MATHEMATICS, KUTZTOWN UNIVERSITY KUTZTOWN, PA 19530 GFFADOUGHERTY@YAHOO.COM

(KINDELIN) DEPARTMENT OF MATHEMATICS, UNIVERSITY OF DAYTON, DAYTON, OH 45302 KINDELIN1@NOTES.UDAYTON.EDU

(REAVES) DEPARTMENT OF MATHEMATICS, MOREHOUSE COLLEGE, ATLANTA, GA 30314 REAVESAM@GMAIL.COM

(WALKER) DEPARTMENT OF PHYSICAL SCIENCES, YORK COLLEGE OF PENNSYLVANIA, YORK, PA 17403 AWALKER6@YCP.EDU

(ZAKAHI) DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, DENISON UNIVERSITY, GRANVILLE, OH 43023 ZAKAHLN@DENISON.EDU