SOME EMBEDDINGS BETWEEN SYMMETRIC R. THOMPSON GROUPS

JULIO AROCA AND COLLIN BLEAK

Abstract. Let $m \leq n \in \mathbb{N}$, and $G \leq \mathrm{Sym}(m)$ and $H \leq \mathrm{Sym}(n)$. In this article we find conditions enabling embeddings between the symmetric R. Thompson groups $V_m(G)$ and $V_n(H)$. When $n \equiv 1 \mod (m-1)$, and under some other technical conditions, we find an embedding of $V_n(H)$ into $V_m(G)$ via topological conjugation. With the same modular condition we also generalise a purely algebraic construction of Birget from 2019 to find a group $H \leq \mathrm{Sym}(n)$ and an embedding of $V_m(G)$ into $V_n(H)$.

1. Introduction

This article concerns embeddability conditions for pairs of groups from the family of *symmetric R. Thompson groups* $\{V_m(G)\}$. The group $V_m(G)$ is the group $V_m(G) = \langle V_m \cup G \rangle$, where $V_m \leq \mathrm{Aut}(\mathfrak{C}_m)$ is the Higman-Thompson group denoted $G_{m,1}$ by Higman in [9], acting on the Cantor space $\mathfrak{C}_m := \{0, 1, \ldots, m-1\}^\omega$, while G is a particular faithful representation of a finite group $\widetilde{G} \leq \mathrm{Sym}(m)$ in $\mathrm{Aut}(\mathfrak{C}_m)$.

The groups $\{V_m(G)\}$ have developed as groups of interest for a variety of reasons. Firstly, they were singled out as natural groups of interest in [14] and [15], and they arise naturally as a fundamental subfamily of Hughes' $\mathcal{FSS}$ groups [11]. The paper [2] shows that for $n \geq 2$, $V_m \cong V_m(G)$ if and only if $\widetilde{G}$ is semiregular (the nontrivial elements of $\widetilde{G}$ have no fixed points), and also, that for $m > 3$ there exists $\widetilde{G}, \widetilde{H} \in \mathrm{Sym}(m)$ with $\widetilde{G} \cong \widetilde{H}$ but where the induced groups $V_m(G)$ and $V_m(H)$ are not isomorphic (the orbit structure of the actions of the elements of the groups $\widetilde{G}$ and $\widetilde{H}$ impacts the isomorphism types of the groups $V_m(G)$ and $V_m(H)$). In another direction, in [7] Farley shows the symmetric R. Thompson groups are CoCF groups (see [10] for the definition of CoCF groups). Thus, if one can show that some group in the family $\{V_m(G)\}$ fails to embed in $V = V_2$, then Lehnert's conjecture will be shown to be false (see [12, 13, 4]).

We investigate conditions on $m \leq n$, $G \leq \mathrm{Sym}(m)$, and $H \leq \mathrm{Sym}(n)$ that guarantee the existence of embeddings between the groups $V_m(G)$ and $V_n(H)$ (we now drop the "tilde" notation on the groups G and H when thinking of them as subgroups of $\mathrm{Sym}(m)$ and $\mathrm{Sym}(n)$, respectively). Thus, this note can be thought of as a continuation of the investigations in [2], and is partly inspired by the work of Birget (in [1], he gives a method to embed V_2 into V_m for $2 \leq m$), and partly by considering some of the questions alluded to in the previous paragraph. In

The first author wants to acknowledge financial support from the Spanish Ministry of Economy and Competitiveness, through the "Severo Ochoa Program for Centres of Excellence in R&D" (SEV-2015-0554). The second author wishes to acknowledge support from EPSRC grant EP/R032866/1 during the creation of this paper.

Date: August 10, 2021.

Key words and phrases. R. Thompson groups, embeddings, topological conjugacy, FSS Groups.

this context, our two embedding results depend on the direction of the embedding ($V_m(G) \rightarrowtail V_n(H)$ or $V_n(H) \rightarrowtail V_m(G)$), and our constructed embeddings require in both cases the Higman condition $n \equiv 1 \mod (m-1)$.

To our knowledge, the first case of a proof that all V_n embed into each other is by Nicolás Matte Bon [5, Coroll. 11.16], although it was almost a folklore statement that such embeddings existed. Before that, the only such embedding theorems that we know of in the literature were those of Higman from [9, Thm. 7.2] ($V_n \rightarrowtail V_m$ when m and n satisfy the Higman condition). We leverage Birget's construction [1] in this paper because of the utility of his *successor function*, which we are able to generalize effectively for embeddings in our enriched context ($V_m(G)$ and $V_n(H)$ as opposed to V_m and V_n).

The embedding $V_m(G) \rightarrowtail V_n(H)$ is algebraic in nature, and is the one which is inspired by the embeddings of Birget, while the embedding $V_n(H) \rightarrowtail V_m(G)$ uses a topological conjugacy by rational group elements (see [8]).

We can now state and discuss the main results of this chapter. For the topological embedding $V_n(H) \rightarrowtail V_m(G)$, we need to define a group $\mathcal{R}_G(\widetilde{S}) \leq \mathrm{Sym}(n)$ (where $G \leq \mathrm{Sym}(m)$ and $\widetilde{S}$ is a prefix code of $\mathfrak{C}_m$ of length n). The group $\mathcal{R}_G(\widetilde{S})$ is well defined if there is an n-element prefix code $\widetilde{S}$ in $\mathcal{A}_m^*$ which is preserved by the action of the iterated permutations of G on $\mathcal{A}_m^*$. In this case $\mathcal{R}_G(\widetilde{S})$ is the subgroup of $\mathrm{Sym}(n)$ induced from the action of G on the elements of $\widetilde{S}$ under a bijection to $\{1, 2, \ldots, n\}$.

Theorem 1.1 (Topological Embedding)**.** *Let $n, m \geq 2$ be natural numbers such that $m < n$, and let $G \leq \mathrm{Sym}(m)$, $H \leq \mathrm{Sym}(n)$. Suppose that:*

(1) *There exists a prefix code $\widetilde{S}$ of $\mathfrak{C}_m$ such that $|\widetilde{S}| = n$,*
(2) *the group $\mathcal{R}_G(\widetilde{S})$ is well defined, and*
(3) *$\mathcal{R}_G(\widetilde{S})$ and H are cyclically isomorphic.*

Then $V_n(H)$ embeds in $V_m(G)$.

Thus, we have the following general observation.

Observation 1.1. In practical terms, natural applications of Theorem 1.1 occur by choosing m and a prefix code that is closed under the action of iterated permutations from some $G \leq \mathrm{Sym}(m)$. Then, one immediately obtains a cyclically isomorphic group H in $\mathrm{Sym}(n)$.

Theorem 1.2 (Algebraic Embedding)**.** *Let $n, m \geq 2$ be natural numbers such that $n = k(m-1) + m$ for some $k \geq 1$, and let $G \leq \mathrm{Sym}(m)$. Let $H = G_{ext} \leq \mathrm{Sym}(n)$ the extended symmetric group of H, whose elements act as the elements of H on the first m elements, and act as the identity on the remaining $n - m$.*

Then $V_m(G)$ embeds in $V_n(H)$.

Thus, to achieve such an embedding, the main obstruction is that m and n satisfy the Higman Condition, i.e.:

Corollary 1.2. *Let $m \leq n$ so that $n \equiv m \mod (m-1)$, and let $G \leq \mathrm{Sym}(m)$. Then, there exists $H \leq \mathrm{Sym}(n)$ so that $V_m(G) \rightarrowtail V_n(H)$.*

As mentioned above, for given m and G semiregular, the paper [2] shows $V_m(G) \cong V_m$, while Higman's book [9] gives an embedding of V_m into R. Thompson's group $V = V_2$. The semiregularity condition above has to do with local groups of germs of the action of $V_m(G)$ on $\mathfrak{C}_m$ (see [2, 3]). Our topological embedding preserves the local groups of germs, while the algebraic embeddings (for G non-trivial) produce complicated local groups of germs. Thus, it is impossible to chain our families of embeddings together to get an embedding from $V_m(G)$ into V_2 when G is not semi-regular.

We therefore ask the following question:

Question 1.3. Does there exist $m \in \mathbb{N}$ and $G \in \mathrm{Sym}(m)$ so that G is not semiregular, but where there is an embedding from $V_m(G)$ into V_2?

Acknowledgements:
The authors would like to thank Javier Aramayona, Jim Belk, and Matthew G. Brin for numerous and enjoyable conversations around the material in this note. We also wish to thank an anonymous referee for helpful comments on exposition.

2. Symmetric Thompson's groups

In this section, we introduce symmetric Thompson's groups, giving an easy way to express its elements as tables (this corresponds to a generalised construction of Higman used by Scott and Röver in the creation of their extensions of V ([16, 17, 18, 15]).

2.1. **Tables.** For natural $n > 1$, let $\mathfrak{C}_n$ be the n-adic Cantor set, which is constructed inductively as follows: $\mathfrak{C}_n^1$ corresponds to first subdividing $\mathfrak{C}_n^0 = [0,1]$ into $2n-1$ closed intervals of equal length (so, sharing endpoints with neighbours), numbered $1, \ldots, 2n-1$ from left to right, and then taking the collection of odd-numbered sub-intervals. Next, $\mathfrak{C}_n^2$ is obtained from $\mathfrak{C}_n^1$ by applying the same procedure to each of the intervals forming $\mathfrak{C}_n^1$, and so on. Then, C_n is the limit of this process, so that

$$\mathfrak{C}_n = \cap_i \mathfrak{C}_n^i.$$

Now, let $\mathcal{A}_n = \{0, \ldots, n-1\}$ and give it the discrete topology. It is easy to build a direct homemorphism from the space $\mathcal{A}_n^{\mathbb{N}}$ equipped with the product topology to $\mathfrak{C}_n$, so every element $\zeta \in \mathfrak{C}_n$ can be expressed as an infinite word $\zeta = w_1 w_2 \ldots$, where $w_i \in \mathcal{A}_n$. It is a classical result of Brouwer from [6] that all of the spaces in the set $\{\mathfrak{C}_n\}$ are abstractly homeomorphic to each other.

We denote by $\mathcal{A}_n^*$ the set of finite words in $\mathcal{A}_n$. The empty word ε is also in $\mathcal{A}_n^*$.

Definition 2.1 (Concatenation)**.** Let $u = u_1 u_2 \ldots u_k, u_i \in \mathcal{A}_n$ be a finite word and $v \in \mathcal{A}_n^* \cup \mathcal{A}_n^{\mathbb{N}}$ with $v = v_1 v_2 \ldots$ (where for all valid indices i we have $v_i \in \mathcal{A}_n$) . The *concatenation* of u with v is the (finite or infinite) word:

$$u \| v = u_1 u_2 \ldots u_k v_1 v_2 \ldots.$$

With concatenation being a fundamental operation, we will often just write the concatenation of two strings without the formal concatenation operator, that is, we might write $u\|v$ as simply uv, reserving the formal use of "$\|$" for situations where we wish to stress that a concatenation is occuring.

Definition 2.2 (Prefix order)**.** Let $u \in \mathcal{A}_n^*$ and $v \in \mathcal{A}_n^* \cup \mathcal{A}_n^{\mathbb{N}}$. We say that u is a *prefix* of v ($u \leq_{pref} v$) if $v = u\|w$, for some $w \in \mathcal{A}_n^* \cup \mathcal{A}_n^{\mathbb{N}}$.

Note that this property is transitive for finite length words: If $u \leq_{pref} v$ and $v \leq_{pref} w$ then $u \leq_{pref} w$. In addition, $u \leq_{pref} u$, as $\varepsilon \in \mathcal{A}_n^*$. That is, $\leq_{pref}$ provides a partial order on $\mathcal{A}_n^*$.

Definition 2.3 (Prefix code)**.** Let S be a finite set of words in $\mathcal{A}_n^*$. Then S is an *prefix code* of $\mathfrak{C}_n$ if for every infinite word $\zeta \in \mathcal{A}_n^{\mathbb{N}}$ there exists one and only one word $s \in S$ such that $s \leq_{pref} \zeta$. (Specifically, a prefix code is a complete anti-chain for the partial order $\leq_{pref}$.)

For convenience, we will use the following notation: let $\sigma \in \mathrm{Sym}(n)$ be an element of the symmetric group of n elements. Given any word $\zeta = z_1z_2z_3\cdots \in \mathcal{A}_n^* \cup \mathcal{A}_n^{\mathbb{N}}$ we define $\sigma(\zeta) = \sigma(z_1)\sigma(z_2)\sigma(z_3)\cdots \in \mathcal{A}_n^* \cup \mathcal{A}_n^{\mathbb{N}}$. Let $\sigma_i \in H \leq \mathrm{Sym}(n)$. (Note that we are using left actions here, so if $\sigma, \tau \in \mathrm{Sym}(n)$ then the product $\tau\sigma$ means employ the permutation σ first, and then employ τ).

With the above notation, an element of $V_n(H)$ is a homeomorphism of $\mathfrak{C}_n$ that can be (non-uniquely) described by a *table* as follows:

$$v = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1 & \sigma_2 & \cdots & \sigma_k \\ q_1 & q_2 & \cdots & q_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix},$$

where $p_i, q_i \in \mathcal{A}_n^*$, $\sigma_i, \tau_i \in H$ and such that the sets $P = \{p_i\}_{i=1}^k$ and $Q = \{q_i\}_{i=1}^k$ are prefix codes of $\mathfrak{C}_n$. We say that $k \geq 1$ is the *length* of the table. The homeomorphism of $\mathfrak{C}_n$ induced can be defined as follows: for every infinite word ζ such that $p_i \leq_{pref} \zeta$, that is $\zeta = p_i \| u$ for some $u \in \mathcal{A}_n^{\mathbb{N}}$, we have

$$v : p_i \| \sigma_i(u) \to q_i \| \tau_i(u).$$

There are infinitely many tables which induce the same homeomorphism of $\mathfrak{C}_n$. We proceed to define the four basic moves we can perform on a table in order to obtain an equivalent one (the four basic moves naturally split as two essential sorts of moves, together with their inverse (or "near-inverse") moves).

The first basic move is *expansion*: for a given prefix code

$$P = \{p_1, \ldots, p_i, \ldots, p_k\},$$

we can consider

$$\widetilde{P} = \{p_1, \ldots, p_i0, \ldots, p_i(n-1), \ldots, p_k\}$$

by expanding the word p_i. This expansion not only occurs in P, as the image of p_i must be also expanded. So we have

$$\widetilde{Q} = \{q_1, \ldots, q_i0, \ldots, q_i(n-1), \ldots, q_k\}.$$

It is easy to see that both $\widetilde{P}$ and $\widetilde{Q}$ are also prefix codes. Then:

$$\begin{bmatrix} p_1 & \cdots & p_i & \cdots & p_k \\ \sigma_1 & \cdots & \sigma_i & \cdots & \sigma_k \\ q_1 & \cdots & q_i & \cdots & q_k \\ \tau_1 & \cdots & \tau_i & \cdots & \tau_k \end{bmatrix} \equiv \begin{bmatrix} p_1 & \cdots & p_i\sigma_i(0) & \cdots & p_i\sigma_i(n-1) & \cdots & p_k \\ \sigma_1 & \cdots & \sigma_i & \cdots & \sigma_i & \cdots & \sigma_k \\ q_1 & \cdots & q_i\tau_i(0) & \cdots & q_i\tau_i(n-1) & \cdots & q_k \\ \tau_1 & \cdots & \tau_i & \cdots & \tau_i & \cdots & \tau_k \end{bmatrix}.$$

One can always perform an expansion, but not all tables look like the result of an expansion. Naturally, the inverse of an expansion (when it is defined) is called a *reduction*.

The second move we can perform on a table is *pushing down* (resp. *pushing up*) the action of all σ_i such that $\sigma_i = Id$ for every $i \in \{1, \ldots, k\}$ (resp. $\tau_i = Id$ for every $i \in \{1, \ldots, k\}$):

$$\begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1 & \sigma_2 & \cdots & \sigma_k \\ q_1 & q_2 & \cdots & q_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix} \equiv \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ Id & Id & \cdots & Id \\ q_1 & q_2 & \cdots & q_k \\ \tau_1\sigma_1^{-1} & \tau_2\sigma_2^{-1} & \cdots & \tau_k\sigma_k^{-1} \end{bmatrix}$$

$$\equiv \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1\tau_1^{-1} & \sigma_2\tau_2^{-1} & \cdots & \sigma_k\tau_k^{-1} \\ q_1 & q_2 & \cdots & q_k \\ Id & Id & \cdots & Id \end{bmatrix}.$$

It is not hard to see that a table gives a well-defined homeomorphism of the appropriate Cantor space, and if two tables are related by a finite sequence of our four moves then they represent the same homeomorphism. The reader can also check that if a homeomorphism of an appropriate Cantor space is represented by two tables, then in fact these tables are in the same equivalence class under our four basic moves on tables. Thus, we can just consider our group elements to be the equivalence classes of tables with the aforementioned relations.

The composition of two different elements $u, v \in V_n(H)$ is easy to compute using the equivalences. Let u, $v \in V_n(H)$, such that u takes the prefix code P to the prefix code Q (resp. v takes P' to Q'). We need to find a prefix code S such that, for every element $s \in S$, there exists one element $q \in Q$ and one element $p' \in P'$ such that $q \leq_{pref} s$ and $p' \leq_{pref} s$. This can always be done by expanding P' and Q until we obtain the same prefix code S. Thus, without loss of generality:

$$u = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1 & \sigma_2 & \cdots & \sigma_k \\ s_1 & s_2 & \cdots & s_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix}, \quad v = \begin{bmatrix} s_1 & s_2 & \cdots & s_k \\ \sigma'_1 & \sigma'_2 & \cdots & \sigma'_k \\ q'_1 & q'_2 & \cdots & q'_k \\ \tau'_1 & \tau'_2 & \cdots & \tau'_k \end{bmatrix}.$$

Finally, we push up the action of u and push down the action of v:

$$u = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1\tau_1^{-1} & \sigma_2\tau_2^{-1} & \cdots & \sigma_k\tau_k^{-1} \\ s_1 & s_2 & \cdots & s_k \\ Id & Id & \cdots & Id \end{bmatrix},$$

$$v = \begin{bmatrix} s_1 & s_2 & \cdots & s_k \\ Id & Id & \cdots & Id \\ q'_1 & q'_2 & \cdots & q'_k \\ \tau'_1(\sigma'_1)^{-1} & \tau'_2(\sigma'_2)^{-1} & \cdots & \tau'_k(\sigma'_k)^{-1} \end{bmatrix},$$

so

$$v \circ u = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1\tau_1^{-1} & \sigma_2\tau_2^{-1} & \cdots & \sigma_k\tau_k^{-1} \\ q'_1 & q'_2 & \cdots & q'_k \\ \tau'_1(\sigma'_1)^{-1} & \tau'_2(\sigma'_2)^{-1} & \cdots & \tau'_k(\sigma'_k)^{-1} \end{bmatrix}.$$

We sum up the previous discussion in the following proposition:

Proposition 2.4. *$V_n(H)$ is a group with the composition.*

3. Topological Embeddings

In this section, we present topological embeddings between symmetric Thompson's groups. The key idea is, given any group $V_n(H)$, to translate the action of an element $\sigma \in H$ into a permutation $\widetilde{\sigma}$ of the elements of some prefix code of $\mathfrak{C}_m$. Therefore, $\widetilde{\sigma} \in V_m(G)$ for some G.

Our method will be first to understand when actions on prefix codes over smaller alphabets can represent embeddings of permutations on larger alphabets which commute with our core operations of expansion and contraction of prefix codes. With that understanding in hand, we can then build the desired embedding from a group $V_n(H)$ to a group $V_m(G)$ for $m \leq n$.

We first establish some useful definitions.

3.1. The Root Group $\mathcal{R}_G(S)$.

Given a linear order $\leq$ on $\mathcal{A}_n$ (we choose $0 < 1 < \ldots < n-1$), there is an induced standard dictionary order:

Definition 3.1 (Dictionary order)**.** Let $\mathcal{A}_n = \{a_0, \ldots, a_{n-1}\}$ be an alphabet with linear order $\leq$. We define the *dictionary order* on $\mathcal{A}_n^*$ as follows. Let $u, v \in \mathcal{A}_n^*$, then $u \leq_{dict} v$ if and only if:

(1) $u \leq_{pref} v$,
(2) $u \not\leq_{pref} v$ and there exist $p, s, t \in \mathcal{A}_n^*$ and $\alpha, \beta \in \mathcal{A}_n$ such that $u = p\alpha s, v = p\beta t$, and $\alpha < \beta$.

Let $2 \leq m < n \in \mathbb{N}$ be fixed, and let $G \leq \mathrm{Sym}(m)$. We assume for the construction Higman's condition: $n = k(m-1)+1$ for some $k \geq 0$ (so, $n \equiv 1 \mod (m-1)$). Then, we can find $S = \{s_0, \ldots, s_{n-1}\}$ an ordered prefix code of $\mathfrak{C}_m$ of length n (using the dictionary order). Observe for now that if $\sigma(S) = S \ \forall \sigma \in G$, then $\sigma : S \to S$ will induce the desired rearrangement $\widetilde{\sigma}$ of the elements of S, that is, $\widetilde{\sigma} \in \mathrm{Sym}_S \cong \mathrm{Sym}(n)$.

We will be interested in the set $\mathcal{T}$ of triples (m, n, G) where $m \leq n$, $n \equiv 1 \mod (m-1)$, and $G \in \mathrm{Sym}(m)$. We will say a triple $(m, n, G) \in \mathcal{T}$ is *satisfiable* if there is a prefix code $S \subset X_m^*$ with $|S| = n$ and where for each $\sigma \in G$ the action of σ on X_m^* preserves S. In this case we say S *is a solution for the triple* (m, n, G).

Definition 3.2 (Root group)**.** Given a triple $(m, n, G) \in \mathcal{T}$ and a solution S, we define the *root group* $\mathcal{R}_G(S)$ to be the group of permutations of S induced by the action of G on S.

It is the case that not every triple $(m, n, G) \in \mathcal{T}$ admits a solution S. However, when it does admit a solution S, it is immediate that $\mathcal{R}_G(S)$ is isomorphic to G.

Definition 3.3 (Cycle type)**.** Let $\sigma \in \mathrm{Sym}(n)$. The *cycle type* c of σ is the multiset (a set, but allowing multiple elements that are equal) of lengths of the cycles in the cycle decomposition of σ. We say that two subgroups $H, H' \in \mathrm{Sym}(n)$ are *cyclically isomorphic* if there exists an isomorphism $\psi : H \to H'$ which preserves the cycle type of every permutation, that is, $c(\sigma) = c(\psi(\sigma))$ for all $\sigma \in H$.

Remark 3.4. Subgroups H and H' are cyclically isomorphic if and only if they are conjugate in $\mathrm{Sym}(n)$, but our focus is on cycle structure and that is why we are using the language we have chosen. (N.B., there exist exotic automorphisms of $\mathrm{Sym}(6)$ which do not arise by conjugation, but these automorphisms change the cycle structure of some elements of order two.)

3.2. **The induced group $\mathcal{G}$.** We proceed to define the group $\mathcal{G} < V_m(G)$ such that $V_n(H)$ is isomorphic to $\mathcal{G}$.

Let $V_m(G)$ and $V_n(H)$ be two symmetric Thompson's groups such that m and n fulfill Higman's condition. Let $\widetilde{\mathcal{A}}_m = \{a_0, a_1, \dots, a_{n-1}\}$ be an ordered prefix code of $\mathfrak{C}_m$ of length n, such that $\mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ is well defined. We may think of $\mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ as a subgroup of $\mathrm{Sym}(n)$ by using the bijection from $\widetilde{\mathcal{A}}_m$ to $\mathcal{A}_n$ induced by the lexicographic ordering of $\widetilde{\mathcal{A}}_m$. Thus, we can define $\mathcal{G}$ as the set of equivalence classes of tables of the form:

$$v = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1 & \sigma_2 & \cdots & \sigma_k \\ q_1 & q_2 & \cdots & q_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix},$$

where $\sigma_i, \tau_i \in \mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ $\forall i$ and the prefix codes P and Q consist of words in the alphabet $\widetilde{\mathcal{A}}_m$, that is, every p_i and q_i is a nonempty concatenation of elements of $\widetilde{\mathcal{A}}_m$.

It is straightforward to check that $\mathcal{G}$ is a group, since the concatenation of two words in $\widetilde{\mathcal{A}}_m^*$ is another word in $\widetilde{\mathcal{A}}_m^*$. Tables of $\mathcal{G}$ are well defined by expansions and pushings, and the action of an element $\sigma \in \mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ on a word in $\widetilde{\mathcal{A}}_m$ gives another word in $\widetilde{\mathcal{A}}_m$ by the definition of root group, so the composition of any two elements in $\mathcal{G}$ gives another element of $\mathcal{G}$.

We recall and prove our topological embedding theorem:

Theorem 1.1 (Topological Embedding)**.** *Let $n, m \geq 2$ be natural numbers such that $m < n$, and let $G \leq \mathrm{Sym}(m)$, $H \leq \mathrm{Sym}(n)$. Suppose that:*

1. *There exists a prefix code $\widetilde{S}$ of $\mathfrak{C}_m$ such that $|\widetilde{S}| = n$,*
2. *the group $\mathcal{R}_G(\widetilde{S})$ is well defined, and*
3. *$\mathcal{R}_G(\widetilde{S})$ and H are cyclically isomorphic.*

Then $V_n(H)$ embeds in $V_m(G)$.

Proof. Let $V_m(G)$ and $V_n(H)$ two symmetric Thompson's groups. Let $\mathcal{A}_n = \{0, 1, \dots, n-1\}$ and $\widetilde{\mathcal{A}}_m = \{a_0, a_1, \dots, a_{n-1}\}$ such that $\mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ is well defined. We define the following *translating map*:

$$\begin{array}{rccc} \widetilde{t}: & \mathcal{A}_n & \longrightarrow & \widetilde{\mathcal{A}}_m \\ & i & \longrightarrow & a_i, \end{array}$$

and

$$\begin{array}{rccc} t: & H & \longrightarrow & \mathcal{R}_G(\widetilde{\mathcal{A}}_m) \\ & \sigma & \longrightarrow & \widetilde{\sigma} \end{array},$$

where t is the isomorphism between H and $\mathcal{R}_G(\widetilde{\mathcal{A}}_m)$. Note that t and $\widetilde{t}$ have the following property:

$$\widetilde{t}(\sigma(i)) = a_{\sigma(i)} = \widetilde{\sigma}(a_i) = t(\sigma)(a_i) = t(\sigma)(\widetilde{t}(i)), \forall \sigma \in H, \forall i \in \mathcal{A}_n,$$

as $\widetilde{\sigma}(a_i) = a_{\sigma(i)}, \forall \sigma \in H, \forall i \in \mathcal{A}_n$, since $\mathcal{R}_G(\widetilde{\mathcal{A}}_m)$ and H are cyclically isomorphic.

Our embedding is as follows:

$$v = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ \sigma_1 & \sigma_2 & \cdots & \sigma_k \\ q_1 & q_2 & \cdots & q_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix} \xrightarrow{\iota} \begin{bmatrix} \widetilde{t}(p_1) & \widetilde{t}(p_2) & \cdots & \widetilde{t}(p_k) \\ t(\sigma_1) & t(\sigma_2) & \cdots & t(\sigma_k) \\ \widetilde{t}(q_1) & \widetilde{t}(q_2) & \cdots & \widetilde{t}(q_k) \\ t(\tau_1) & t(\tau_2) & \cdots & t(\tau_k) \end{bmatrix}.$$

We see that ι commutes with expansions and pushings, that is, $\iota(\exp(v)) = \exp(\iota(v))$ and $\iota(\text{push}(v)) = \text{push}(\iota(v))\ \forall v \in V_n(H)$:

$$\text{push}(v) = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ Id & Id & \cdots & Id \\ q_1 & q_2 & \cdots & q_k \\ \tau_1\sigma_1^{-1} & \tau_2\sigma_2^{-1} & \cdots & \tau_k\sigma_k^{-1} \end{bmatrix},$$

$$\iota(\text{push}(v)) = \begin{bmatrix} \widetilde{t}(p_1) & \widetilde{t}(p_2) & \cdots & \widetilde{t}(p_k) \\ Id & Id & \cdots & Id \\ \widetilde{t}(q_1) & \widetilde{t}(q_2) & \cdots & \widetilde{t}(q_k) \\ t(\tau_1\sigma_1^{-1}) & t(\tau_2\sigma_2^{-1}) & \cdots & t(\tau_k\sigma_k^{-1}) \end{bmatrix},$$

$$\text{push}(\iota(v)) = \begin{bmatrix} \widetilde{t}(p_1) & \widetilde{t}(p_2) & \cdots & \widetilde{t}(p_k) \\ Id & Id & \cdots & Id \\ \widetilde{t}(q_1) & \widetilde{t}(q_2) & \cdots & \widetilde{t}(q_k) \\ t(\tau_1)t(\sigma_1^{-1}) & t(\tau_2)t(\sigma_2^{-1}) & \cdots & t(\tau_k)t(\sigma_k^{-1}) \end{bmatrix}.$$

As t is an isomorphism of groups, the commutativity follows. On the other hand, suppose that we expand the prefix code $P = \{p_1, \ldots, p_k\}$ on p_i (we argue below that ι commutes with expanding, but our argument uses the pushed version of v: it is easy to see that this is sufficient):

$$\exp(v) = \begin{bmatrix} p_1 & \cdots & p_i0 & \cdots & p_i(n-1) & \cdots & p_k \\ Id & \cdots & Id & \cdots & Id & \cdots & Id \\ q_1 & \cdots & q_i\|\tau_i\sigma_i^{-1}(0) & \cdots & q_i\|\tau_i\sigma_i^{-1}(n-1) & \cdots & q_k \\ \tau_1\sigma_1^{-1} & \cdots & \tau_i\sigma_i^{-1} & \cdots & \tau_i\sigma_i^{-1} & \cdots & \tau_k\sigma_k^{-1} \end{bmatrix}.$$

The table for $\iota(\exp(v))$ is:

$$\begin{bmatrix} \widetilde{t}(p_1) & \cdots & \widetilde{t}(p_i0) & \cdots & \widetilde{t}(p_i(n-1)) & \cdots & \widetilde{t}(p_k) \\ Id & \cdots & Id & \cdots & Id & \cdots & Id \\ \widetilde{t}(q_1) & \cdots & \widetilde{t}(q_i\|\tau_i\sigma_i^{-1}(0)) & \cdots & \widetilde{t}(q_i\|\tau_i\sigma_i^{-1}(n-1)) & \cdots & \widetilde{t}(q_k) \\ t(\tau_1\sigma_1^{-1}) & \cdots & t(\tau_i\sigma_i^{-1}) & \cdots & t(\tau_i\sigma_i^{-1}) & \cdots & t(\tau_k\sigma_k^{-1}) \end{bmatrix},$$

Finally, the table for $\exp(\iota(v))$ is:

$$\begin{bmatrix} \widetilde{t}(p_1) & \cdots & \widetilde{t}(p_i)a_0 & \cdots & \widetilde{t}(p_i)a_{n-1} & \cdots & \widetilde{t}(p_k) \\ Id & \cdots & Id & \cdots & Id & \cdots & Id \\ \widetilde{t}(q_1) & \cdots & \widetilde{t}(q_i)\|t(\tau_i\sigma_i^{-1})(a_0) & \cdots & \widetilde{t}(q_i)\|t(\tau_i\sigma_i^{-1})(a_{n-1}) & \cdots & \widetilde{t}(q_k) \\ t(\tau_1\sigma_1^{-1}) & \cdots & t(\tau_i\sigma_i^{-1}) & \cdots & t(\tau_i\sigma_i^{-1}) & \cdots & t(\tau_k\sigma_k^{-1}) \end{bmatrix},$$

Both tables $\exp(\iota(v)) = \iota(\exp(v))$ are equal as:

$$\widetilde{t}(p_i\|j) = \widetilde{t}(p_i)\|\widetilde{t}(j) = \widetilde{t}(p_i)\|\widetilde{t}(a_j),$$

$$\widetilde{t}(q_i\|\tau_i\sigma_i^{-1}(j)) = \widetilde{t}(q_i)\|\widetilde{t}(\tau_i\sigma_i^{-1}(j)) = \widetilde{t}(q_i)\|t(\tau_i\sigma_i^{-1})(\widetilde{t}(j)) = \widetilde{t}(q_i)\|t(\tau_i\sigma_i^{-1})(a_j).$$

(We have used the concat symbol “‖” in these tables and the immediate discussion after, wherever we think it makes terms easier to read. In our later explanations we will generally refrain from using it.)

To finish the proof, we need to show that $\iota(v) \circ \iota(u) = \iota(v \circ u)$. This follows easily from the fact that ι commutes with expansions and pushings. Without loss of generality, consider:

$$u = \begin{bmatrix} p_1 & p_2 & \cdots & p_k \\ Id & Id & \cdots & Id \\ s_1 & s_2 & \cdots & s_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \end{bmatrix}, \quad \iota(u) = \begin{bmatrix} \widetilde{t}(p_1) & \widetilde{t}(p_2) & \cdots & \widetilde{t}(p_k) \\ Id & Id & \cdots & Id \\ \widetilde{t}(s_1) & \widetilde{t}(s_2) & \cdots & \widetilde{t}(s_k) \\ t(\tau_1) & t(\tau_2) & \cdots & t(\tau_k) \end{bmatrix},$$

$$v = \begin{bmatrix} s_1 & s_2 & \cdots & s_k \\ \tau_1 & \tau_2 & \cdots & \tau_k \\ q'_1 & q'_2 & \cdots & q'_k \\ \tau'_1 & \tau'_2 & \cdots & \tau'_k \end{bmatrix}, \quad \iota(v) = \begin{bmatrix} \widetilde{t}(s_1) & \widetilde{t}(s_2) & \cdots & \widetilde{t}(s_k) \\ t(\tau_1) & t(\tau_2) & \cdots & t(\tau_k) \\ \widetilde{t}(q'_1) & \widetilde{t}(q'_2) & \cdots & \widetilde{t}(q'_k) \\ t(\tau'_1) & t(\tau'_2) & \cdots & t(\tau'_k) \end{bmatrix}.$$

Thus:

$$\iota(v) \circ \iota(u) = \iota(v \circ u) = \begin{bmatrix} \widetilde{t}(p_1) & \widetilde{t}(p_2) & \cdots & \widetilde{t}(p_k) \\ Id & Id & \cdots & Id \\ \widetilde{t}(q'_1) & \widetilde{t}(q'_2) & \cdots & \widetilde{t}(q'_k) \\ t(\tau'_1) & t(\tau'_2) & \cdots & t(\tau'_k) \end{bmatrix}.$$

□

4. Algebraic Embeddings

In this section we present some algebraic embeddings $V_m(G) \rightarrowtail V_n(H)$ between symmetric Thompson's groups (recall here $n - m = k(m-1)$ for some positive k, $G \leq \mathrm{Sym}(m)$ and with H a particular extended version of G in $\mathrm{Sym}(n)$).

We call these embeddings “algebraic” as they do not arise via topological conjugacy. In particular, these embeddings do not preserve the orbit lengths of the points of $\mathfrak{C}_n$ (when $n > m$, which is our primary case of interest).

4.1. **Successors.**

Here, we give the key idea for our algebraic embeddings, which relies on extending an idea of Birget into our context.

The *successor* of an element, expressed as a table, was defined in [1] in order to embed $V_2(Id)$ in $V_n(Id)$, for all $n \geq 2$. We generalise Birget's definition.

Definition 4.1 (Set of prefixes)**.** [1] Let $P \subset \mathcal{A}_n^*$ be a prefix code of $\mathfrak{C}_n$. We define the *set of prefixes of* P, $\mathrm{spref}(P)$ as follows:

$$\mathrm{spref}(P) = \{w \in \mathcal{A}_n^* : \exists p \in P,\, w <_{pref} p\}.$$

In other words, $\mathrm{spref}(P)$ is the set of strict prefixes of the elements of P.

We are embedding a symmetric Thompson's group on alphabet $\mathcal{A}_m$ into a symmetric Thompson's group on alphabet $\mathcal{A}_n$, where $m \leq n$. For this, we will assume $\mathcal{A}_m \subseteq \mathcal{A}_n$. And in particular we set $\mathcal{A}_m = \{a_0, a_1, \ldots, a_{m-1}\}$ and $\mathcal{A}_n = \mathcal{A}_m \cup \{a_m, a_{m+1}, \ldots, a_{n-1}\}$ with symbols with distinct indices being distinct (so that $|\mathcal{A}_n| = n$).

In what follows, we take a prefix code $P \subset a_{m-1} \| \mathcal{A}_m^*$ (so each element of P begins with the letter a_{m-1}) and transform it to a new prefix code $\mathrm{Succ}(P) \subset \mathcal{A}_n^*$ by appending letters from the set $\{a_m, a_{m+1}, \ldots, a_n\}$.

Definition 4.2 (Successor)**.** Let $P \subset a_{m-1}\|\{a_0, \ldots, a_{m-1}\}^*$ be a prefix code (complete, were we to remove the initial prefix letter a_{m-1}, so that $|P| \equiv 1 \mod (m-1)$) with $|P| = l \geq 1$, and let $\{p_1, \ldots, p_l\}$ be the ordered list of all the elements of P, using the *reverse* dictionary order.

We build a new prefix code $\mathrm{Succ}(P)$ inductively using our ordered list $(p_1, p_2, \ldots, p_l)$.

Let k be the smallest non-negative integer so that $n - m = k(m-1)$ (this k will exist when m and n satisfy Higman's condition, which we require to build our embeddings).

We define (inductively) nested sets $P_{s,i}$, where s will grow from 1 to l, and for each value of s, we will have i grow from 1 to k.

Set $\mathcal{A}_{m,n} := \{a_m, a_{m+1}, \ldots, a_{n-1}\}$. For every $p_s \in P$, and $i \in \{1, 2, \ldots, k\}$ the *i-th successor* $(p_s)'_i$ of p_s is the element of $\mathrm{spref}(P)\|\mathcal{A}_{m,n}$ defined as follows, assuming that

$$P_{s,i-1} = \left\{ \begin{array}{c} (p_1)'_1, (p_1)'_2, \ldots, (p_1)'_k, \\ (p_2)'_1, (p_2)'_2, \ldots, (p_2)'_k, \\ \vdots \\ (p_s)'_1, (p_s)'_2, \ldots, (p_s)'_{i-1} \end{array} \right\}$$

has already been defined, we set:

$$(p_s)'_i = \min\{xa_j \in \mathrm{spref}(P)\|\mathcal{A}_{m,n} : p_s <_{dict} xa_j \text{ and } xa_j \notin P_{s,i-1}\},$$

where min uses the dictionary order in $\{a_0, \ldots, a_{n-1}\}$.

Example 4.3. Suppose $m = 3$ and $n = 5$, so that $k = 1$. In the definition above, $a_{m-1} = 2$. So, consider the set $P = \{20, 210, 211, 212, 22\}$. Now, $k = 1$ and $\mathrm{spref}(P) = \{\varepsilon, 2, 21\}$. We obtain

$$\begin{array}{ll} p_1 = 22 & (p_1)'_1 = 23 \\ p_2 = 212 & (p_2)'_1 = 213 \\ p_3 = 211 & (p_3)'_1 = 214 \\ p_4 = 210 & (p_4)'_1 = 24 \\ p_5 = 20 & (p_5)'_1 = 3. \end{array}$$

Remark 4.4. The three constants, n, m, k are not arbitrary, as the system of successors needs to be well defined. If every element has k successors, then:

$$n - m = k(m-1),\ k \geq 0,$$

which is Higman's condition.

Proof. If we expand an element $p_i \in P$, we need to assign successors to each element $p_i a_j$ for every $0 \leq j \leq m-1$. In particular, the number of successors k of every leaf does not vary, and each element $p_i a_r$ for every $m \leq r \leq n-1$ needs to be the successor of some element in $\widetilde{P} = (P \backslash \{p_i\}) \cup \{p_i a_0, \ldots, p_i a_{m-1}\}$. Then $\widetilde{P}$ has $m-1$ more elements than P and there are $n-m$ new elements $p_i a_r$ for $m \leq r \leq n-1$. Thus, we need $m-1$ to evenly divide $n-m$, and k is the factor of this division. □

We proceed to prove the following lemma, essential for the proof of Theorem 1.2:

Lemma 4.5. *Suppose $m \leq n$ are naturals so that there is k natural with $n - m = k(m-1)$. Suppose l is a positive integer congruent to m modulo $m-1$. Let $S = \{a_m, \ldots, a_{n-1}\}$ and let $P \subset a_{m-1}\|\{a_0, \ldots, a_{m-1}\}^*$ be an l-element prefix code, ordered as $p_l <_{dict} p_{l-1} <_{dict} \ldots <_{dict} p_1$. Let i with $1 \leq i \leq l$. Then, the* *successors* $(p_i)'_1$*,* $(p_i)'_2$*, …,* $(p_i)'_k$ *are well defined, and furthermore, the expansion in which we replace* P *by* $\widetilde{P} = (P\backslash\{p_i\}) \cup p_i\{a_0, \ldots, a_{m-1}\}$ *has successors* $(p_ia_j)'_i$ *uniquely determined as follows:*

$$\begin{aligned}
(p_ia_{m-1})'_1 &= p_ia_m\\
&\vdots\\
(p_ia_{m-1})'_k &= p_ia_{m+k-1}\\
(p_ia_{m-2})'_1 &= p_ia_{m+k}\\
&\vdots\\
(p_ia_{m-2})'_k &= p_ia_{m+2k-1}\\
&\vdots\\
(p_ia_1)'_1 &= p_ia_{m+(m-2)k}\\
&\vdots\\
(p_ia_1)'_k &= p_ia_{m+(m-1)k-1} = p_ia_n\\
(p_ia_0)'_1 &= (p_i)'_1\\
&\vdots\\
(p_ia_0)'_k &= p_ia_{m+(m-1)k} = (p_i)'_k.
\end{aligned}$$

Proof. We prove the two statements by induction on l.

Base Case ($l = 1$)*:*
If $l = 1$ then $P = \{a_{m-1}\}$. We have $\operatorname{spref} P = \{\varepsilon\}$. It then follows that the k successors are, the set $\{a_m, a_{m+1}, \ldots, a_{m+k-1}\}$, noting that these are given in order and are the results of the inductive definition of the k successors of a_{m-1}. Thus we have in the base case that the successors are well defined. We need to verify the existence of well defined successors for an expansion of $P = \{a_{m-1}\}$. In this case, P admits only one expansion, which is precisely the set $\widetilde{P} = \{a_{m-1}a_{m-1}, a_{m-1}a_{m-2}, \ldots, a_{m-1}a_0\}$. We have $\operatorname{spref}(\widetilde{P}) = \{\varepsilon, a_{m-1}\}$ and we have

$$\begin{aligned}
(a_{m-1}a_{m-1})'_1 &= a_{m-1}a_m\\
&\vdots\\
(a_{m-1}a_{m-1})'_k &= a_{m-1}a_{m+k-1}\\
(a_{m-1}a_{m-2})'_1 &= a_{m-1}a_{m+k}\\
&\vdots\\
(a_{m-1}a_{m-2})'_k &= a_{m-1}a_{m+2k-1}\\
&\vdots\\
(a_{m-1}a_1)'_1 &= a_{m-1}a_{m+(m-2)k}\\
&\vdots\\
(a_{m-1}a_1)'_k &= a_{m-1}a_{m+(m-1)k-1} = a_{m-1}a_n\\
(a_{m-1}a_0)'_1 &= (a_{m-1})'_1\\
&\vdots\\
(a_{m-1}a_0)'_k &= a_{m-1}a_{m+(m-1)k} = (a_{m-1})'_k.
\end{aligned}$$

We can directly observe these successors are well defined and distinct. Thus, the statement is true for $l = 1$.

Inductive Case ($l > 1$):
Now let us assume that $\widetilde{P}$ is a result of v expansions from the one-element prefix code $\{a_{m-1}\}$, for some $v \geq 1$, where for any prefix code resulting from u expansions from $\{a_{m-1}\}$ for $0 \leq u < v$ the statement of the lemma holds. We will show that the successors of our expansion $\widetilde{P}$ are well defined. In particular, if P is the prefix code (of size l) arising from doing only the first $t-1$ expansions from $\{a_{m-1}\}$ towards the prefix code $\widetilde{P}$, and p_i is the element of P which is being replaced by and m-fold expansion to create $\widetilde{P}$, then by induction, the successors $(p_i)'_1$, $(p_i)'_2$, …, $(p_i)'_k$ of $p_i \in P$ are well defined.

Note that $P_{i-1,k} = \widetilde{P}_{i-1,k}$, as the involved subsets of both P and $\widetilde{P}$ are equal. Thus the first successor to assign is $(p_i a_{m-1})'_1$. Suppose that $(p_i a_{m-1})'_1 <_{dict} p_i a_m$, then $(p_i a_{m-1})'_1 = pa_t$ for some $p \in \operatorname{spref}(P) : p <_{dict} p_i$ and $a_t \in \{a_m, \ldots, a_{n-1}\}$. Thus, before expanding P, pa_t is one of the successors of some $p_r \in P$.

If $p_r <_{dict} p_i$, then the set of successors of p_i is defined before the set of successors of p_r. Because p_i is expanded, the set of k successors of $p_i a_{m-1}$ is equal to the set of successors of p_i, and this set does not contain pa_t, which is a contradiction. On the other hand, if $p_r >_{dict} p_i$, then $pa_t \in P_{i-1,k} = \widetilde{P}_{i-1,k}$, which is also a contradiction. Thus $(p_i a_{m-1})'_1 = p_i a_m$. We can use a similar argument for all $(p_i a_{m-1})'_1 \ldots (p_i a_1)'_k$.

For $p_i a_0$, all successors of the form $p_i a_s$, $a_s \in \{a_m, \ldots, a_{n-1}\}$, have already been assigned. Thus, the remaining k successors are precisely the k successors of p_i, taken in order. □

Remark 4.6. Birget in [1] gives a formula for the i-th successor of an element, for the case of $m = 2$. The statement of Lemma 4.5 above shows the natural generalisation of that formula holds when we have the Higman Condition (as we must for successors to be well defined). The resulting formula is given as follows:

Let $P \subset a_{m-1} \| \{a_0, \ldots, a_{m-1}\}^*$ be a prefix code with $|P| \geq 2$, such that the elements of P are ordered in reverse dictionary order. Then every element of $w \in P$ can be written uniquely in the form $ua_i a_0^t$, where $u \in \{a_0, \ldots, a_{m-1}\}^*$ and $t \geq 0$. The i-th successor of w is:

$$(w)'_i = (ua_j a_0^t)'_i = ua_{m-1+(m-1-j)k+i}.$$

We stress that this formula is only valid if P is ordered in reverse dictionary order.

4.2. **The algebraic embedding.** We proceed to define the algebraic embedding of $V_m(G)$ in $V_n(H) = V_n(G_{ext})$. Let $g \in V_m(G)$, given by the following table:

$$g = \begin{bmatrix} p_1 & p_2 & \cdots & p_l \\ \sigma_1 & \sigma_2 & \cdots & \sigma_l \\ q_1 & q_2 & \cdots & q_l \\ \tau_1 & \tau_2 & \cdots & \tau_l \end{bmatrix}$$

We define the embedding $\iota(g)$ below. The resulting tables are large, and our notation requires some explanation. The idea of the embedding is to use the identity map initially, and at a_{m+k} and later letters, but under the address a_{m-1} we place the prefix code p_1 to p_l, and we also require action under the successors. The first row then has entries following the ordered list given here (wrapped at natural locations due to page length constraints):

$$\begin{gathered}
a_0, a_1, \ldots, a_{m-2}, \\
a_{m-1}p_1, a_{m-1}p_2, \ldots, a_{m-1}p_l, \\
(a_{m-1}p_1)'_1, (a_{m-1}p_2)'_1, \ldots, (a_{m-1}p_l)'_1, \\
(a_{m-1}p_1)'_2, (a_{m-1}p_2)'_2, \ldots, (a_{m-1}p_l)'_2, \\
\cdots \\
(a_{m-1}p_1)'_k, (a_{m-1}p_2)'_k, \ldots, (a_{m-1}p_l)'_k, \\
a_{m+k}, a_{m+k+1}, \ldots, a_{n-1}.
\end{gathered}$$

We use vertical bars "|" in our table at the same locations that we placed line-wraps in the row detailed above, for clarity of grouping. The element $\iota(g)$ is now given by the following table:

$$\left[\begin{array}{ccc|ccc|ccc|c}
a_0 & \cdots & a_{m-2} & a_{m-1}p_1 & \cdots & a_{m-1}p_l & (a_{m-1}p_1)'_1 & \cdots & (a_{m-1}p_l)'_1 & \cdots \\
Id & \cdots & Id & \sigma'_1 & \cdots & \sigma'_l & \sigma'_1 & \cdots & \sigma'_l & \cdots \\
a_0 & \cdots & a_{m-2} & a_{m-1}q_1 & \cdots & a_{m-1}q_l & (a_{m-1}q_1)'_1 & \cdots & (a_{m-1}q_l)'_1 & \cdots \\
Id & \cdots & Id & \tau'_1 & \cdots & \tau'_l & \tau'_1 & \cdots & \tau'_l & \cdots
\end{array}\right.$$

$$\left.\begin{array}{c|ccc|ccc}
\cdots & (a_{m-1}p_1)'_k & \cdots & (a_{m-1}p_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & \sigma'_1 & \cdots & \sigma'_l & Id & \cdots & Id \\
\cdots & (a_{m-1}q_1)'_k & \cdots & (a_{m-1}q_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & \tau'_1 & \cdots & \tau'_l & Id & \cdots & Id
\end{array}\right]$$

Note that the set of successors of $P = \{p_1, \ldots, p_l\}$ are assigned supposing that $p_l <_{dict} \ldots <_{dict} p_1$. Therefore, the set of successors of $Q = \{q_1, \ldots, q_l\}$ is assigned following the order $q_l \to \cdots \to q_1$, which does not need to follow the dictionary order on Q.

Indeed, the first and third rows of $\iota(g)$ are both prefix codes of $\mathfrak{C}_n$. On the one hand, suppose that the number of columns of g is $l = m + d(m-1)$ for some $d \geq 0$. It follows that the number of columns of $\iota(g)$ whose elements of the first row start with a_{m-1} is $n + d(n-1)$ (observe that the last k terms from the successor substitution will not begin with a_{m-1}). On the other hand, as the number of columns of g is $(m + d(m-1))$, and we assign k successors to every column, we have $(m + d(m-1))(k+1)$ columns on $\iota(g)$. As we have Higman's Condition, the reader can verify that $(m + d(m-1))(k+1) = n + d(n-1) + k$. From this, we see firstly that $(n-1)|k$, but more importantly, this embedding/successor operation does not place any constraints on the number of expansions d that were used to create the original prefix code for the domain of g.

We now recall and prove our algebraic embedding theorem:

Theorem 1.2 (Algebraic Embedding)**.** *Let $n, m \geq 2$ be natural numbers such that $n = k(m-1) + m$ for some $k \geq 1$, and let $G \leq \mathrm{Sym}(m)$. Let $H = G_{ext} \leq \mathrm{Sym}(n)$ the extended symmetric group of H, whose elements act as the elements of H on the first m elements, and act as the identity on the remaining $n - m$.*

Then $V_m(G)$ embeds in $V_n(H)$.

Proof. If we push down the action of every σ_i, we have:

$$\mathrm{push}(g) = \begin{bmatrix}
p_1 & p_2 & \cdots & p_l \\
Id & Id & \cdots & Id \\
q_1 & q_2 & \cdots & q_l \\
\tau_1\sigma_1^{-1} & \tau_2\sigma_2^{-1} & \cdots & \tau_l\sigma_l^{-1}
\end{bmatrix}$$

Thus the table for $\iota(\text{push}(g))$ is:

$$\left[\begin{array}{c|ccc|ccc|c}
a_0 \cdots a_{m-2} & a_{m-1}p_1 & \cdots & a_{m-1}p_l & (a_{m-1}p_1)'_1 & \cdots & (a_{m-1}p_l)'_1 & \cdots \\
Id \cdots \ Id & Id & \cdots & Id & Id & \cdots & Id & \cdots \\
a_0 \cdots a_{m-2} & a_{m-1}q_1 & \cdots & a_{m-1}q_l & (a_{m-1}q_1)'_1 & \cdots & (a_{m-1}q_l)'_1 & \cdots \\
Id \cdots \ Id & (\tau_1\sigma_1^{-1})' & \cdots & (\tau_l\sigma_l^{-1})' & (\tau_1\sigma_1^{-1})' & \cdots & (\tau_l\sigma_l^{-1})' & \cdots
\end{array}\right.$$

$$\left.\begin{array}{c|ccc|ccc}
\cdots & (a_{m-1}p_1)'_k & \cdots & (a_{m-1}p_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & Id & \cdots & Id & Id & \cdots & Id \\
\cdots & (a_{m-1}q_1)'_k & \cdots & (a_{m-1}q_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & (\tau_1\sigma_1^{-1})' & \cdots & (\tau_l\sigma_l^{-1})' & Id & \cdots & Id
\end{array}\right]$$

On the other hand the table for $\text{push}(\iota(g))$ is:

$$\left[\begin{array}{c|ccc|ccc|c}
a_0 \cdots a_{m-2} & a_{m-1}p_1 & \cdots & a_{m-1}p_l & (a_{m-1}p_1)'_1 & \cdots & (a_{m-1}p_l)'_1 & \cdots \\
Id \cdots \ Id & Id & \cdots & Id & Id & \cdots & Id & \cdots \\
a_0 \cdots a_{m-2} & a_{m-1}q_1 & \cdots & a_{m-1}q_l & (a_{m-1}q_1)'_1 & \cdots & (a_{m-1}q_l)'_1 & \cdots \\
Id \cdots \ Id & \tau'_1(\sigma'_1)^{-1} & \cdots & \tau'_l(\sigma'_l)^{-1} & \tau'_1(\sigma'_1)^{-1} & \cdots & \tau'_l(\sigma'_l)^{-1} & \cdots
\end{array}\right.$$

$$\left.\begin{array}{c|ccc|ccc}
\cdots & (a_{m-1}p_1)'_k & \cdots & (a_{m-1}p_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & Id & \cdots & Id & Id & \cdots & Id \\
\cdots & (a_{m-1}q_1)'_k & \cdots & (a_{m-1}q_l)'_k & a_{m+k} & \cdots & a_{n-1} \\
\cdots & \tau'_1(\sigma'_1)^{-1} & \cdots & \tau'_l(\sigma'_l)^{-1} & Id & \cdots & Id
\end{array}\right]$$

Recall from the statement of Theorem 1.2 that for an element $\tau \in \text{Sym}(m)$, the extended version τ' of τ in $\text{Sym}(n)$ is that element of $\text{Sym}(n)$ which agrees with τ on the set $\mathcal{A}_m$ and acts as the identity on the points of $\mathcal{A}_{m,n}$ in $\mathcal{A}_n$. Thus, both tables are equal, as $(\tau_i\sigma_i^{-1})' = (\tau'_i)(\sigma'_i)^{-1}$, $\forall i \in \{i, \ldots, l\}$. If we expand g on p_i:

$$\exp(g) = \begin{bmatrix}
p_1 & p_2 & \cdots & p_ia_0 & \cdots & p_ia_{m-1} & \cdots & p_l \\
Id & Id & \cdots & Id & \cdots & Id & \cdots & Id \\
q_1 & q_2 & \cdots & q_i\tau_i(a_0) & \cdots & q_i\tau_i(a_{m-1}) & \cdots & q_l \\
\tau_1 & \tau_2 & \cdots & \tau_i & \cdots & \tau_i & \cdots & \tau_l
\end{bmatrix}$$

Then the table for $\iota(\exp(g))$ is:

$$\left[\begin{array}{c|ccccccc|c}
a_0 \cdots a_{m-2} & a_{m-1}p_1 & \cdots & a_{m-1}p_ia_0 & \cdots & a_{m-1}p_ia_{m-1} & \cdots & a_{m-1}p_l & \cdots \\
Id \cdots \ Id & Id & \cdots & Id & \cdots & Id & \cdots & Id & \cdots \\
a_0 \cdots a_{m-2} & a_{m-1}q_1 & \cdots & a_{m-1}q_i\tau_i(a_0) & \cdots & a_{m-1}q_i\tau_i(a_{m-1}) & \cdots & a_{m-1}q_l & \cdots \\
Id \cdots \ Id & \tau'_1 & \cdots & \tau'_i & \cdots & \tau'_i & \cdots & \tau'_l & \cdots
\end{array}\right.$$

$$\left.\begin{array}{c|ccccc|ccc}
\cdots & \cdots & (a_{m-1}p_ia_0)'_j & \cdots & (a_{m-1}p_ia_{m-1})'_j & \cdots & a_{m+k} & \cdots & a_{n-1} \\
\cdots & \cdots & Id & \cdots & Id & \cdots & Id & \cdots & Id \\
\cdots & \cdots & (a_{m-1}q_i\tau_i(a_0))'_j & \cdots & (a_{m-1}q_i\tau_i(a_{m-1}))'_j & \cdots & a_{m+k} & \cdots & a_{n-1} \\
\cdots & \cdots & \tau'_i & \cdots & \tau'_i & \cdots & Id & \cdots & Id
\end{array}\right]$$

On the other hand, the table for $\exp(\iota(g))$ is:

$$\left[\begin{array}{cc|ccccccc|c} a_0 \cdots & a_{m-2} & a_{m-1}p_1 & \cdots & a_{m-1}p_i a_0 & \cdots & a_{m-1}p_i a_{n-1} & \cdots & a_{m-1}p_l & \cdots \\ Id \cdots & Id & Id & \cdots & Id & \cdots & Id & \cdots & Id & \cdots \\ a_0 \cdots & a_{m-2} & a_{m-1}q_1 & \cdots & a_{m-1}q_i\tau_i'(a_0) & \cdots & a_{m-1}q_i\tau_i'(a_{n-1}) & \cdots & a_{m-1}q_l & \cdots \\ Id \cdots & Id & \tau_1' & \cdots & \tau_i' & \cdots & \tau_i' & \cdots & \tau_l' & \cdots \end{array}\right.$$

$$\left.\begin{array}{c|ccccccc|ccc} \cdots & \cdots & (a_{m-1}p_1)_j' & \cdots & (a_{m-1}p_i)_j' & \cdots & (a_{m-1}p_l)_j' & \cdots & a_{m+k} & \cdots & a_{n-1} \\ \cdots & \cdots & Id & \cdots & Id & \cdots & Id & \cdots & Id & \cdots & Id \\ \cdots & \cdots & (a_{m-1}q_1)_j' & \cdots & (a_{m-1}q_i)_j' & \cdots & (a_{m-1}q_l)_j' & \cdots & a_{m+k} & \cdots & a_{n-1} \\ \cdots & \cdots & \tau_1' & \cdots & \tau_i' & \cdots & \tau_l' & \cdots & Id & \cdots & Id \end{array}\right]$$

It is straightforward to check that for both tables the columns starting at $a_{m-1}p_ia_s$ for $0 \leq s \leq m-1$ are equal, as $\tau_i'(s) = \tau_i(s), \forall s \in \{0, \dots, m-1\}$.

For $m \leq s \leq n-1$ by Lemma 4.5, we know there is a correspondence between the first two rows of $\iota(\exp(g))$ and the first two rows of $\exp(\iota(g))$. On the other hand, from the description of the algebraic embedding, the order in which successors for every q_j are selected depends only on the order of $\{p_1, \dots, p_l\}$, so we have the following calculations:

$$\begin{array}{lll} (a_{m-1}q_i\tau_i(a_{m-1}))_1' & = a_{m-1}q_ia_m & = a_{m-1}q_i\tau_i'(a_m) \\ & \vdots & \vdots \\ (a_{m-1}q_i\tau_i(a_{m-1}))_k' & = a_{m-1}q_ia_{m+k-1} & = a_{m-1}q_i\tau_i'(a_{m+k-1}) \\ (a_{m-1}q_i\tau_i(a_{m-2}))_1' & = a_{m-1}q_ia_{m+k} & = a_{m-1}q_i\tau_i'(a_{m+k}) \\ & \vdots & \vdots \\ (a_{m-1}q_i\tau_i(a_{m-2}))_k' & = a_{m-1}q_ia_{m+2k-1} & = a_{m-1}q_i\tau_i'(a_{m+2k-1}) \\ & \vdots & \vdots \\ (a_{m-1}q_i\tau_i(a_1))_1' & = a_{m-1}q_ia_{m+(m-2)k} & = a_{m-1}q_i\tau_i'(a_{m+(m-2)k}) \\ & \vdots & \vdots \\ (a_{m-1}q_i\tau_i(a_1))_k' & = a_{m-1}q_ia_n & = a_{m-1}q_i\tau_i'(a_{m+(m-1)k-1}) \\ (a_{m-1}q_i\tau_i(a_0))_1' & = (a_{m-1}q_i)_1' & \\ & \vdots & \\ (a_{m-1}q_i\tau_i(a_0))_k' & = (a_{m-1}q_i)_k'. & \end{array}$$

That is, e.g., $(a_{m-1}q_i\tau_i(a_{m-1}))_1'$ comes first in the choice of successor as $(a_{m-1}p_ia_{m-1})_1'$ appears first under the order of the p_i, independent of τ_i. Thus, the latter two rows of these tables are also equivalent.

Finally, it is easy to see that $\iota(h \circ g) = \iota(h) \circ \iota(g)$, as ι commutes with expansions and pushings. We only need to obtain row equality on the first part of the table as the remaining part depends entirely on P (resp. on Q for the element h):

$$\iota(g) = \begin{bmatrix} \cdots & a_{m-1}p_i & \cdots & (a_{m-1}p_i)_j' & \cdots \\ \cdots & Id & \cdots & Id & \cdots \\ \cdots & a_{m-1}q_i & \cdots & (a_{m-1}q_i)_j' & \cdots \\ \cdots & \tau_i' & \cdots & \tau_i' & \cdots \end{bmatrix},$$

$$\iota(h) = \begin{bmatrix} \cdots & a_{m-1}q_i & \cdots & (a_{m-1}q_i)'_j & \cdots \\ \cdots & \tau'_i & \cdots & \tau'_i & \cdots \\ \cdots & a_{m-1}r_i & \cdots & (a_{m-1}r_i)'_j & \cdots \\ \cdots & \tau''_i & \cdots & \tau''_i & \cdots \end{bmatrix},$$

$$\iota(h) \circ \iota(g) = \begin{bmatrix} \cdots & a_{m-1}p_i & \cdots & (a_{m-1}p_i)'_j & \cdots \\ \cdots & Id & \cdots & Id & \cdots \\ \cdots & a_{m-1}r_i & \cdots & (a_{m-1}r_i)'_j & \cdots \\ \cdots & \tau''_i & \cdots & \tau''_i & \cdots \end{bmatrix} = \iota(h \circ g).$$

Thus the result follows. □

References

[1] J. C. Birget. New embeddings between the Higman-Thompson groups. *Communications in Algebra*, 48(8):3429–3438, 2020.

[2] C. Bleak, C. Donoven, and J. Jonušas. Some isomorphism results for Thompson-like groups $V_n(G)$. *Israel J. Math.*, 222(1):1–19, 2017.

[3] C. Bleak and D. Lanoue. A family of non-isomorphism results. *Geom. Dedicata*, 146:21–26, 2010.

[4] C. Bleak, F. Matucci, and M. Neunhöffer. Embeddings into Thompson's group V and $co\mathcal{CF}$ groups. *J. Lond. Math. Soc. (2)*, 94(2):583–597, 2016.

[5] N. M. Bon. Rigidity properties of full groups of pseudogroups over the cantor set. https://arxiv.org/pdf/1801.10133.pdf, 2018.

[6] L. E. J. Brouwer. On the structure of perfect sets of points. In *KNAW, Proceedings*, volume 12, pages 1909–1910, 1910.

[7] D. Farley. Local similarity groups with context-free co-word problem. In *Topological methods in group theory*, volume 451 of *London Math. Soc. Lecture Note Ser.*, pages 67–91. Cambridge Univ. Press, Cambridge, 2018.

[8] R. I. Grigorchuk, V. Nekrashevych, and V. Sushchanskiĭ. Automata, dynamical systems, and groups. *Proc. Steklov Inst. Math*, 231(4):128–203, 2000.

[9] G. Higman. *Finitely presented infinite simple groups*. Department of Pure Mathematics, Department of Mathematics, I.A.S. Australian National University, Canberra, 1974. Notes on Pure Mathematics, No. 8 (1974).

[10] D. F. Holt, S. Rees, C. E. Röver, and R. M. Thomas. Groups with context-free co-word problem. *J. London Math. Soc. (2)*, 71(3):643–657, 2005.

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

[12] J. Lehnert. *Gruppen von quasi-automorphismen.* PhD thesis, Goethe Universität, Frankfurt, 2008.

[13] 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.

[14] V. V. Nekrashevych. Cuntz-Pimsner algebras of group actions. *J. Operator Theory*, 52(2):223–249, 2004.

[15] C. E. Röver. Constructing finitely presented simple groups that contain Grigorchuk groups. *J. Algebra*, 220(1):284–313, 1999.

[16] E. A. Scott. A construction which can be used to produce finitely presented infinite simple groups. *J. Algebra*, 90(2):294–322, 1984.

[17] E. A. Scott. The embedding of certain linear and abelian groups in finitely presented simple groups. *J. Algebra*, 90(2):323–332, 1984.

[18] E. A. Scott. A finitely presented simple group with unsolvable conjugacy problem. *J. Algebra*, 90(2):333–353, 1984.

Instituto de Ciencias Matemáticas, Madrid, Spain.
Email address: julio.aroca@icmat.es

University of St Andrews, St Andrews, Scotland.
Email address: cb211@st-andrews.ac.uk