GROUPS WITH CONTEXT-FREE CO-WORD PROBLEM AND EMBEDDINGS INTO THOMPSON'S GROUP V

ROSE BERNS-ZIEVE, DANA FRY, JOHNNY GILLINGS, HEATHER MATHEWS

Abstract. We describe a generalized Thompson group $V_{(G,\theta)}$ for each finite group G with homomorphism θ: $G \to G$. Our group is constructed using the *cloning map* concept introduced by Witzel and Zaremsky. We prove that $V_{(G,\theta)}$ is co$\mathcal{CF}$ for any homomorphism θ and finite group G by constructing a pushdown automaton and showing that the co-word problem of $V_{(G,\theta)}$ is the cyclic shift of the language our automaton accepts.

Lehnert conjectures that a group has context-free co-word problem exactly if it is a finitely generated subgroup of V. The groups $V_{(G,\theta)}$ where θ is not the identity homomorphism do not appear to have obvious embeddings into V.

We build off the work of Bleak and Salazar-Diaz regarding demonstrative groups. We extend the class of known finitely generated demonstrative subgroups of V to include virtually cyclic groups. Additionaly we construct embeddings for amalgamated free products where the amalgamated group is a demonstrative subgroup of V.

1. Introduction

Let G be a group and let $\Sigma \subseteq G$ be a finite set that generates G. Then, the *word problem of G with respect to the free monoid* Σ^* is the set of all words in Σ^* that are equivalent to the the identity in G. The *co-word problem of G with respect to* Σ^* is the complement of the word problem. Both the word problem and the co-word problem of G are languages. The Chomsky Hierarchy is a containment hierarchy of the classes of formal languages. It states that the set of regular languages is a subset of context-free languages; the set of context-free languages is a subset of context sensitive languages; and the set of context sensitive languages is a subset of recursive languages. We will focus in particular on context-free languages. A language is context-free if it is accepted by a pushdown automaton. If the co-word problem of G is a context-free language, then we say G is co$\mathcal{CF}$. This property does not depend on the choice of generators for the monoid generating set. The class of co$\mathcal{CF}$ groups was first studied by Holt, Rees, Röver, and Thomas [5]. They showed that the class is closed under taking finite direct products, taking restricted standard wreath products with virtually free top groups, and passing to finitely generated subgroups and finite index overgroups.

One group of particular interest is Thompson's group V, which is an infinite but finitely presented simple group. Lehnert and Schweitzer demonstrate that Thompson's group V is co$\mathcal{CF}$. This group is of interest to us because of the conjecture, formulated by Lehnert and revised by Bleak, Matucci, and Neunhöffer, that any group with context-free co-word problem embeds in V, i.e.,

Conjecture 1.1. Thompson's group V is a universal co$\mathcal{CF}$ group.

In this paper we prove two classes of results, one related to embeddings into V, and the other questioning if conjecture 1.1 is true.

Bleak and Salazar-Diaz define the class of demonstrative subgroups and use this class to produce embeddings of free products and wreath products into V. They also show that the class of groups that embed into V is closed under taking finite index overgroups. Their method of proof appeals to results of Kaloujnine and Krasner. Here, we use induced actions to give a direct proof of the latter result. Our method of proof shows, moreover, that if the original embedding is demonstrative, then so is the embedding of the finite index overgroup. We also give a direct proof that the amalgamated free product $G_1 *_H G_2$ embeds into V whenever G_1 and G_2 are virtually cyclic and H has finite index in both factors.

It appears that the virtually cyclic groups are the only finitely generated groups known to have demonstrative embeddings into V. If V is a universal co$\mathcal{CF}$ group, then it should be possible, by the results of Holt, Rees, Röver, and Thomas, to find an embedding of $G \wr F_2$ into V, where G is co$\mathcal{CF}$ and F_2 is the free group on two generators. The easiest way to find such an embedding would be to show that F_2 has a demonstrative embedding into V. We are thus led to ask:

Question 1.2. *Does there exist a demonstrative embedding of F_2 into V?*

We do not see a simple solution for this question. In other attempts to verify conjecture 1.1, we consider the work of Witzel and Zaremsky. We look at a specific group $V_{(G,\theta)}$ that arises from their family of groups equipped with a cloning system. The group $V_{(G,\theta)}$ appears to be quite similar to V. We define a surjective homomorphism Φ from $V_{(G,\theta)} \to V$, which implies that $V_{(G,\theta)}$ acts on the Cantor set. However, by our construction, $V_{(G,\theta)}$ seems to have no obvious faithful actions on the Cantor set when θ is not the identity homomorphism.

In our main result, we prove that $V_{(G,\theta)}$ is co$\mathcal{CF}$ for all pairs of θ and finitely generated G. We begin by detailing a construction of a pushdown automaton and we show that the co-word problem is equivalent to the cyclic shift of the language accepted by the automaton, therefore proving that the co-word problem is context-free.

This result may imply that, for some θ, $V_{(G,\theta)}$ could be a context-free group that does not embed into V. This would be a startling counter-example to conjecture 1.1.

We briefly outline the paper. Section 2 provides the necessary background for the reader to understand the concepts discussed in the two following sections. In section 3, we give our proofs and examples of all ideas related to demonstrative subgroups. Finally, in section 4 we prove the main result of our paper.

2. Background

2.1. **Pushdown Automata.**

Definition 2.1. Let Σ be a finite set, called an *alphabet.* We call elements of the alphabet *symbols.* The *free monoid* on Σ, denoted Σ^*, is the set of all finite strings of symbols from Σ. This is inclusive of the empty string, which we denote as ϵ. The operation is concatenation. An element of the free monoid is a *word.* A subset of the free monoid is a *language.*

Example 2.2. Let $\Sigma = \{0, 1\}$. The free monoid Σ^* contains all finite concatenations of 0 and 1 in any order. An example word is 01101.

Definition 2.3. Let G be a group. A *finite monoid generating set* is a finite alphabet Σ with a surjective monoid homomorphism $\Phi : \Sigma^* \to G$. The *word problem* of a group G (with respect to Σ), denoted $WP_\Sigma(G)$, is the kernel of Φ. The complement of the word problem is the *co-word problem*, denoted $CoWP_\Sigma(G)$.

Definition 2.4. Let Σ, Γ be alphabets and let $\#$ be an element of Γ. A *pushdown automaton* with stack alphabet Γ and input alphabet Σ is defined as a directed graph with a finite set of vertices V, a set of *transitions* (directed edges) δ, an initial state $v_0 \in V$, and a set of *terminal states* $T \subseteq V$.

A transition is labeled by an ordered triplet $(w_1, w_2, w_3) \in (\Sigma \cup \{\epsilon\}) \times \Gamma^* \times \Gamma^*$. When *following* a transition, the pushdown automaton (PDA) reads and deletes w_1 from its input tape, reads and deletes w_2 from its memory stack (shortened to stack for the duration of this paper), and writes w_3 on its stack. If w_1, w_2, or w_3 equals ϵ, the automaton does not execute the action associated with that coordinate. We only consider *generalized* PDA, which can add multiple words on its stack at a time. A PDA accepts languages either by terminal state, or by empty stack. This must be specified upon creation of the automaton. See Definitions 2.6 and 2.7.

Definition 2.5. ([3],Definition 2.6) Let P be a pushdown automaton. We describe a class of directed paths in P, called the *valid paths*, by induction on length. The path of length 0 starting at the initial vertex $v_0 \in P$ is valid; its stack value is $\# \in \Gamma^*$. Let $t_1 \dots t_n (n \geq 0)$ be a valid path in P, where t_1 is the transition crossed first. Let t_{n+1} be a transition whose initial vertex is the terminal vertex of t_n; we suppose that the label of t_{n+1} is (s, w_1, w_2). The path $t_1 \dots t_n t_{n+1}$ is also valid, provided that the stack value of $t_1 \dots t_n$ has w_1 as a prefix; that is, if the stack value of $t_1 \dots t_n$ has the form $w_1 w' \in \Gamma^*$ for some $w' \in \Gamma^*$. The stack value of $t_1 \dots t_n t_{n+1}$ is then $w_2 w'$. We let *val(p)* denote the stack value of a valid path p.

The *label* of a valid path $t_1 \dots t_n$ is $s_n \dots s_1$, where s_i is the first coordinate of the label for t_i (an element of Σ, or the empty string). The label of a valid path p will be denoted $\ell(p)$.

Definition 2.6. Let p be a valid path of a pushdown automaton P. We say that p is a *successful path* when

(1) $val(p) = \epsilon$ if P accepts by empty stack, or
(2) The terminal vertex of p is in T if P accepts by terminal state.

Definition 2.7. Let P be a PDA. The language *accepted by* P, denoted $\mathcal{L}_P$, is
$\mathcal{L}_P = \{w \in \Sigma^* \mid w = \ell(p)$ for some successful path p$\}$.

Definition 2.8. A subset of the free monoid Σ^* is called a *(non-deterministic) context-free language* if it is $\mathcal{L}_P$ for some pushdown automaton P.

Let P be a PDA. We operate P in the following way:

(1) A word $\hat{w}$ is placed on the input tape and read symbol by symbol. By our convention, P reads the input tape from right to left.

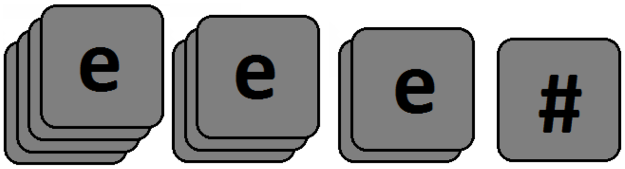

Figure 1. The stack of a PDA. As an element is read from the top, the next element appears as the new top if nothing is written to the stack.

(2) Next, P follows transitions until $\hat{w} = \ell(p)$ for some successful path p.
(3) If some successful path p exists, P accepts $\hat{w}$.

2.2. **Thompson's Group V.**

Definition 2.9. A *metric space* S is a set where a function d: $S \times S \to \mathbb{R}$ acts as a global distance measure, called a *metric*, with the following conditions for any three points, $x_1, x_2, x_3 \in S$:

(1) $d(x_1, x_2) \geq 0$
(2) $d(x_1, x_2) = 0 \iff x_1 = x_2$
(3) $d(x_1, x_2) = d(x_2, x_1)$
(4) (Triangle Inequality) $d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$

Definition 2.10. Let $X = \{0, 1\}$ and consider X^*. As in [1], we define an infinite rooted tree, $\mathcal{T}_2$, as follows:

Let X* be the set of nodes for $\mathcal{T}_2$. For $u, v \in X^*$, there exists an edge from u to v if $ux = v$ for some $x \in X$. Suppose $uw = v$ for some $w \in X^*$. Then we say u is the *prefix* of v and we define the set $u* = \{v \mid uw = v \text{ for some } w \in \mathcal{T}_2\}$.

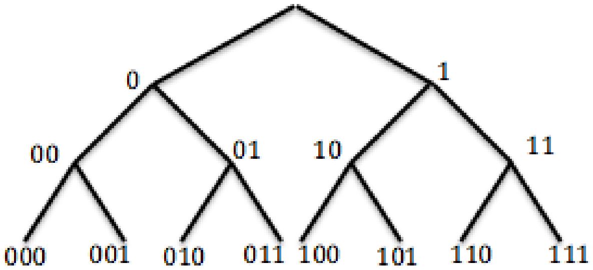

Figure 2. The top portion of the infinite binary tree $\mathcal{T}_2$ with some of its nodes labeled.

Definition 2.11. An *infinite path* in $\mathcal{T}_2$ is an infinite string of 0s and 1s. *Ends($\mathcal{T}_2$)* is the collection of all such infinite paths.

Ends($\mathcal{T}_2$) is a metric space with *metric* d: $Ends(\mathcal{T}_2) \times Ends(\mathcal{T}_2) \to \mathbb{R}$ defined by $d(\zeta_1, \zeta_2) = e^{-l}$, where $\zeta_1, \zeta_2 \in Ends(\mathcal{T}_2)$ and l is the length of the prefix shared between ζ_1 and ζ_2.

Let $S = \{\zeta \mid d(\zeta, w) \leq r \text{ with } \zeta, w \in Ends(\mathcal{T}_2)\}$. Then $S = B_r(w)$ is the *metric ball around w with radius r.*

Thompson's Group V acts as self-homeomorphisms of the Cantor set, and each element of V can be represented by a binary tree pair. Furthermore, the leaves of the trees can be represented in binary code where a branch to the left is denoted by "0" and a branch to the right by "1."

Using the notation established in [2], V is generated by a set of maps A, B, C and π_0 from one binary tree to another. We define the generators of V by the prefix changes they represent, which are equivalent to the tree diagrams in figure 3.

$$\begin{array}{cccc}
0* \mapsto 00* & 0* \mapsto 0* & 0* \mapsto 1* & 0* \mapsto 10* \\
10* \mapsto 01* & 10* \mapsto 100* & 10* \mapsto 0* & 10* \mapsto 0* \\
11* \mapsto 1* & 110* \mapsto 101* & 11* \mapsto 10* & 11* \mapsto 11* \\
 & 111* \mapsto 11* & & \\
A & B & C & \pi_0
\end{array}$$

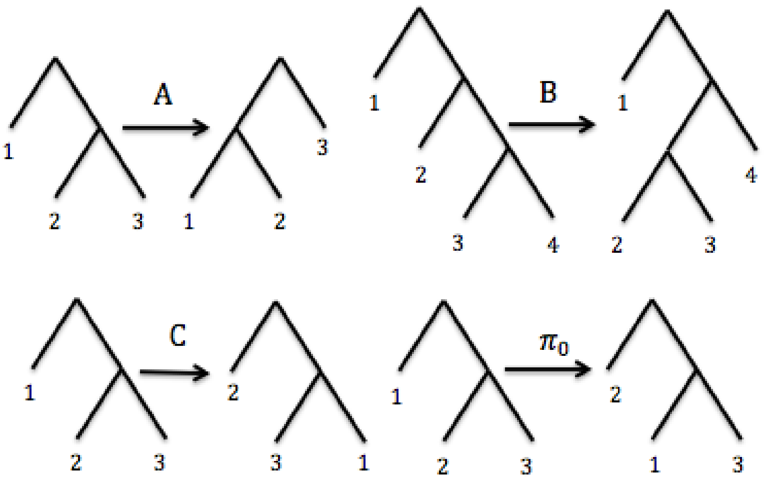

FIGURE 3. Elements A, B, C, and π_0 of V represented as tree pairs.

2.3. Generalized Thompson Groups $V_{(G,\theta)}$.

Definition 2.12. [9] The *forest monoid*, $\mathcal{F}$ is a sequence of rooted trees $(T_i)_{i\in\mathbb{N}}$ where all but finitely many are trivial. For two elements $E_1, E_2 \in \mathcal{F}$, their product, E_1E_2, is obtained by attaching the ith leaf of E_1 with the ith root of E_2.

Given a finite group, G, define

$$H = S_\infty \ltimes_\phi (\oplus_{i=1}^{\infty} G)$$

where $\phi : S_\infty \to Aut(\oplus_{i=1}^{\infty} G)$ by $\phi(\sigma)(g_1, ..., g_k, ...) = (g_{\sigma^{-1}(1)}, ...g_{\sigma^{-1}(k)}, ...)$ for $\sigma \in S_\infty$.

Multiplication of group elements $(\sigma_1, (g_1, ..., g_k, ...)), (\sigma_2, (g'_1, ..., g'_k, ...)) \in H$ works as follows:

$$\begin{aligned}
(\sigma_1, (g_1, ..., g_k, ...)(\sigma_2, (g'_1, ..., g'_k, ...)) &= (\sigma_1\sigma_2, \phi(\sigma_2)(g_1, ..., g_k, ...)(g'_1, ..., g'_k)) \\
&= (\sigma_1\sigma_2, (g_{\sigma_2^{-1}(1)}g'_1, ..., g_{\sigma_2^{-1}(k)}g'_k, ...))
\end{aligned}$$

Our group $V_{(G,\theta)}$ arises from the cloning system construction in [9]. A cloning system consists of a group H, a homomorphism $\rho : H \to S_\omega$, and a cloning map κ_k. A cloning system with these three elements that satisfies conditions $CS1, CS2, CS3$ of Proposition

2.7 [9] defines a BZS product, $\mathcal{F} \bowtie H$. We do not go into detail on the BZS product; for a full discussion see [9].

Elements of $V_{(G,\theta)}$ are ordered pairs of elements from a subgroup of the BZS product defined by the following cloning system:

$$H = S_\infty \ltimes_\phi (\oplus_{i=1}^{\infty} G),$$

$$\rho(\sigma, (g_1, ..., g_k)) = \sigma,$$

and κ_k which acts on the right by:

$$(\sigma, (g_1, ..., g_k, ...))\kappa_k = (\sigma\zeta, (g_1, ..., g_k, \theta(g_k), ...))$$

where ζ is the cloning map for the symmetric group defined in Example 2.9 .[9] and θ is an arbitrary homomorphism from $G \to G$.

We can think of elements of $V_{(G,\theta)}$ as equivalence classes of tree pairs much like Thompson's group V. The difference is that tree pairs in $V_{(G,\theta)}$ have group elements from G attached to their leaves. A tree pair (a, b) can be modified within its equivalence class by adding canceling carets or canceling group elements.

Canceling carets are added to corresponding leaves of the domain and range trees just as they would be in Thompson's group V unless there is a group element g on that leaf, in which case we put a g on the left branch of the new caret and $\theta(g)$ on the right branch. Canceling group elements are added to corresponding leaves of the domain and range trees and are combined with any group element already on the leaves. We can multiply two elements $(a, b), (c, d) \in V_{(G,\theta)}$, by choosing equivalent tree pairs (a', b') and (c', d') where $b' = c'$. Then $(a, b)(c, d) = (a', b')(c', d') = (a', d')$.

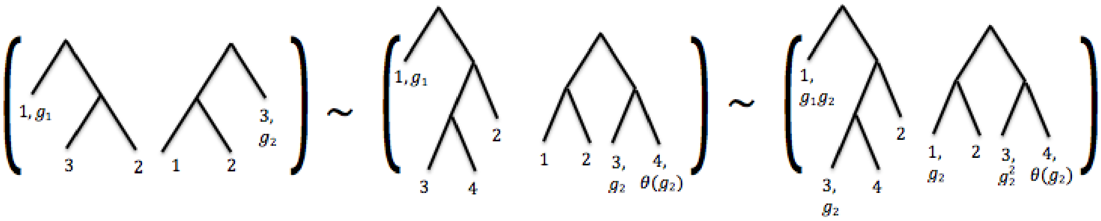

FIGURE 4. Three equivalent tree pairs, the second is obtained by adding a canceling caret and the third by canceling group elements.

Note that any element $(a, b) \in V_{(G,\theta)}$ can be expressed with no group elements in the domain tree by adding canceling group elements.

Let $\{g_1, ..., g_n\}$ be a generating set for G, then $\{A, B, C, \pi_0\} \cup \{g_{ja}, g_{jb}, g_{jc}, g_{jd}, g_{je} \mid 1 \leq j \leq n\}$ is a generating set for $V_{G,\theta}$, where $g_{ja}, g_{jb}, g_{jc}, g_{jd}, g_{je}$ are defined as in Figure 6.

Remark 2.13. When $\theta = id_G$, the identity homomorphism, $V_{(G,\theta)}$ embeds in V. Consider $V_{(G,\theta)}$ where θ is the identity homomorphism and $G = \{g_1, ..., g_n\}$ and $(a, b) \in V_{(G,\theta)}$. We choose a partition $W = \{w_1*, ..., w_n*\}$ where w_i will be the prefix associated with group element g_i. Consider the jth leaf of the tree pair, $b_j* \to a_j g_j*$. If $g_i g_j = g_k$, then $w_i b_j*$ is mapped to $w_k b_j*$. However, when $\theta \neq id_g$, the same method of embedding fails.

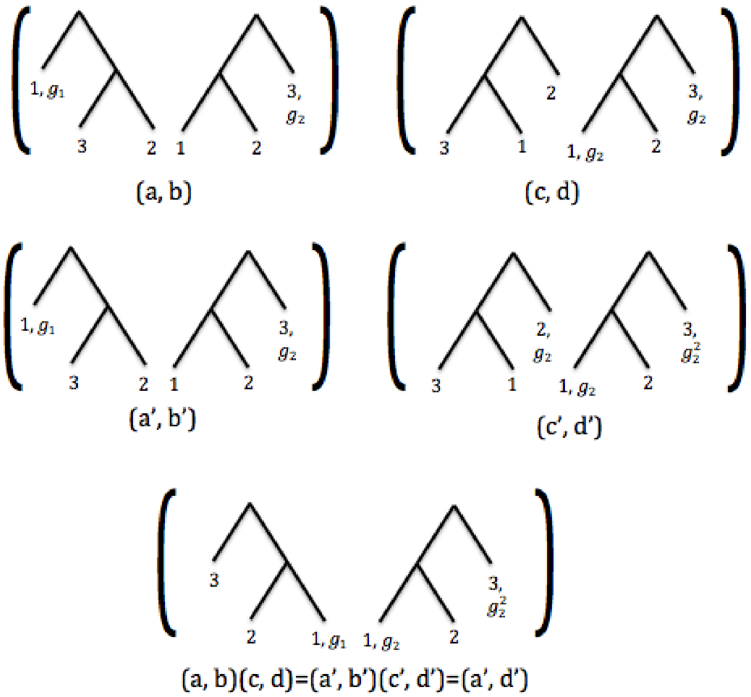

FIGURE 5. Multiplication of two group elements (a,b) and (c,d).

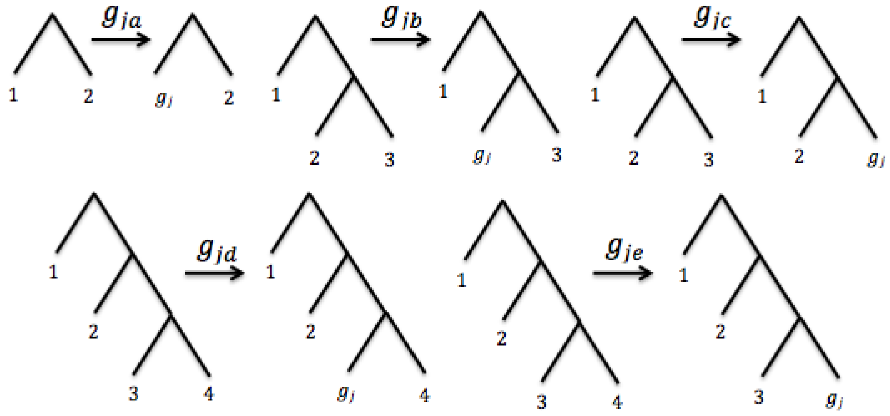

FIGURE 6. generators.

3. Demonstrative Groups

Definition 3.1. [1] Suppose a group G acts by homeomorphisms on a topological space X. For a group $H \leq G$, the action of H in G is *demonstrative* if and only if there exists an open set $U \subset X$ such that for all $h_1, h_2 \in G$, $h_1 U \cap h_2 U \neq \emptyset$ if and only if $h_1 = h_2$. The set U is called a *demonstration set.*

For this discussion, let $G = V$ and $X = Ends(\mathcal{T}_2)$. In this case, if U is a metric ball (i.e. $U = w*$ for some $w* \subseteq Ends(\mathcal{T}_2)$), then we refer to U as a *demonstration node.*

Definition 3.2. Let $H \leq G$ and let H act on a topological space S. We define $G \times_H S$ to be $\{(g, s) : g \in G, s \in S\}/\sim$, where $(gh, s) \sim (g, h \cdot s)$.

The *induced action* of G on S is $* : G \times (G \times_H S) \to G \times_H S$ defined by $g_1 * (g_2, s) = (g_1g_2, s)$.

Theorem 3.3. *If $H \leq G$ where $[G : H] = m$, for some $m \in \mathbb{N}$, then G embeds in V. Moreover, if H embeds as a demonstrative subgroup in V, then G embeds as a demonstrative subgroup of V.*

Proof. Assume $H \leq G$. Choose a left transversal $T = \{t_1, t_2, ..., t_m\}$ for H in G with $t_1 = 1$. We can induce an action of G on $G \times_H Ends(\mathcal{T}_2)$ by:

$$g \cdot (t_i, x) = (gt_i, x), \text{ for } x \in Ends(\mathcal{T}_2)$$

We know we can write gt_i as t_jh for unique $t_j \in T$ and $h \in H$. So,

$$(gt_i, x) = (t_jh, x) = (t_j, h \cdot x)$$

Now, we can embed $G \times_H Ends(\mathcal{T}_2) \hookrightarrow Ends(\mathcal{T}_2)$ by choosing a partition of $Ends(\mathcal{T}_2)$, $W = \{w_1*, w_2*, \ldots, w_m*\}$ and an injective function $\phi : T \to W$ by $\phi(t_i) = w_i*$. Now define $\Phi : G \times_H Ends(\mathcal{T}_2) \to Ends(\mathcal{T}_2)$ by $\Phi((t_i, x)) = \phi(t_i)x = w_ix$.

It can be easily checked that

$$g \cdot (w_ix) = g \cdot \Phi(t_i, x) = \Phi(gt_i, x) = \Phi(t_j, h \cdot x) = w_jh(x)$$

is a group action of G on $Ends(\mathcal{T}_2)$. Additionally, because elements of H act as elements of V, so do elements of G. Therefore, G embeds in V.

Now, assume H has a demonstrative embedding in V with demonstration node $a_1a_2...a_n*$ for $a_i \in \{0, 1\}$.

We will show that $w_ia_1 \ldots a_n*$ is a demonstration node for G. We compute the action of each of $g, g' \in G$ on $w_ia_1 \ldots a_n*$:

$$g \cdot w_ia_1 \ldots a_n* = w_jh(a_1 \ldots a_n*)$$
$$g' \cdot w_ia_1 \ldots a_n* = w_kh'(a_1 \ldots a_n*)$$

Since $w_1*, \ldots, w_m*$ partition $Ends(\mathcal{T}_2)$, if $w_j * \cap w_k* \neq \emptyset$ then $j = k$. Moreover, since H is a demonstrative subgroup of G with demonstration node $a_1a_2 \ldots a_n*$, $h(a_1 \ldots a_n*) \cap h'(a_1 \ldots a_n*) \neq \emptyset$ if and only if $h = h'$. Thus, $w_jh(a_1 \ldots a_n*) \cap w_kh'(a_1 \ldots a_n*) \neq \emptyset$ if and only if $j = k$ and $h = h'$, in other words, if and only if $g = g'$. Thus, G is demonstrative in V with demonstration node $w_ia_1 \ldots a_n*$ for any $i \in \{1, ..., m\}$. □

We will use the following statement of the Ping Pong Lemma from [8].

Lemma 3.4. *([8] ,Proposition 12.4) Let G be a group generated by two of its subgroups G_1 and G_2 where $G_1 and G_2$ are such that $G_1 \cap G_2 = H$ and $H \subsetneq G_1, G_2$ where H does not have index 2 in both G_1 and G_2. Let G act on a set Ω where Ω_1 and Ω_2 are disjoint non-empty subsets of Ω. If*

(1) $\Omega_1(G_1 - H) \subset \Omega_2, \quad \Omega_2(G_2 - H) \subset \Omega_1$
(2) $\Omega_1 H \subset \Omega_1, \quad \Omega_2 H \subset \Omega_2.$

then $G = G_1 *_H G_2$, the free product of G_1 and G_2 with H amalgamated.

Theorem 3.5. *If H is a demonstrative subgroup of V and G_1, G_2 are such that $[G_1 : H] = m$ and $[G_2 : H] = n$, then there exists an embedding of $G_1 *_H G_2$, where each of G_1, G_2, H is demonstrative.*

Proof. Since $[G_1 : H], [G_2 : H] < \infty$ we can construct demonstrative embeddings of both G_1 and G_2 in V, by following the process given in Theorem 3.3 with the following choice of partitions:

$$W_1 = \{w_1*, w_2*, ..., w_m*\}$$

and

$$W_2 = \{\hat{w}_1*, \hat{w}_2, ..., \hat{w}_n*\}.$$

Where $w_1* = 0*$ and $w_2*, ..., w_m*$ partition $1*$ and $\hat{w}_1* = 1*$ and $\hat{w}_2*, ..., \hat{w}_n*$ partition $0*$ Now using Lemma 3.4, with $\Omega_1 = 0*$ and $\Omega_2 = 1*$.

For arbitrary $h \in H$,

$$h \cdot \Omega_1 = h \cdot 0* = w_1 h(*) = 0* \subseteq \Omega_1$$

and

$$h \cdot \Omega_2 = h \cdot 1* = \hat{w}_1 h(*) = 1* \subseteq \Omega_2.$$

Let $f \in (G_1 - H)$

$$f \cdot \Omega_1 = f \cdot 0* = w_j h_k(*) = w_j * .$$

Since $f \notin H, j \neq 1$ therefore $w_j \neq 0$, so $w_j* \subseteq 1* = \Omega_2$.

Let $f' \in (G_2 - H)$

$$f \cdot \Omega_2 = f \cdot 1* = \hat{w}_j h_k(*) = \hat{w}_j * .$$

Since $f \notin H, j \neq 1$,therefore $\hat{w}_j \neq 1$, so $\hat{w}_j* \subseteq 0* = \Omega_1$.

Thus, by Lemma 3.4, this is an embedding of $G_1 *_H G_2$. □

Example 3.6. Consider the braid group on three strands, B_3. We know $B_3 \cong G_1 *_H G_2$ where $G_1 = G_2 = H = \mathbb{Z}$, $[G_1 : H] = 2$, $[G_2 : H] = 3$, $G_1 \cong \langle g_1 \rangle$, and $G_2 \cong \langle g_2 \rangle$.

The following is a demonstrative embedding of H in V with demonstration node $10*$:

$$\begin{aligned} h : 0* &\mapsto 00* \\ 10* &\mapsto 01* \\ 11* &\mapsto 1* \end{aligned}$$

Use the left transversals $T_1 = \{1, g_1\}$ and $T_2 = \{1, g_2, (g_2)^2\}$.

We now compute the induced actions of G_1 and G_2 on $G_1 \times_H Ends(\mathcal{T}_2)$ and $G_2 \times_H Ends(\mathcal{T}_2)$, respectively. For any $x \in Ends(\mathcal{T}_2)$, we have

$$\begin{aligned} g_1(1, x) &= (g_1, x) \\ g_1(g_1, x) &= (1, h(x)), \end{aligned}$$

and

$$\begin{aligned} g_2(1, x) &= (g_2, x) \\ g_2(g_2, x) &= ((g_2)^2, x) \\ g_2((g_2)^2, x) &= (1, h(x)) \end{aligned}$$

Now, we define two partitions $W_1 = \{0*, 1*\}$ and $W_2 = \{00*, 01*, 1*\}$ and functions ϕ_1 and ϕ_2 from the two transversals to these partitions where $\phi_1 : T_1 \to W_1$ by taking 1 to $0*$ and g_1 to $1*$ and

$$\begin{aligned} \phi_2 : T_2 &\to W_2 \\ 1 &\mapsto 1* \\ g_2 &\mapsto 00* \\ (g_2)^2 &\mapsto 01*. \end{aligned}$$

Define $\Phi_1 : G_1 \times_H Ends(\mathcal{T}_2) \to Ends(\mathcal{T}_2)$ by $\Phi_1((t_i, x)) = \phi_1(t_i)x = w_i x$ and $\Phi_2 : G_2 \times_H Ends(\mathcal{T}_2) \to Ends(\mathcal{T}_2)$ by $\Phi_2((\hat{t_i}, x)) = \phi_2(\hat{t_i})x = \hat{w}_i x$. These identifications determine an action of G_1 and G_2 on $Ends(\mathcal{T}_2)$. Combining this with the map for H we get $\hat{g}_1$ and $\hat{g}_2$ as generating elements for the embedding of $G_1 *_H G_2$ in V.

$$\begin{aligned} \hat{g}_1 : 0* &\mapsto 1* \\ 10* &\mapsto 000* \\ 110* &\mapsto 001* \\ 111* &\mapsto 01* \end{aligned}$$

and

$$\begin{aligned} \hat{g}_2 : 00* &\mapsto 01* \\ 010* &\mapsto 100* \\ 0110* &\mapsto 101* \\ 0111* &\mapsto 11* \\ 1* &\mapsto 00* \end{aligned}$$

These are illustrated by the tree pairs in Figure 7.

We know from Theorem 3.5 that $\Psi : G_1 *_H G_2 \to V$ by $\Psi(g_1) = \hat{g}_1, \Psi(g_2) = \hat{g}_2$ will give an embedding of $G_1 *_H G_2$ in V.

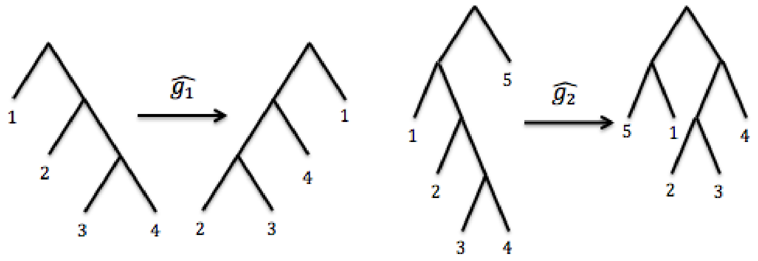

FIGURE 7. Tree pairs representing $\hat{g}_1$ and $\hat{g}_2$.

4. Main Result

Lemma 4.1. *Let $\Sigma = \{A, B, C, \pi_0, g_{ij}\}$. Define $\Phi : \Sigma^* \to V$ by the homomorphism*

$$\begin{aligned} \Phi : A &\mapsto A \\ B &\mapsto B \\ C &\mapsto C \\ \pi_0 &\mapsto \pi_0 \\ g_{ij} &\mapsto 1_V \end{aligned}$$

If $w = b_1 \cdots b_m \in \Sigma^*$ satisfies $\Phi(w) \neq 1$, then there is some cyclic permutation $b_j \cdots b_m b_1 \cdots b_{j-1}$ and some $B \in \{a_1a_2a_3* : a_i \in 0, 1\}$ that satisfy $b_j \cdots b_m b_1 \cdots b_{j-1}(B) \cap B \neq \emptyset$.

Proof. Consider $w = b_1 \cdots b_m \in \Sigma^*$. If $\Phi(w) \neq 1$, then $\Phi(w) \in CoWP(V)$. We know $\{a_1a_2a_3* : a_i \in 0, 1\}$ is a test partition for V [3]. So, there is some cyclic permutation $\Phi(w)'$ of $\Phi(w)$ and some $B \in \{a_1a_2a_3* : a_i \in \{0, 1\}\}$ such that $\Phi(w)'(B) \cap B \neq \emptyset$. Since Φ takes all the generators of group elements to 1, Φ will preserve the shape of any tree pair. Thus, if $\Phi(w)'$ is such that $\Phi(w)'(B) \cap B \neq \emptyset$, then $w'(B) \cap B \neq \emptyset$ where w' is some cyclic shift $b_j \cdots b_m b_1 \cdots b_{j-1}$ of w. □

Definition 4.2. Let $\mathcal{L}$ be a language. The *cyclic shift* of $\mathcal{L}$, denoted $\mathcal{L}^\circ$, is

$$\mathcal{L}^\circ = \{w_2w_1 \in \Sigma^* \mid w_1w_2 \in \mathcal{L}, w_1, w_2 \in \Sigma^*\}.$$

A *cyclic permutation* w' of a word $w = w_1w_2$ is $w' = w_2w_1$. Note that context-free languages are closed under cyclic shifts [7].

Theorem 4.3. *The group $V_{(G,\theta)}$ is co$\mathcal{CF}$.*

Proof. We design an automaton P to accept by empty stack, with stack alphabet $\Gamma = \{0, 1, g \mid g \in G \backslash \{1_G\}\}$ and input alphabet $\Sigma = \{A, B, C, \pi_0, g_{ij}\}$, where each g_{ij} is a generator of G.

We define the sets $\mathcal{L}_{B_i}$ and $\mathcal{L}_G$:

$\mathcal{L}_{B_i} = \{w \in \Sigma^* \mid w(B_i) \cap B_j \neq \emptyset$ for some $j \neq i\}$.

$\mathcal{L}_G = \{w \in \Sigma^* \mid w(B_i) \cap B_i \neq \emptyset$ and w expressed as a reduced tree pair has some nontrivial group element on the range tree$\}$.

We design P such that $\mathcal{L}_\mathcal{P} = (\bigcup_{i=1}^{8} \mathcal{L}_{B_i}) \cup \mathcal{L}_G$. Figure 8 depicts the outline of the automaton. Note that unlabeled arrows represent $(\epsilon, \epsilon, \epsilon)$ transitions.

To start off, P enters a non-deterministic loading phase where a finite string of 1's and 0's is entered onto the stack by taking either $(\epsilon, \epsilon, 0)$ or $(\epsilon, \epsilon, 1)$ transitions.

The loading phase ends by taking a transition $(\epsilon, \epsilon, B_k)$ where

$$B_k \in \{000, 001, 010, 100, 011, 110, 101, 111\},$$

i.e. B_k for $k \in \{1, ..., 8\}$ is one of the 8 metric balls in the test partition.

Next, P enters the reading phase which has a single state for each generator where the first element on the input tape is read and that generator is applied to the appropriate prefix at the top of the stack. For example, the reading phase for A would read and delete a 0, and then add 10; the reading phase for g_{2b} would read and delete 10, and then add $10g_2$.

After the reading phase, P enters the clean-up state, which consists of the pushing and combining of stack elements. This phase allows P to "clean-up" the stack so that there

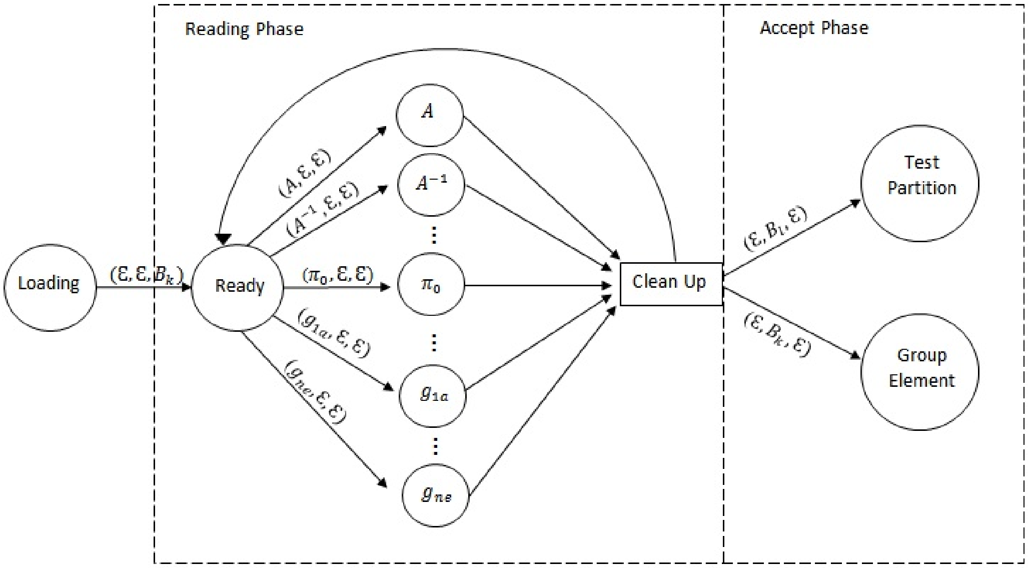

FIGURE 8. The automaton for $V_{(G,\theta)}$.

are at least three 0's and 1's for the next element on the input tape to successfully act on the stack. First, P "pushes" elements within the first three spots to the fourth spot on the stack. For example, as shown in Figure 4, one set of transitions will read and delete $0g0$ or $0g1$ from the stack and then add $00g$ or $01(\theta(g))$, respectively, to the stack for all $g \in G\backslash\{1_G\}$. Similar transitions can be followed if the group element is preceded by the prefix 0, 1, 01, 10, 00, or 11.

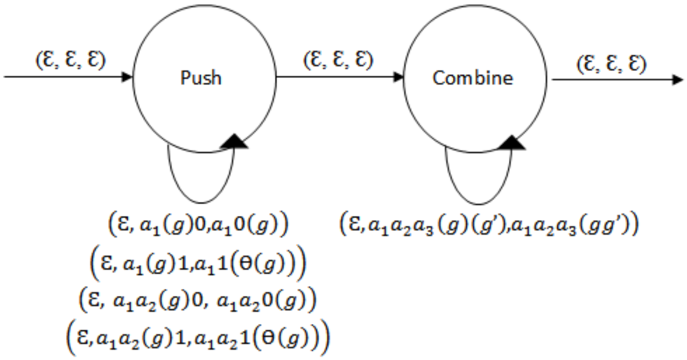
(ε, ε, ε)
Push
(ε, ε, ε)
Combine
(ε, ε, ε)
(ε, $a_1(g)0, a_1 0(g)$)
(ε, $a_1(g)1, a_1 1(\theta(g))$)
(ε, $a_1 a_2(g)0$, $a_1 a_2 0(g)$)
(ε, $a_1 a_2(g)1, a_1 a_2 1(\theta(g))$)
(ε, $a_1 a_2 a_3 (g)(g'), a_1 a_2 a_3 (gg')$)

FIGURE 9. The clean-up phase. Let $a_1, a_2, a_3 \in \{0, 1\}$ and $g, g' \in G\backslash\{I_G\}$.

Next, P enters the combining state where group elements are rewritten as a single element of the group (in accordance with the group operation). For example, one collection of edges is able to read and delete $010(g)(g')$ and add $010(gg')$ for all $g, g' \in G\backslash\{1_G\}$. Note that if

$gg' = 1_G$, then the path reads and deletes $010(g)(g')$, and writes 010. Similar paths exist when combining any two group elements preceded by any three-digit prefix.

After exiting the clean-up phase, P reads the next element on the input tape and repeats the process of the reading phase. When the input tape is empty and P has gone through the reading and clean-up phases, P then moves onto one of two accept states.

If the word took one metric ball B_k in the test partition to some other metric ball B_l, then we enter the *test partition accept state*. At this point we "unload" all of the 0's and 1's and group elements off the stack with paths (ϵ, x, ϵ) for $x \in \{0, 1, g \mid g \in G\backslash\{1_G\}\}$. Once every stack element has been deleted, P takes the path $(\epsilon, \#, \epsilon)$ which deletes the start symbol and accepts the word. So the language accepted by the test partition accept state is $(\bigcup_{i=1}^{8} \mathcal{L}_{B_i})$.

If the word does not displace metric ball B_k, then we enter the *group element accept state*. Here, we delete every 0 and 1 on the stack until P arrives at a group element. The group element is then"pushed" further down the stack, and the 0 or 1 it pushes past is deleted. For example, one path is $(\epsilon, g0, g)$ for $g \in G\backslash\{1_G\}$. If the group element is followed on the stack by a second group element, then they are "combined" in a manner mimicking the previously described combining portion of the clean-up phase. This is repeated until there are no 0's or 1's left on the stack. At this point, if there is still a group element remaining on the stack followed by the start symbol, then they are both deleted and thus the word is accepted. However, if there is no group element on the stack then the start symbol cannot be deleted so the word is not accepted. So the language accepted by the group element accept state is $\mathcal{L}_G$.

Thus, $\mathcal{L}_P = (\bigcup_{i=1}^{8} \mathcal{L}_{B_i}) \cup \mathcal{L}_G$.

Claim: CoWP($V_{(G,\theta)}$)= $(\mathcal{L}_P)^\circ$.

Let $w \in \mathcal{L}_P, w \in \mathcal{L}_G$ or $w \in \mathcal{L}_{B_i}$. If $w \in \mathcal{L}_G$, then, for some $a_1.....a_n*$ loaded onto P, $w(a_1.....a_n*)$ contains group elements, so $w(a_1.....a_n*) \neq a_1.....a_n*$. Thus, $w \in$ CoWP($V_{(G,\theta)}$). If $w \in \mathcal{L}_{B_i}$ for some $1 \leq i \leq 8$, then $w(B_i) \cap B_j$ for $j \neq i$, so $w \in$ CoWP($V_{(G,\theta)}$. Therefore, $(\mathcal{L}_P) \subseteq$ CoWP($V_{(G,\theta)}$). The CoWP of a group is closed under cyclic shift, and thus $(\mathcal{L}_P)^\circ \subseteq$ CoWP($V_{(G,\theta)}$).

Let $w \in$CoWP($V_{(G,\theta)}$). We will use the surjective homomorphism Φ from Lemma 4.1.

If $w \notin ker(\Phi)$, then$\Phi(w) \neq 1_V$. By Lemma 4.1, there is some cyclic permutation w' and some B_i such that $w'(B_i) \cap B_i \neq \emptyset$, i.e. $w' \in \mathcal{L}_{B_i}$. Therefore, $w \in (\mathcal{L}_{B_i})^\circ \subseteq (\mathcal{L}_P)^\circ$.

If $w \in ker(\Phi)\backslash 1_{V_{(G,\theta)}}$, then $\Phi(w) = 1_V$, which implies that w (as a reduced tree pair) does not change any prefixes; it simply adds group elements. Therefore, $w \in \mathcal{L}_G \subseteq (\mathcal{L}_P)^\circ$.

We now have that CoWP($V_{(G,\theta)}$)= $(\mathcal{L}_P)^\circ$. A language is context-free if and only if it's shuffle is also context free. Since $\mathcal{L}_P$ is context-free, CoWP($V_{(G,\theta)}$) is context-free, and $V_{(G,\theta)}$ is co$\mathcal{CF}$.

□

Remark 4.4. We attempted a similar method of proof with the group generated by Thompson's group V and the Grigorchuk group G, $R = \langle V, G\rangle$. Like $V_{(G,\theta)}$, elements of R can be thought of as Thompson's group V elements with Grigorchuk group elements, g attached to the leaves. However, the Grigorchuk group elements continue to act on the tree whereas the group elements in $V_{(G,\theta)}$ just sit at the end of the branches. This became a problem because it is impossible to complete the calculation of the action of g on any finite test string loaded onto an automaton. We also ran into problems because the test partitions argument used in Theorem 4.3. and for Finite Similarty Structure Groups [6] fails.

Acknowledgments. We would like to first thank our research advisor, Dr. Dan Farley, for aiding us in our research endeavors and providing us with the knowledge needed for us to be successful. We also thank our graduate assistant, Hannah Hoganson. Her constant support and aide enabled us to achieve our goals for this program. We thank SUMSRI for giving us the opportunity to participate in undergraduate research in mathematics, and we thank the faculty and staff for making this program possible. In addition, we would like to express our gratitude to Miami University for providing funding, housing, and a place to research. Finally, we thank the National Science Foundation for funding SUMSRI and making our research possible.

References

[1] Collin Bleak and Olga Salazar-Díaz. Free products in R. Thompson's group V. *Trans. Amer. Math. Soc.*, 365(11):5967–5997, 2013.

[2] J.W Cannon, W.J Floyd, and W. R Parry. Introductory notes on richard thompson's groups. *L Enseignement Mathematique*, 42:215–256, 1996.

[3] Daniel Farley. Local similarity groups with context-free co-word problem. Unpublished.

[4] Daniel Farley and Bruce Hughes. Braided diagram groups and local similarity groups. 2013.

[5] Derek F. Holt, Sarah Rees, Claas E. Röver, and Richard M. Thomas. Groups with context-free co-word problem. *J. London Math. Soc. (2)*, 71(3):643–657, 2005.

[6] Bruce Hughes. Local similarities and the Haagerup property. *Groups Geom. Dyn.*, 3(2):299–315, 2009. With an appendix by Daniel S. Farley.

[7] J. Lehnert and P. Schweitzer. The co-word problem for the Higman-Thompson group is context-free. *Bull. Lond. Math. Soc.*, 39(2):235–241, 2007.

[8] Roger C. Lyndon and Paul E. Schupp. *Combinatorial Group Theory.* Springer Science and Business Media, 2001.

[9] Stefan Witzel and Mathew C.B. Zaremsky. Thompson groups for systems of groups, and their finiteness properties. *arXiv:1405.5491, 48 pages*, 2014.