Journal of Pure and Applied Algebra 213 (2009) 264–278

Contents lists available at ScienceDirect

Journal of Pure and Applied Algebra

journal homepage: www.elsevier.com/locate/jpaa

Monoid generalizations of the Richard Thompson groups

Jean-Camille Birget

Department of Computer Science, Rutgers University at Camden, Camden, NJ 08102, USA

ARTICLE INFO

Article history:
Received 9 April 2007
Received in revised form 27 April 2008
Available online 24 July 2008
Communicated by M. Sapir

MSC:
20F10
20F05
20M05
68Q17

ABSTRACT

The groups $G_{k,1}$ of Richard Thompson and Graham Higman can be generalized in a natural way to monoids, that we call $M_{k,1}$, and to inverse monoids, called $Inv_{k,1}$; this is done by simply generalizing bijections to partial functions or partial injective functions. The monoids $M_{k,1}$ have connections with circuit complexity (studied in other papers). Here we prove that $M_{k,1}$ and $Inv_{k,1}$ are congruence-simple for all k. Their Green relations $\mathcal{J}$ and $\mathcal{D}$ are characterized: $M_{k,1}$ and $Inv_{k,1}$ are $\mathcal{J}$-0-simple, and they have $k-1$ non-zero $\mathcal{D}$-classes. They are submonoids of the multiplicative part of the Cuntz algebra $\mathcal{O}_k$. They are finitely generated, and their word problem over any finite generating set is in P. Their word problem is coNP-complete over certain infinite generating sets.

1. Thompson–Higman monoids

Since their introduction by Richard J. Thompson in the mid 1960s [27,24,28], the Thompson groups have had a great impact on infinite group theory. Graham Higman generalized the Thompson groups to an infinite family [18]. These groups and some of their subgroups have appeared in many contexts; see e.g., [10,6,13,8,15,16,7,9,21].

The definition of the Thompson–Higman groups lends itself easily to generalizations to inverse monoids and to more general monoids. These monoids are also generalizations of the finite symmetric monoids (of all functions on a set), and this is related to circuit complexity; more details on the latter connection appear in [2,3,5].

By definition the Thompson–Higman group $G_{k,1}$ consists of all maximally extended isomorphisms between finitely generated essential right ideals of A^*, where A is an alphabet of cardinality k. The multiplication is defined to be composition followed by maximal extension: for any $\varphi, \psi \in G_{k,1}$, we have $\varphi \cdot \psi = \max(\varphi \circ \psi)$. Every element $\varphi \in G_{k,1}$ can also be given by a bijection $\varphi : P \to Q$ where $P, Q \subset A^*$ are two finite maximal prefix codes over A; this bijection can be described concretely by a finite function *table*. Section 1.1 gives all the needed definitions.

It is natural to generalize the maximally extended *isomorphisms* between finitely generated essential right ideals of A^* to *homomorphisms*, and to drop the requirement that the right ideals be essential. This leads to interesting monoids, or inverse monoids, which we call Thompson–Higman monoids. Our generalization of the Thompson–Higman groups to monoids will also generalize the embedding of these groups into the Cuntz algebras [4,25], which provides an additional motivation for our definition. Moreover, since these homomorphisms are close to being arbitrary finite string transformations, there is a connection between these monoids and combinational boolean circuits; the study of the connection between Thompson–Higman groups and circuits was started in [5,3] and will be developed more generally for monoids in [2].

1.1. Definition of the Thompson–Higman groups and monoids

We need some basic definitions, that are similar to the introductory material used for defining the Thompson–Higman groups $G_{k,1}$; we follow [4] (which is similar to [26]). We use a finite alphabet A of cardinality $|A| = k$, and we list its elements

E-mail address: birget@camden.rutgers.edu.

0022-4049/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.jpaa.2008.06.012

as $A = \{a_1, \ldots, a_k\}$. Let A^* denote the set of all finite *words* over A (i.e. all finite sequences of elements of A); this includes the *empty word* ε. The *length* of $w \in A^*$ is denoted by $|w|$; let A^n denote the set of words of length n. For two words $u, v \in A^*$ we denote their *concatenation* by uv or by $u \cdot v$; for sets $B, C \subseteq A^*$ the concatenation is $BC = \{uv : u \in B, v \in C\}$. A *right ideal* of A^* is a subset $R \subseteq A^*$ such that $RA^* \subseteq R$. A generating set of a right ideal R is a set C such that R is the intersection of all right ideals that contain C; equivalently, $R = CA^*$. A right ideal R is called *essential* iff R has a non-empty intersections with every right ideal of A^*. For words $u, v \in A^*$, we say that u is a *prefix* of v iff there exists $z \in A^*$ such that $uz = v$. A *prefix code* is a subset $C \subseteq A^*$ such that no element of C is a prefix of another element of C. A prefix code is *maximal* iff it is not a strict subset of another prefix code. One can prove that a right ideal R has a unique minimal (under inclusion) generating set, and that this minimal generating set is a prefix code; this prefix code is maximal iff R is an essential right ideal.

For right ideals $R' \subseteq R \subseteq A^*$ we say that R' is *essential in* R iff R' intersects all right subideals of R in a non-empty way.

Tree interpretation: The free monoid A^* can be pictured by its right Cayley graph, which is the rooted infinite regular k-ary tree with vertex set A^* and edge set $\{(v, va) : v \in A^*, a \in A\}$. We simply call this the *tree of* A^*. It is a directed tree, with all paths moving away from the root ε (the empty word); by "path" we will always mean a directed path. A word v is a prefix of a word w iff v is is an ancestor of w in the tree. A set P is a prefix code iff no two elements of P are on the same path. A set R is a right ideal iff any path that starts in R has all its vertices in R. A finitely generated right ideal R is essential iff every infinite path of the tree eventually reaches R (and then stays in it from there on). For two finitely generated right ideals $R' \subset R$, R' is essential in R iff any infinite path starting in R eventually reaches R'. In other words for finitely generated right ideals $R' \subseteq R$, R' is essential in R iff R' and R have the same *ends*. For the tree of A^* we consider also the "boundary" A^ω (all infinite words), a.k.a. the set of "ends" of the tree.

A *right ideal homomorphism* of A^* is a total function $\varphi : R_1 \to A^*$ such that R_1 is a right ideal of A^*, and for all $x_1 \in R_1$ and all $w \in A^* : \varphi(x_1 w) = \varphi(x_1)\, w$.

We denote functions as acting on the *left*. For any partial function $f : A^* \to A^*$, let $\mathsf{Dom}(f)$ denote the domain and let $\mathsf{Im}(f)$ denote the image (range) of f. For a right ideal homomorphism $\varphi : R_1 \to A^*$ it is easy to see that $\mathsf{Im}(\varphi)$ is also right ideal of A^*, which is finitely generated (as a right ideal) if $R_1 = \mathsf{Dom}(\varphi)$ is finitely generated.

A right ideal homomorphism $\varphi : R_1 \to R_2$, where $R_1 = \mathsf{Dom}(\varphi)$ and $R_2 = \mathsf{Im}(\varphi)$, can be described by a total surjective function $P_1 \to S_2$, which is a restriction of φ; here P_1 is the prefix code that generates R_1 as a right ideal, and S_2 is a set (not necessarily a prefix code) that generates R_2 as a right ideal; so $R_1 = P_1A^*$ and $R_2 = S_2A^*$. The function $P_1 \to S_2$ is called the *table* of φ. The prefix code P_1 is called the *domain code* of φ and we write $P_1 = \mathsf{domC}(\varphi)$. When S_2 is a prefix code we call S_2 the *image code* of φ and we write $S_2 = \mathsf{imC}(\varphi)$.

An injective right ideal homomorphism is called a right ideal *isomorphism*. A right ideal homomorphism $\varphi : R_1 \to R_2$ is called *total* iff the domain right ideal R_1 is essential. And φ is called *surjective* iff the image right ideal R_2 is essential.

Definition 1.1. An *essential restriction* of a right ideal homomorphism $\varphi : R_1 \to A^*$ is a right ideal homomorphism $\Phi : R_1' \to A^*$ such that R_1' is essential in R_1, and such that for all $x_1' \in R_1' : \varphi(x_1') = \Phi(x_1')$. We say that φ is an *essential extension* of Φ iff Φ is an essential restriction of φ.

Note that if Φ is an essential restriction of φ then $R_2' = \mathsf{Im}(\Phi)$ will automatically be essential in $R_2 = \mathsf{Im}(\varphi)$.

Proposition 1.2. (1) *Let φ, Φ be homomorphisms between finitely generated right ideals of A^*, where $A = \{a_1, \ldots, a_k\}$. Then Φ is an essential restriction of φ iff Φ can be obtained from φ by starting from the table of φ and applying a finite number of* restriction steps *of the following form:*

Replace (x, y) in a table by $\{(xa_1, ya_1), \ldots, (xa_k, ya_k)\}$.

(2) *Every homomorphism between finitely generated right ideals of A^* has a* unique *maximal essential extension.*

Proof. For details, see [1]. This is similar to the analogous properties of the Thompson–Higman group $G_{k,1}$ (given in [4], Lemma 2.2 and Prop. 2.1), going back to Thompson. □

Important remark: As we saw, every right ideal homomorphism can be described by a table $P \to S$ where P is a prefix code and S is a set. But we also have: Every right ideal homomorphism φ has an essential restriction φ' whose table $P' \to Q'$ is such that *both P' and Q' are prefix codes*; moreover, Q' can be chosen to be a subset of A^n for some $n \le \max\{|s| : s \in S\}$.

Example (with alphabet $A = \{a, b\}$): $\begin{pmatrix} a & b \\ a & aa \end{pmatrix}$ has an essential restriction $\begin{pmatrix} aa & ab & b \\ aa & ab & aa \end{pmatrix}$.

Definition 1.3. The Thompson–Higman *partial function monoid* $M_{k,1}$ consists of all maximal essential extensions of homomorphisms between finitely generated right ideals of A^*. The multiplication is composition followed by maximal essential extension.

In the remainder of this Section we prove *associativity* of the multiplication of $M_{k,1}$.

By $\mathsf{riHom}(A^*)$ we denote the monoid of all right ideal homomorphisms between finitely generated right ideals of A^*, with function composition as multiplication. We consider the equivalence relation $\equiv$ defined for $\varphi_1, \varphi_2 \in \mathsf{riHom}(A^*)$ by: $\varphi_1 \equiv \varphi_2$ iff $\max(\varphi_1) = \max(\varphi_2)$.

By existence and uniqueness of the maximal essential extension (Proposition 1.2(2)) each $\equiv$-equivalence class contains exactly one element of $M_{k,1}$. We want to prove:

Proposition 1.4. *The equivalence relation $\equiv$ is a monoid congruence on* $\mathsf{riHom}(A^*)$*, and $M_{k,1}$ is isomorphic (as a monoid) to* $\mathsf{riHom}(A^*)/\equiv$*. Hence, $M_{k,1}$ is associative.*

First, some lemmas (whose proofs are straightforward; see [1] for details).

Lemma 1.5. *If $R_i' \subseteq R_i$ ($i = 1, 2$) are finitely generated right ideals with R_i' essential in R_i, then $R_1' \cap R_2'$ is essential in $R_1 \cap R_2$.* □

Lemma 1.6. *All $\varphi_1, \varphi_2 \in \mathsf{riHom}(A^*)$ have restrictions $\Phi_1, \Phi_2 \in \mathsf{riHom}(A^*)$ (not necessarily essential restrictions) such that:*

- $\mathsf{Dom}(\Phi_2) = \mathsf{Im}(\Phi_1) = \mathsf{Dom}(\varphi_2) \cap \mathsf{Im}(\varphi_1)$, *and*
- $\Phi_2 \circ \Phi_1 = \varphi_2 \circ \varphi_1$. □

Lemma 1.7. *For all $\varphi_1, \varphi_2 \in \mathsf{riHom}(A^*)$: $\mathsf{max}(\varphi_2 \circ \varphi_1) = \mathsf{max}(\mathsf{max}(\varphi_2) \circ \varphi_1) = \mathsf{max}(\varphi_2 \circ \mathsf{max}(\varphi_1))$.*

Proof. We only prove the first equality; the proof of the second one is similar. Let $R' = \mathsf{Dom}(\varphi_2) \cap \mathsf{Im}(\varphi_1)$. By Lemma 1.6 we can restrict φ_1 and φ_2 to φ_1', φ_2', so that $\varphi_2' \circ \varphi_1' = \varphi_2 \circ \varphi_1$, and $\mathsf{Dom}(\varphi_2') = \mathsf{Im}(\varphi_1') = R'$. Similarly, let $R'' = \mathsf{Dom}(\mathsf{max}(\varphi_2)) \cap \mathsf{Im}(\varphi_1)$. We restrict φ_1 and $\mathsf{max}(\varphi_2)$ to φ_1'', respectively φ_2'', so that $\varphi_2'' \circ \varphi_1'' = \mathsf{max}(\varphi_2) \circ \varphi_1$, and $\mathsf{Dom}(\varphi_2'') = \mathsf{Im}(\varphi_1'') = R''$.

Then $R' \subseteq R''$, and R' is essential in R'', by Lemma 1.5. Hence, $\varphi_2 \circ \varphi_1$ is an essential restriction of $\mathsf{max}(\varphi_2) \circ \varphi_1$. Hence by uniqueness of the maximal essential extension, $\mathsf{max}(\mathsf{max}(\varphi_2) \circ \varphi_1) = \mathsf{max}(\varphi_2 \circ \varphi_1)$. □

Proof of Proposition 1.4. If $\varphi_2 \equiv \psi_2$ then by definition, $\mathsf{max}(\varphi_2) = \mathsf{max}(\psi_2)$, hence by Lemma 1.7:

$$\mathsf{max}(\varphi_2 \circ \varphi) = \mathsf{max}(\mathsf{max}(\varphi_2) \circ \varphi) = \mathsf{max}(\mathsf{max}(\psi_2) \circ \varphi) = \mathsf{max}(\psi_2 \circ \varphi),$$

for all $\varphi \in \mathsf{riHom}(A^*)$. Thus (by the definition of $\equiv$), $\varphi_2 \circ \varphi \equiv \psi_2 \circ \varphi$, so $\equiv$ is a right congruence. Similarly one proves that $\equiv$ is a left congruence.

Since every $\equiv$-equivalence class contains exactly one element of $M_{k,1}$ there is a one-to-one correspondence between $\mathsf{riHom}(A^*)/\equiv$ and $M_{k,1}$. Moreover, the map $\varphi \in \mathsf{riHom}(A^*) \longmapsto \mathsf{max}(\varphi) \in M_{k,1}$ is a homomorphism, by Lemma 1.7 and by the definition of multiplication in $M_{k,1}$. Hence $\mathsf{riHom}(A^*)/\equiv$ is isomorphic to $M_{k,1}$. □

1.2. Other Thompson–Higman monoids

We mention a few more families of Thompson–Higman monoids, whose definitions come about naturally in analogy with $M_{k,1}$. The Thompson–Higman *total* function monoid $totM_{k,1}$ and the Thompson–Higman *surjective* function monoid $surM_{k,1}$ consist of maximal essential extensions of homomorphisms between finitely generated right ideals of A^* where the domain, respectively, the image ideal, is an *essential* right ideal.

The Thompson–Higman *inverse* monoid $Inv_{k,1}$ consists of all maximal essential extensions of isomorphisms between finitely generated (not necessarily essential) right ideals of A^*.

Every element $\varphi \in totM_{k,1}$ can be described by a function $P \to Q$, called the *table* of φ, where $P, Q \subset A^*$ with P a finite *maximal* prefix code over A. Similarly, for $surM_{k,1}$ the prefix code Q is maximal. Every $\varphi \in Inv_{k,1}$ can be described by a bijection $P \to Q$ with $P, Q \subset A^*$ finite prefix codes (not necessarily maximal).

It is easy to prove that essential extension and restriction of right ideal homomorphisms, as well as composition of such homomorphisms, preserve injectiveness, totality, and surjectiveness. Thus $totM_{k,1}$, $surM_{k,1}$, and $Inv_{k,1}$ are submonoids of $M_{k,1}$. The monoids $M_{k,1}$, $totM_{k,1}$, and $surM_{k,1}$ are *regular*. (A monoid M is regular iff for every $m \in M$ there exists $x \in M$ such that $mxm = m$.) The monoid $Inv_{k,1}$ is an inverse monoid. (A monoid M is inverse iff for every $m \in M$ there exists one and only one $x \in M$ such that $mxm = m$ and $x = xmx$.)

We consider the submonoids $totInv_{k,1}$ and $surInv_{k,1}$ of $Inv_{k,1}$, described by bijections $P \to Q$ where $P, Q \subset A^*$ are two finite prefix codes with P, respectively Q maximal. The (unique) inverses of elements in $totInv_{k,1}$ are in $surInv_{k,1}$, and vice versa, so these submonoids of $Inv_{k,1}$ are not regular. We have $totInv_{k,1} \cap surInv_{k,1} = G_{k,1}$.

For all $n > 0$, $M_{k,1}$ contains the symmetric monoids PF_{k^n} of all partial functions on k^n elements, represented by all elements of $M_{k,1}$ with a table $P \to Q$ where $P, Q \subseteq A^n$. Hence $M_{k,1}$ contains all finite monoids. And $Inv_{k,1}$ contains $\mathcal{I}_{k^n}$ (the finite symmetric inverse monoid of all injective partial functions on A^n).

1.3. Cuntz algebras and Thompson–Higman monoids

All the monoids, inverse monoids, and groups, defined above, are submonoids of the multiplicative part of the Cuntz algebra $\mathcal{O}_k$.

The algebra $\mathcal{O}_k$, introduced by Dixmier [14] (for $k = 2$) and Cuntz [12], is a k-generated star-algebra (over the field of complex numbers) with identity element **1** and zero **0**, given by the following finite presentation over a generating set

$A = \{a_1, \ldots, a_k\}$. Since this is a star-algebra presentation we automatically have the star-inverses $\{\overline{a}_1, \ldots, \overline{a}_k\}$; for clarity we use overlines rather than stars. The relations of the presentation are:

$$\overline{a}_i a_i = \mathbf{1}, \quad \text{for } i = 1, \ldots, k;$$
$$\overline{a}_i a_j = \mathbf{0}, \quad \text{when } i \neq j,\ 1 \leq i, j \leq k;$$
$$a_1\overline{a}_1 + \cdots + a_k\overline{a}_k = \mathbf{1}.$$

The Cuntz algebras are actually C^*-algebras with many remarkable properties (proved in [12]), but here we only need them as star-algebras (without their norm and Cauchy completion).

In [4] and independently in [25] it was proved that the Thompson–Higman group $G_{k,1}$ is the subgroup of $\mathcal{O}_k$ consisting of the elements that have an expression of the form $\sum_{x \in P} f(x)\overline{x}$ where we require the following: P and Q range over all finite maximal prefix codes over the alphabet $\{a_1, \ldots, a_k\}$, and f is any bijection $P \to Q$. Another proof is given in [20]. More generally we also have:

Theorem 1.8. *The Thompson–Higman monoid $M_{k,1}$ is a submonoid of the multiplicative part of the Cuntz algebra $\mathcal{O}_k$.*

Proof outline. The Thompson–Higman partial function monoid $M_{k,1}$ is the set of all elements of $\mathcal{O}_k$ that have an expression of the form $\sum_{x \in P} f(x)\overline{x}$ where $P \subset A^*$ ranges over all finite prefix codes, and f ranges over functions $P \to A^*$. The details of the proof are very similar to the proofs in [4,25]; the definition of *essential* restriction (and extension) and Proposition 1.2 insure that the same proof goes through. $\square$

The embeddability into the Cuntz algebra is a further justification of the definitional choices that we made for the Thompson–Higman monoid $M_{k,1}$.

2. Structure and simplicity of the Thompson–Higman monoids

We give some structural properties of the Thompson–Higman monoids; in particular, we show that $M_{k,1}$ and $Inv_{k,1}$ are simple for all k.

2.1. Group of units, $\mathcal{J}$-relation, simplicity

By definition, the group of units of a monoid M is the set of invertible elements (i.e. the elements $u \in M$ for which there exists $x \in M$ such that $xu = ux = \mathbf{1}$). The next Proposition is fairly straightforward; for details, see [1].

Proposition 2.1. *The Thompson–Higman group $G_{k,1}$ is the* group of units *of the monoids $M_{k,1}$, $totM_{k,1}$, $surM_{k,1}$, and $Inv_{k,1}$.* $\square$

We will now characterize some of the Green relations of $M_{k,1}$ and of $Inv_{k,1}$, and we prove simplicity. By definition, two elements x, y of a monoid M are *$\mathcal{J}$-related* (denoted $x \equiv_{\mathcal{J}} y$) iff x and y belong to exactly the same ideals of M. The *$\mathcal{J}$-preorder* of M is defined as follows: $x \leq_{\mathcal{J}} y$ iff x belongs to every ideal that y belongs to. Thus, $x \equiv_{\mathcal{J}} y$ iff $x \leq_{\mathcal{J}} y$ and $y \leq_{\mathcal{J}} x$; and $x \leq_{\mathcal{J}} y$ iff there exist $\alpha, \beta \in M$ such that $x = \alpha y \beta$. A monoid M is called *$\mathcal{J}$-simple* iff M has only one $\mathcal{J}$-class. A monoid M is called *0-$\mathcal{J}$-simple* iff M has exactly two $\mathcal{J}$-classes, one of which consist of just a zero element (equivalently, M has only two ideals, one of which is a zero element, and the other is M itself). See [11,17] for more information on the $\mathcal{J}$-relation. Cuntz [12] proved that the multiplicative part of the C^*-algebra $\mathcal{O}_k$ is a 0-$\mathcal{J}$-simple monoid, and that as an algebra $\mathcal{O}_k$ is simple. We will now prove similar results for the Thompson–Higman monoids.

Proposition 2.2. *The inverse monoid $Inv_{k,1}$ and the monoid $M_{k,1}$ are* 0-$\mathcal{J}$*-simple. The monoid $totM_{k,1}$ is $\mathcal{J}$-simple.*

Proof. When $\varphi \in M_{k,1}$ (or $\in Inv_{k,1}$) is not the empty map, there are $x_0, y_0 \in A^*$ such that $y_0 = \varphi(x_0)$. Let $\alpha = \{(\varepsilon \mapsto x_0)\}$ and $\beta = \{(y_0 \mapsto \varepsilon)\}$ (where ε denotes the empty word). Then $\beta\,\varphi\,\alpha(.) = \{(\varepsilon \mapsto \varepsilon)\} = \mathbf{1}$. So, every non-zero element of $M_{k,1}$ (and of $Inv_{k,1}$) is in the same $\mathcal{J}$-class as the identity element. In the case of $totM_{k,1}$ we take $\alpha = \{(\varepsilon \mapsto x_0)\}$ as before, and $\beta' : Q \mapsto \{\varepsilon\}$, where Q is any finite maximal prefix code containing y_0. Then $\beta'\,\varphi\,\alpha(.) = \{(\varepsilon \mapsto \varepsilon)\} = \mathbf{1}$. $\square$

Thompson proved that $V\ (= G_{2,1})$ is a simple group; Higman proved more generally that when k is even then $G_{k,1}$ is simple, and when k is odd then $G_{k,1}$ contains a simple normal subgroup of index 2. We will show next that in the monoid case we have *simplicity for all k*. For a monoid M, "simple", or more precisely, *"congruence-simple"* means that the only congruences on M are the trivial congruences (i.e. the equality relation, and the congruence that lumps all of M into one class).

Theorem 2.3. *The Thompson–Higman monoids $Inv_{k,1}$ and $M_{k,1}$ are congruence-simple for all k.*

Proof. Let $\equiv$ be any congruence that is not the equality relation. We will show that the whole monoid is congruent to the empty map $\mathbf{0}$. If we have $\Phi \equiv \mathbf{0}$, $\Phi \neq \mathbf{0}$ then (by 0-$\mathcal{J}$-simplicity of $M_{k,1}$) we have $M_{k,1} = \{\alpha\,\Phi\,\beta : \alpha, \beta \in M_{k,1}\}$ for every $\Phi \in M_{k,1}$ with $\Phi \neq 0$. Hence, all elements of $M_{k,1}$ are $\equiv \mathbf{0}$. (The same applies to $Inv_{k,1}$.) On the other hand, if $\varphi \equiv \psi$ and $\varphi \neq \psi$ for any $\varphi, \psi \in M_{k,1} - \{\mathbf{0}\}$, then there exist $x_0, y_0, y_1 \in A^*$ such that $\varphi(x_0) = y_0 \neq y_1 = \psi(x_0)$. Then for $\alpha = \{(y_0 \mapsto y_0)\}$, $\beta = \{(x_0 \mapsto x_0)\} \in Inv_{k,1} \subseteq M_{k,1}$ we have $\alpha\,\varphi\,\beta(.) = \{(x_0 \mapsto y_0)\}$, and $\alpha\,\psi\,\beta(.) = \mathbf{0}$. So, $\alpha\,\varphi\,\beta \equiv \alpha\,\psi\,\beta$, $\alpha\,\varphi\,\beta \neq \mathbf{0}$, but $\alpha\,\psi\,\beta = \mathbf{0}$. Hence the previous paragraph, applied to $\Phi = \alpha\,\varphi\,\beta$, implies that the entire monoid is $\equiv \mathbf{0}$. $\square$

2.2. $\mathcal{D}$-relation

Besides the $\mathcal{J}$-relation and the $\mathcal{J}$-preorder, based on ideals, there are the $\mathcal{R}$- and $\mathcal{L}$-relations and $\mathcal{R}$- and $\mathcal{L}$-preorders, based on right (or left) ideals. Two elements $x, y \in M$ are *$\mathcal{R}$-related* (denoted $x \equiv_{\mathcal{R}} y$) iff x and y belong to exactly the same right ideals of M. The *$\mathcal{R}$-preorder* is defined as follows: $x \leq_{\mathcal{R}} y$ iff x belongs to every right ideal that y belongs to. Hence $x \equiv_{\mathcal{R}}$ iff $x \leq_{\mathcal{R}} y$ and $y \leq_{\mathcal{R}} x$; also, $x \leq_{\mathcal{R}} y$ iff there exists $\alpha \in M$ such that $x = y\alpha$. In a similar way one defines $\equiv_{\mathcal{L}}$ and $\leq_{\mathcal{L}}$. Finally, there is the *$\mathcal{D}$-relation* of M, which is defined as follows: $x \equiv_{\mathcal{D}} y$ iff there exists $s \in M$ such that $x \equiv_{\mathcal{R}} s \equiv_{\mathcal{L}} y$; this is equivalent to the existence of $t \in M$ such that $x \equiv_{\mathcal{L}} t \equiv_{\mathcal{R}} y$. For more information on these concepts, see for example [11,17].

The following invariants with respect to essential restriction follow easily from Proposition 1.2:

Proposition 2.4. *Let $\varphi_1 : P_1 \to Q_1$ be a table for an element of $M_{k,1}$, where $P_1, Q_1 \subset A^*$ are finite prefix codes. Let $\varphi_2 : P_2 \to Q_2$ be another finite table for the* same *element of $M_{k,1}$, obtained from the table φ_1 by an essential restriction. Then $P_2, Q_2 \subset A^*$ are finite prefix codes and we have*

$$|P_1| \equiv |P_2| \bmod (k-1),$$
$$|Q_1| \equiv |Q_2| \bmod (k-1).$$

These modular congruences also hold for essential extensions, provided that we only extend to tables in which the image is a prefix code. □

As a consequence of Proposition 2.4 it makes sense to talk about $|\mathrm{domC}(\varphi)|$ and $|\mathrm{imC}(\varphi)|$ as elements of $\mathbb{Z}_{k-1}$, independently of the representation of φ by a right-ideal homomorphism.

Theorem 2.5. *For any non-zero elements φ, ψ of $M_{k,1}$ (or of $Inv_{k,1}$) the $\mathcal{D}$-relation is characterized as follows:*

$$\varphi \equiv_{\mathcal{D}} \psi \quad \text{iff} \quad |\mathrm{imC}(\varphi)| \equiv |\mathrm{imC}(\psi)| \bmod (k-1).$$

Hence, $M_{k,1}$ and $Inv_{k,1}$ have $k-1$ non-zero $\mathcal{D}$-classes. In particular, $M_{2,1}$ and $Inv_{2,1}$ are 0*-$\mathcal{D}$-simple (also called* 0-bisimple*).*

The proof of Theorem 2.5 uses several lemmas.

Lemma 2.6 (*[5] Lemma 6.1; Arxiv version of [5] Lemma 9.9*). *For every finite alphabet A and every integer $i \geq 0$ there exists a maximal prefix code of cardinality $1 + (|A| - 1)i$. And every finite maximal prefix code over A has cardinality $1 + (|A| - 1)i$, for some integer $i \geq 0$.*

It follows that when $|A| = 2$, there are finite prefix codes over A of every finite cardinality. □

As a consequence of this Lemma we have for all $\varphi \in G_{k,1}$: $\|\varphi\| \equiv 1 \bmod (k-1)$. Thus, except for the Thompson group V (when $k = 2$), there is a constraint on the table size of the elements of the group.

In the following, id_Q denotes the element of $Inv_{k,1}$ with table $\{(x \mapsto x) : x \in Q\}$, where $Q \subset A^*$ is any finite prefix code. The following Lemma is not difficult (see [1]).

Lemma 2.7. (1) *For any $\varphi \in M_{k,1}$ (or $\in Inv_{k,1}$) with table $P \to Q$ we have: $\varphi \equiv_{\mathcal{R}} \mathrm{id}_Q$.*
(2) *If S, T are finite prefix codes with $|S| = |T|$ then $\mathrm{id}_S \equiv_{\mathcal{D}} \mathrm{id}_T$.*
(3) *If $\varphi_1 : P_1 \to Q_1$ and $\varphi_2 : P_2 \to Q_2$ are such that $|Q_1| = |Q_2|$ then $\varphi_1 \equiv_{\mathcal{D}} \varphi_2$.* □

Lemma 2.8. (1) *For any $m \geq k$ let i be the residue of m modulo $k-1$ in the range $2 \leq i \leq k$, and let us write $m = i + (k-1)j$, for some $j \geq 0$. Then there exists a prefix code $Q_{i,j}$ of cardinality $|Q_{i,j}| = m$, such that $\mathrm{id}_{Q_{i,j}}$ is an essential restriction of $\mathrm{id}_{\{a_1,\ldots,a_i\}}$. Hence, $\mathrm{id}_{Q_{i,j}} = \mathrm{id}_{\{a_1,\ldots,a_i\}}$ as elements of $Inv_{k,1}$.*

(2) *In $M_{k,1}$ and in $Inv_{k,1}$ we have $\mathrm{id}_{\{a_1\}} \equiv_{\mathcal{D}} \mathrm{id}_{\{a_1,\ldots,a_k\}} = \mathbf{1}$.*

Proof. (1) For any $m \geq k$ there exist $i, j \geq 0$ such that $1 \leq i \leq k$ and $m = i + (k-1)j$. We consider the prefix code

$$Q_{i,j} = \{a_2, \ldots, a_i\} \cup \bigcup_{r=1}^{j-1} a_1^r (A - \{a_1\}) \cup a_1^j A.$$

It is easy to see that $Q_{i,j}$ is a prefix code, which is maximal iff $i = k$; see Fig. 1. Clearly, $|Q_{i,j}| = i+(k-1)j$. Since $Q_{i,j}$ contains $a_1^j A$, we can perform an essential extension of $\mathrm{id}_{Q_{i,j}}$ by replacing the table entries $\{(a_1^j a_1, a_1^j a_1), (a_1^j a_2, a_1^j a_2), \ldots, (a_1^j a_k, a_1^j a_k)\}$ by (a_1^j, a_1^j). This replaces $Q_{i,j}$ by $Q_{i,j-1}$. So, $\mathrm{id}_{Q_{i,j}}$ can be essentially extended to $\mathrm{id}_{Q_{i,j-1}}$. By repeating this we find that $\mathrm{id}_{Q_{i,j}}$ is the same element (in $M_{k,1}$ and in $Inv_{k,1}$) as $\mathrm{id}_{Q_{i,0}} = \mathrm{id}_{\{a_1,\ldots,a_i\}}$.
(2) By essential restriction, $\mathrm{id}_{\{a_1\}} = \mathrm{id}_{\{a_1a_1, a_1a_2, \ldots, a_1a_k\}}$, in $M_{k,1}$ and in $Inv_{k,1}$. And by Lemma 2.7(2), $\mathrm{id}_{\{a_1a_1, a_1a_2, \ldots, a_1a_k\}} \equiv_{\mathcal{D}} \mathrm{id}_{\{a_1,\ldots,a_k\}}$; the latter, by essential extension, is $\mathbf{1}$. □

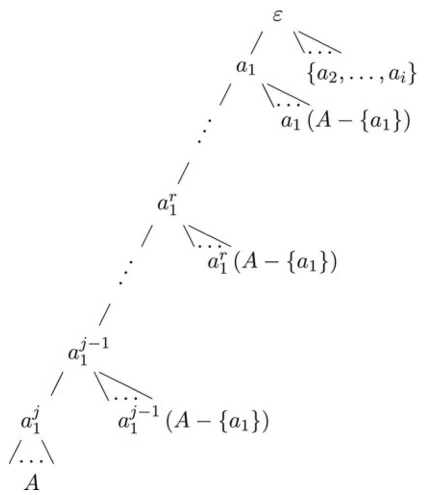

Fig. 1. The prefix tree of $Q_{i,j}$.

Lemma 2.9. *For all $\varphi, \psi \in Inv_{k,1}$: If $\varphi \geq_{\mathcal{L}(M_{k,1})} \psi$ then $\varphi \geq_{\mathcal{L}(I_{k,1})} \psi$ (where $\geq_{\mathcal{L}(M_{k,1})}$ and $\geq_{\mathcal{L}(I_{k,1})}$ are respectively the $\mathcal{L}$-preorder of $M_{k,1}$ and $Inv_{k,1}$).*

The same holds with $\geq_{\mathcal{L}}$ replaced by $\equiv_{\mathcal{L}}$, $\geq_{\mathcal{R}}$, $\equiv_{\mathcal{R}}$, $\equiv_{\mathcal{D}}$, $\geq_{\mathcal{J}}$ and $\equiv_{\mathcal{J}}$.

Proof. If $\psi = \alpha\,\varphi$ for some $\alpha \in M_{k,1}$ then let us define α' by $\alpha' = \alpha \cdot \mathsf{id}_{\mathsf{Im}(\varphi)}$. Then, $\psi\,\varphi^{-1} = \alpha\varphi\varphi^{-1} = \alpha \cdot \mathsf{id}_{\mathsf{Im}(\varphi)} = \alpha'$, hence $\alpha' \in Inv_{k,1}$ (since $\varphi, \psi \in Inv_{k,1}$). Moreover, $\alpha'\,\varphi = \alpha \cdot \mathsf{id}_{\mathsf{Im}(\varphi)}\varphi = \alpha\varphi = \psi$. $\square$

So far we have shown that in $M_{k,1}$ and in $Inv_{k,1}$, every non-zero element is $\equiv_{\mathcal{D}}$ to one of the $k-1$ elements $\mathsf{id}_{\{a_1,\ldots,a_i\}}$, for $i = 1, \ldots, k-1$. Moreover if two elements of $M_{k,1}$ or $Inv_{k,1}$ have tables $\varphi_1 : P_1 \to Q_1$ and $\varphi_2 : P_2 \to Q_2$, then we have: If $|Q_1| \equiv |Q_2| \bmod (k-1)$ then $\varphi_1 \equiv_{\mathcal{D}} \varphi_2$. We still need to prove the converse of this. It is sufficient to prove the converse for $Inv_{k,1}$, by Lemma 2.9 and because every element of $M_{k,1}$ is $\equiv_{\mathcal{D}}$ to an element of $Inv_{k,1}$ (namely $\mathsf{id}_{\{a_1,\ldots,a_i\}}$).

Lemma 2.10. *Let $\varphi, \psi \in Inv_{k,1}$. If $\varphi \equiv_{\mathcal{D}} \psi$ in $Inv_{k,1}$, then $\|\varphi\| \equiv \|\psi\| \bmod (k-1)$.*

Proof. (1) We first prove that if $\varphi \equiv_{\mathcal{L}} \psi$ then $|\mathsf{domC}(\varphi)| \equiv |\mathsf{domC}(\psi)| \bmod (k-1)$.

By definition, $\varphi \equiv_{\mathcal{L}} \psi$ iff $\varphi = \beta\,\psi$ and $\psi = \alpha\,\varphi$ for some $\alpha, \beta \in Inv_{k,1}$. By Lemma 1.6 there are restrictions β' and ψ' of β, respectively ψ, and an essential restriction Φ of φ such that:

$$\Phi = \beta' \circ \psi', \quad \text{and} \quad \mathsf{Dom}(\beta') = \mathsf{Im}(\psi').$$

It follows that $\mathsf{Dom}(\Phi) \subseteq \mathsf{Dom}(\psi')$, since if $\psi'(x)$ is not defined then $\Phi(x) = \beta' \circ \psi'(x)$ is not defined either. Similarly, there is an essential restriction Ψ of ψ and a restriction φ' of φ and such that $\mathsf{Dom}(\Psi) \subseteq \mathsf{Dom}(\varphi')$.

Thus, the restriction of both φ and ψ to the intersection $\mathsf{Dom}(\Phi) \cap \mathsf{Dom}(\Psi)$ yields restrictions φ'', respectively ψ'' such that $\mathsf{Dom}(\varphi'') = \mathsf{Dom}(\psi'')$.

Claim: φ'' and ψ'' are essential restrictions of φ, respectively ψ.

Indeed, every right ideal R of A^* that intersects $\mathsf{Dom}(\psi)$ also intersects $\mathsf{Dom}(\Psi)$ (since Ψ is an essential restriction of ψ). Since $\mathsf{Dom}(\Psi) \subseteq \mathsf{Dom}(\varphi') \subseteq \mathsf{Dom}(\varphi)$, R also intersects $\mathsf{Dom}(\varphi)$. Moreover, since Φ is an essential restriction of φ, R also intersects $\mathsf{Dom}(\Phi)$. Thus, $\mathsf{Dom}(\Phi)$ is essential in $\mathsf{Dom}(\psi)$. Since $\mathsf{Dom}(\Psi)$ is also essential in $\mathsf{Dom}(\psi)$, $\mathsf{Dom}(\Phi) \cap \mathsf{Dom}(\Psi)$ is essential in $\mathsf{Dom}(\psi)$; indeed, in general, the intersection of two right ideals R_1, R_2 that are essential in a right ideal R_3, is essential in R_3 (this is a special case of Lemma 1.5). This means that ψ'' is an essential restriction of ψ. Similarly, φ'' is an essential restriction of φ. [This proves the Claim.]

So, φ'' and ψ'' are essential restrictions such that $\mathsf{Dom}(\varphi'') = \mathsf{Dom}(\psi'')$. Hence, $\mathsf{domC}(\varphi'') = \mathsf{domC}(\psi'')$; Proposition 2.4 then implies that $|\mathsf{domC}(\varphi)| \equiv |\mathsf{domC}(\varphi'')| = |\mathsf{domC}(\psi'')| \equiv |\mathsf{domC}(\psi)| \bmod (k-1)$.

(2) Next, let us prove that if $\varphi \equiv_{\mathcal{R}} \psi$ then $|\mathsf{imC}(\varphi)| \equiv |\mathsf{imC}(\psi)| \bmod (k-1)$. This follows from (1) since $\mathsf{imC}(\varphi) = \mathsf{domC}(\varphi^{-1})$, and since in $Inv_{k,1}$ we have $\varphi \equiv_{\mathcal{R}} \psi$ iff $\varphi^{-1} \equiv_{\mathcal{L}} \psi^{-1}$.

The lemma now follows from (1) and (2), since for elements of $Inv_{k,1}$, $|\mathsf{imC}(\varphi)| = |\mathsf{domC}(\varphi)| = \|\varphi\|$, and since the D-relation is the composite of the $\mathcal{L}$-relation and the $\mathcal{R}$-relation. $\square$

Proof of Theorem 2.5. We saw already that for $\varphi_1 : P_1 \to Q_1$ and $\varphi_2 : P_2 \to Q_2$ we have: If $|Q_1| \equiv |Q_2| \bmod (k-1)$ then $\varphi_1 \equiv_{\mathcal{D}} \varphi_2$. In particular, when $|Q_1| \equiv i \bmod (k-1)$ then $\varphi_1 \equiv_{\mathcal{D}} \mathsf{id}_{\{a_1,\ldots,a_i\}}$. It follows from Lemma 2.10 that the elements $\mathsf{id}_{\{a_1,\ldots,a_i\}}$ (for $i = 1, \ldots, k-1$) are all in different $\mathcal{D}$-classes. $\square$

We have characterized the $\mathcal{D}$- and $\mathcal{J}$-relations of $M_{k,1}$ and $Inv_{k,1}$. We leave the general study of the Green relations of $M_{k,1}$, $Inv_{k,1}$, and the other Thompson–Higman monoids for future work. The main result of this paper, to be proved next, is that the Thompson–Higman monoids $M_{k,1}$ and $Inv_{k,1}$ are finitely generated and that their word problem over any finite generating set is in P.

3. Finite generating sets

We will show that $Inv_{k,1}$ and $M_{k,1}$ are finitely generated. An application of the latter fact is that a finite generating set of $M_{k,1}$ can be used to build combinational circuits for finite boolean functions that do not have fixed-length inputs or outputs. In engineering, non-fixed length inputs or outputs make sense, for example, if the inputs or outputs are handled sequentially, and if the possible input strings come form a prefix code.

First, we need some more definitions about prefix codes. The *prefix tree* of a prefix code $P \subset A^*$ is, by definition, a tree whose vertex set is the set of all the prefixes of the elements of P, and whose edge set is $\{(x, xa) : a \in A, \alpha \text{ is a prefix of some element of } P\}$. The tree is rooted, with root ε (the empty word). Thus, the prefix tree of P is a subtree of the tree of A^*. The set of leaves of the prefix tree of P is P itself. The vertices that are not leaves are called *internal vertices*. We will say more briefly an "internal vertex of P" instead of internal vertex of the prefix tree of P. An internal vertex has between 1 and k children; an internal vertex is called *saturated* iff it has k children.

One can prove easily that a prefix code P is maximal iff every internal vertex of the prefix tree of P is saturated. Hence, every prefix code P can be embedded in a maximal prefix code (which is finite when P is finite), obtained by saturating the prefix tree of P. Moreover we have the following three Lemmas (whose proofs are straightforward, see [1]).

Lemma 3.1. *For any two finite non-maximal prefix codes $P_1, P_2 \subset A^*$ there are finite maximal prefix codes $P_1', P_2' \subset A^*$ such that $P_1 \subset P_1'$, $P_2 \subset P_2'$, and $|P_1'| = |P_2'|$.* □

Lemma 3.2. *Let P and Q be finite prefix codes of A^* with $|P| = |Q|$. If P and Q are both maximal prefix codes, or if both are non-maximal, then there is an element of $G_{k,1}$ that maps P onto Q. On the other hand, if one of P and Q is maximal and the other one is not maximal, then there is no element of $G_{k,1}$ that maps P onto Q.* □

Notation: For $u, v \in A^*$, the element of $Inv_{k,1}$ with one-element domain code $\{u\}$ and one-element image code $\{v\}$ is denoted by $(u \mapsto v)$. When $(u \mapsto v)$ is composed with itself j times the resulting element of $Inv_{k,1}$ is denoted by $(u \mapsto v)^j$.

Lemma 3.3. (1) *For all $j > 0 : (a_1 \mapsto a_1a_1)^j = (a_1 \mapsto a_1^{j+1})$.*
(2) *Let $S = \{a_1^j a_1, a_1^j a_2, \ldots, a_1^j a_i\}$, for some $1 \le i \le k-1$, $0 \le j$. Then id_S is generated by the $k+1$ elements $\{(a_1 \mapsto a_1a_1), (a_1a_1 \mapsto a_1)\} \cup \{\mathsf{id}_{\{a_1a_1, a_1a_2, \ldots, a_1a_i\}} : 1 \le i \le k-1\}$.*
(3) *For all $j \ge 2 : (\varepsilon \mapsto a_1^j)(.) = (a_1 \mapsto a_1a_1)^{j-1} \cdot (\varepsilon \mapsto a_1)(.)$.* □

Theorem 3.4. *The inverse monoid $Inv_{k,1}$ is finitely generated.*

Proof. We will use the fact that the Thompson–Higman group $G_{k,1}$ is finitely generated. In particular, if $\varphi \in Inv_{k,1}$, $g_1, g_2 \in G_{k,1}$, and if $g_2\varphi g_1$ can be expressed as a product p over a fixed finite set of elements of $Inv_{k,1}$, then it follows that $\varphi = g_2^{-1} p g_1^{-1}$ can also be expressed as a product over a fixed finite set of elements of $Inv_{k,1}$.

For $\varphi \in Inv_{k,1}$ with $\mathsf{domC}(\varphi) = P$ and $\mathsf{imC}(\varphi) = Q$ we distinguish four cases, depending on the maximality or non-maximality of P and Q.
(1) If P and Q are both maximal prefix codes then $\varphi \in G_{k,1}$, and we can express φ over a finite generating set of $G_{k,1}$.
(2) Assume P and Q are both non-maximal prefix codes. By Lemma 3.1 there are finite maximal prefix codes P', Q' such that $P \subset P'$, $Q \subset Q'$, and $|P'| = |Q'|$; and by Lemma 2.6, $|P'| = |Q'| = 1 + (k-1)N$ for some $N \ge 0$. Consider the following maximal prefix code C, of cardinality $|P'| = |Q'| = 1 + (k-1)N$:

$$C = \bigcup_{r=0}^{N-2} a_1^r (A - \{a_1\}) \cup a_1^{N-1} A.$$

So C is just the finite maximal prefix code $Q_{i,j}$ when $i = k$ and $j = N-1$ (introduced in the proof of Lemma 2.8, Fig. 1). The elements $g_1 : C \to P'$ and $g_2 : Q' \to C$ of $G_{k,1}$ can be chosen so that $\psi = g_2\varphi g_1(.)$ is a partial identity with $\mathsf{domC}(\psi) = \mathsf{imC}(\psi)$ consisting of the $|P|$ first elements of C in the dictionary order. So, ψ is the identity map restricted to these $|P|$ first elements of C, and ψ is undefined on the rest of C. To describe $\mathsf{domC}(\psi) = \mathsf{imC}(\psi)$ in more detail, let us write $|P| = i + (k-1)\ell$, for some i, ℓ with $1 \le i < k$ and $0 \le \ell \le N-1$. Then

$$\mathsf{domC}(\psi) = \mathsf{imC}(\psi) = a_1^{N-1} A \cup \bigcup_{r=j+1}^{N-2} a_1^r (A - \{a_1\}) \cup a_1^j \{a_2, \ldots, a_i\}$$

where $j = N - 1 - \ell$. Since $\psi = \mathrm{id}_{\mathrm{domC}(\psi)}$, we claim that by essential maximal extension,

$$\psi = \mathrm{id}_S \quad \text{(as elements of } Inv_{k,1}\text{), where } S = \{a_1^j a_1, a_1^j a_2, \ldots, a_1^j a_i\},$$

with i, j as in the description of $\mathrm{domC}(\psi) = \mathrm{imC}(\psi)$ above, i.e. $1 < i < k$, $N-1 \geq j = N-1-\ell \geq 0$, and $|P| = i+(k-1)\ell$. Indeed, if $|P| < k$ then S is just $\mathrm{domC}(\psi)$, with $i = |P|$, and $\ell = 0$ (hence $j = N-1$). If $|P| \geq k$ then the maximum essential extension of ψ will replace the $1 + (k-1)\ell$ elements $a_1^{N-1}A \cup \bigcup_{r=N-j+1}^{N-2} a_1^r(A - \{a_1\})$ by the single element $a_1^{N-\ell+1} = a_1^{j+1}$. What remains is the set

$$S = \{a_1^{j+1}\} \cup a_1^j\{a_2, \ldots, a_i\}.$$

Finally, by Lemma 3.3, id_S (where $S = \{a_1^j a_1, a_1^j a_2, \ldots, a_1^j a_i\}$) can be generated by the $k+1$ elements $\{(a_1 \mapsto a_1a_1), (a_1a_1 \mapsto a_1)\} \cup \{\mathrm{id}_{\{a_1a_1, a_1a_2, \ldots, a_1a_i\}} : 1 \leq i \leq k-1\}$.

(3) Assume P is a maximal prefix code and Q is non-maximal. Let Q' be the finite maximal prefix code obtained by saturating the prefix tree of Q. Then $Q \subset Q'$, $|Q'| = 1 + (k-1)N'$, and $|P| = 1 + (k-1)N$ for some $N' > N \geq 0$. We consider the maximal prefix codes C and C' as defined in the proof of (2), using N' for defining C'. We can choose $g_1 : C \to P$ and $g_2 : Q' \to C'$ in $G_{k,1}$ so that $\psi = g_2\varphi g_1(.)$ is the dictionary-order preserving map that maps C to the first $|C|$ elements of C'. So we have

$$\mathrm{domC}(\psi) = C, \quad \text{and}$$
$$\mathrm{imC}(\psi) = S_0, \quad \text{where } S_0 \subset C' \text{ consist of the } |C| \text{ first elements of } C', \text{ in dictionary order.}$$

Since $|C| = 1 + (k-1)N$, we can describe S_0 in more detail by

$$S_0 = \bigcup_{r=N'-N}^{N'-2} a_1^r(A - \{a_1\}) \cup a_1^{N'-1}A.$$

Next, by essential maximal extension we obtain $\psi = (\varepsilon \mapsto a_1^{N'-N})$. Indeed, we saw that $|P| = 1 + (k-1)N$. If $|P| = 1$ then $P = \{\varepsilon\}$, and $\psi = (\varepsilon \mapsto a_1^{N'})$. If $|P| \geq k$ then maximum essential extension of ψ will replace all the elements of C by the single element ε, and it will replace all the elements of S_0 by the single element $a_1^{N'-N}$.

Finally, by Lemma 3.3, $(\varepsilon \mapsto a_1^{N'-N})$ is generated by the two elements $(\varepsilon \mapsto a_1)$ and $(a_1 \mapsto a_1a_1)$.

(4) The case where P is a non-maximal maximal prefix code and Q is maximal can be derived from case (3) by taking the inverses of the elements from case (3). $\square$

Theorem 3.5. *The monoid $M_{k,1}$ is finitely generated.*

Proof. Let $\varphi : P \to Q$ be the table of any element of $M_{k,1}$. The map of the table itself is total and surjective. So if $|P| = |Q|$ (and in particular, if φ is the empty map) then $\varphi \in Inv_{k,1}$, hence φ can be expressed over the finite generating set of $Inv_{k,1}$. In the rest of the proof we assume $|P| > |Q|$. The main observation is the following.

Claim. φ can be written as the composition of finitely many elements $\varphi_i \in M_{k,1}$ with tables $P_i \to Q_i$ such that $0 \leq |P_i| - |Q_i| \leq 1$. (This follows by induction on $|P| - |Q|$; see [1].)

By the Claim we only need to consider elements $\varphi \in M_{k,1}$ with tables $P \to Q$ such that $|P| = |Q| + 1$. We denote $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_{n-1}\}$, with $\varphi(p_j) = q_j$ for $1 \leq j \leq n-1$, and $\varphi(p_{n-1}) = \varphi(p_n) = q_{n-1}$. We define the following prefix code C with $|C| = |P|$:

- if $|P| = i \leq k$ then $C = \{a_1, \ldots, a_i\}$; note that $i \geq 2$, since $|P| > |Q| > 0$;
- if $|P| > k$ then $C = \{a_2, \ldots, a_i\} \cup \bigcup_{r=1}^{j-1} a_1^r(A - \{a_1\}) \cup a_1^j A$,

where i, j are such that $|P| = i + (k-1)j$, $2 \leq i \leq k$, and $1 \leq j$ (see Fig. 1). Let us write C in increasing dictionary order as $C = \{c_1, \ldots, c_n\}$. The last element of C in the dictionary order is thus $c_n = a_i$.

We now write $\varphi(.) = \psi_3\,\psi_2\,\psi_1(.)$, where ψ_1, ψ_2, ψ_3 are as follows:

- $\psi_1 : P \longrightarrow C$ is bijective and is defined by $p_j \mapsto c_j$ for $1 \leq j \leq n$;
- $\psi_2 : C \longrightarrow C - \{a_i\}$ is the identity map on $\{c_1, \ldots, c_{n-1}\}$, and $\psi_2(c_n) = c_{n-1}$.
- $\psi_3 : C - \{a_i\} \longrightarrow Q$ is bijective and is defined by $c_j \mapsto q_j$ for $1 \leq j \leq n-1$.

It follows that ψ_1 and ψ_3 can be expressed over the finite generating set of $Inv_{k,1}$. On the other hand, ψ_2 has a maximum essential extension, as follows.

- If $2 \leq |P| = i \leq k$ then

$$\psi_2 = \left(\begin{array}{c|c|c|c|c} a_1 & \ldots & a_{i-2} & a_{i-1} & a_i \\ a_1 & \ldots & a_{i-2} & a_{i-1} & a_{i-1} \end{array}\right) = \left(\begin{array}{c|c} \mathrm{id}_{\{a_1,\ldots,a_{i-1}\}} & a_i \\ & a_{i-1} \end{array}\right).$$

- If $|P| = i + (k-1)j > k$ and if $i > 2$ then, after maximal essential extension, ψ_2 also becomes

$$\max(\psi_2) = \left(\begin{array}{c|c} \mathrm{id}_{\{a_1,\ldots,a_{i-1}\}} & \begin{array}{c} a_i \\ a_{i-1} \end{array} \end{array}\right).$$

- If $|P| = i + (k-1)j > k$ and if $i = 2$ then, after essential extensions,

$$\max(\psi_2) = \left(\begin{array}{c|c|c|c|c|c} a_1a_1 & \ldots & a_1a_{k-2} & a_1a_{k-1} & a_1a_k & a_2 \\ a_1a_1 & \ldots & a_1a_{k-2} & a_1a_{k-1} & a_1a_k & a_1a_k \end{array}\right) = \left(\begin{array}{c|c} \mathrm{id}_{a_1A} & \begin{array}{c} a_2 \\ a_1a_k \end{array} \end{array}\right) = \left(\begin{array}{c|c} a_1 & a_2 \\ a_1 & a_1a_k \end{array}\right).$$

In summary, we have factored φ over a finite set of generators of $Inv_{k,1}$ and k additional generators in $M_{k,1}$. $\square$

Note that the proofs of Theorems 3.4 and 3.5 are constructive; they provide factorization algorithms.

In [18] (p. 49) Higman introduces a four-element generating set for $G_{2,1}$; a special property of these generators is that their domain codes and their image codes only contain words of length ≤ 2, and that $||\gamma(x)| - |x|| \le 1$ for every generator γ and every $x \in \mathrm{domC}(\gamma)$. The generators in the finite generating set of $M_{k,1}$ that we introduced above also have those properties. Thus we obtain:

Corollary 3.6. *The monoid $M_{2,1}$ has a finite generating set such that all the generators have the following property: The domain codes and the image codes only contain words of length ≤ 2, and $||\gamma(x)| - |x|| \le 1$ for every generator γ and every $x \in \mathrm{domC}(\gamma)$.* $\square$

The following question remains open: Are $Inv_{k,1}$ and $M_{k,1}$ finitely **presented**?

4. The word problem of the Thompson–Higman monoids

We will show that the word problem of $M_{k,1}$ over any finite generating set can be decided in deterministic polynomial time, i.e. belongs to the complexity class P. It follows immediately that all finitely generated submonoids of $M_{k,1}$ have their word problem (over any finite generating set) in P.

In [4] it was shown that the word problem of the Thompson–Higman group $G_{k,1}$ over any finite generating set is in P. In fact, it is in the parallel complexity class AC_1 [4], and it is co-context-free [22]. In [5] it was shown that the word problem of the Thompson–Higman group $G_{k,1}$ over the infinite generating set $\Gamma_{k,1} \cup \{\tau_{i,i+1} : i > 0\}$ is coNP-complete, where $\Gamma_{k,1}$ is any finite generating set of $G_{k,1}$; the position transposition $\tau_{i,i+1} \in G_{k,1}$ has $\mathrm{domC}(\tau_{i,i+1}) = \mathrm{imC}(\tau_{i,i+1}) = A^{i+1}$, and is defined by $u\alpha\beta \mapsto u\beta\alpha$ for all letters $\alpha, \beta \in A$ and all words $u \in A^{i-1}$. We will see below that the word problem of $M_{k,1}$ over $\Gamma_{k,1} \cup \{\tau_{i,i+1} : i > 0\}$ is also coNP-complete, where $\Gamma_{k,1}$ is any finite generating set of $M_{k,1}$.

4.1. The image code formula

Our proof in [4] that the word problem of $G_{k,1}$ (over any finite generating set) is in P was based on the *table size formula*:

$$\forall \varphi, \psi \in G_{k,1} : \|\psi \circ \varphi\| \le \|\psi\| + \|\varphi\|.$$

Here $\|\varphi\|$ denotes the table size of φ, i.e. the cardinality of $\mathrm{domC}(\varphi)$ (Proposition 3.5, Theorem 3.8, and Proposition 4.2 in [4]). In $M_{k,1}$ the above formula does not hold in general, as the following example shows.

Proposition 4.1. *For every $n > 0$ there exists $\Phi_n = \varphi_2^{n-1}\varphi_1 \in M_{2,1}$ (for some $\varphi_1, \varphi_2 \in M_{2,1}$) with the following properties:*

The table sizes are $\|\Phi_n\| = 2^n$, and $\|\varphi_2\| = \|\varphi_1\| = 2$. So, $\|\Phi_n\|$ is exponentially larger than $(n-1) \cdot \|\varphi_2\| + \|\varphi_1\|$. Hence the table size formula does not hold in $M_{2,1}$.

The word lengths of φ_1, φ_2, and Φ_n (over the finite generating set of $M_{2,1}$ from the previous Section) satisfy $|\varphi_1| = 1$, $|\varphi_2| \le 2$, and $|\Phi_n| < 2n$. So the table size of Φ_n is exponentially larger than its word length: $\|\Phi_n\| > 2^{|\Phi_n|/2}$.

Proof. Consider $\varphi_1, \varphi_2 \in M_{2,1}$ given by the tables $\varphi_1 = \{(0 \mapsto 0), (1 \mapsto 0)\}$, and $\varphi_2 = \{(00 \mapsto 0), (01 \mapsto 0)\}$. One verifies that $\Phi_n = \varphi_2^{n-1} \circ \varphi_1(.)$ sends every bitstring of length n to the word 0; its domain code is $\{0, 1\}^n$, its image code is $\{0\}$, and it is its maximum essential extension. Thus, $\|\varphi_2^{n-1} \circ \varphi_1\| = 2^n$, whereas $(n-1) \cdot \|\varphi_2\| + \|\varphi_1\| = 2 \cdot n$. Also, $\varphi_2(.) = (0 \mapsto 0, 1 \mapsto 0) \cdot (0 \mapsto \varepsilon)$, so $|\varphi_1| = 1$, $|\varphi_2| \le 2$, and $|\Phi_n| \le 2n - 1$. $\square$

Definition 4.2. The *table size* of the right-ideal homomorphism $\theta : PA^* \to QA^*$ where $P, Q \subset A^*$ are prefix codes, is by definition $\|\theta\| = |P|$.

The length of the longest word in the table $P \to Q$ of θ is denoted by $\ell(\theta)$; more precisely, $\ell(\theta) = \max\{|s| : s \in \mathrm{domC}(\theta) \cup \mathrm{imC}(\theta)\}$.

For any finite prefix code $Q \subseteq A^*$ we denote the length of the longest word in Q by $\ell(Q)$.

We will use the following facts that are easy to prove. If $R \subset A^*$ is a right ideal and φ is a right-ideal homomorphism then $\varphi(R)$ and $\varphi^{-1}(R)$ are right ideals. And (Lemma 3.3 of [4]): *If $P, Q, S \subseteq A^*$ are such that $PA^* \cap QA^* = SA^*$, and if S is a prefix code then $S \subseteq P \cup Q$.*

We need two Lemmas (whose proofs are straightforward, see [1]).

Lemma 4.3. *Assume $\theta : PA^* \to QA^*$ is a right-ideal homomorphism, and assume $SA^* \subseteq PA^*$, where $P, Q, S \subset A^*$ are finite prefix codes. Then there is a finite prefix code $R \subset A^*$ such that $\theta(SA^*) = RA^*$ and $R \subseteq \theta(S)$.* $\square$

Lemma 4.4. *Let θ be a right-ideal homomorphism with image $\mathsf{Im}(\theta) = QA^*$ such that $Q \subset A^*$ is a prefix code. Then $\theta^{-1}(Q)$ is a prefix code, and $\mathsf{domC}(\theta) = \theta^{-1}(Q)$.* $\square$

The next theorem is a useful generalization of the table size formula of $G_{k,1}$ to the monoid $M_{k,1}$.

Theorem 4.5 (*Image Code Formula*). *Let $\varphi_1 : P_1A^* \to Q_1A^*$ and $\varphi_2 : P_2A^* \to Q_2A^*$ be right-ideal homomorphisms, where $P_1, P_2, Q_1, Q_2 \subset A^*$ are finite prefix codes. Then*
(1) $|\mathsf{imC}(\varphi_2 \circ \varphi_1)| \le |\mathsf{imC}(\varphi_2)| + |\mathsf{imC}(\varphi_1)|$,
(2) $\ell(\varphi_2 \circ \varphi_1) \le \ell(\varphi_2) + \ell(\varphi_1)$.

Proof. (1) We generalize the proof of Proposition 3.5 in [4]. We have $\mathsf{Dom}(\varphi_2 \circ \varphi_1) = \varphi_1^{-1}(Q_1A^* \cap P_2A^*)$ and $\mathsf{Im}(\varphi_2 \circ \varphi_1) = \varphi_2(Q_1A^* \cap P_2A^*)$. So the following maps are total and onto, on the indicated sets:

$$\varphi_1^{-1}(Q_1A^* \cap P_2A^*) \xrightarrow{\varphi_1} Q_1A^* \cap P_2A^* \xrightarrow{\varphi_2} \varphi_2(Q_1A^* \cap P_2A^*).$$

By Lemma 3.3 of [4] (quoted above) we have $Q_1A^* \cap P_2A^* = SA^*$ for some finite prefix code S with $S \subseteq Q_1 \cup P_2$. Moreover, by Lemma 4.3 we have $\varphi_2(SA^*) = R_2A^*$ for some finite prefix code R_2 such that $R_2 \subseteq \varphi_2(S)$. Now, since $S \subseteq Q_1 \cup P_2$ we have $R_2 \subseteq \varphi_2(S) \subseteq \varphi_2(Q_1) \cup \varphi_2(P_2) = \varphi_2(Q_1) \cup Q_2$. Thus, $|\mathsf{imC}(\varphi_2 \circ \varphi_1)| = |R_2| \le |\varphi_2(Q_1)| + |Q_2| \le |Q_1| + |Q_2|$.
(2.a) Let us first look at $\mathsf{imC}(\varphi_2 \circ \varphi_1)$. We saw above that $\mathsf{imC}(\varphi_2 \circ \varphi_1) = R_2 \subseteq \varphi_2(Q_1) \cup Q_2$. The longest words in Q_2 are of length $\le \ell(\varphi_2)$ $(\le \ell(\varphi_2) + \ell(\varphi_1))$.

And for a longest word y in $\varphi_2(Q_1)$ we have $y = \varphi_2(q_1)$ for some $q_1 \in Q_1 \cap P_2A^*$ (we have $q_1 \in P_2A^*$ since φ_2 is defined on q_1). Thus, $q_1 = p_2w$ for some $p_2 \in P_2, w \in A^*$, hence $|w| \le |q_1|$. Now $y = \varphi_2(p_2w) = \varphi_2(p_2)\, w$, hence $|y| = |\varphi_2(p_2)| + |w| \le \ell(\varphi_2) + |q_1| \le \ell(\varphi_2) + \ell(\varphi_1)$.
(2.b) Let us now look at $\mathsf{domC}(\varphi_2 \circ \varphi_1)$. We saw above that $\mathsf{Dom}(\varphi_2 \circ \varphi_1) = \varphi_1^{-1}(SA^*)$, where $S \subseteq Q_1 \cup P_2$. By Lemma 4.4, $\mathsf{domC}(\varphi_2 \circ \varphi_1) = \varphi_1^{-1}(S)$. Hence, $\mathsf{domC}(\varphi_2 \circ \varphi_1) = \varphi_1^{-1}(S) \subseteq \varphi_1^{-1}(Q_1) \cup \varphi_1^{-1}(P_2) \subseteq P_1 \cup \varphi_1^{-1}(P_2)$. Let us consider $x \in P_1 \cup \varphi_1^{-1}(P_2)$.

For $x \in P_1$ we obviously have $|x| \le \ell(\varphi_1)$ $(\le \ell(\varphi_2) + \ell(\varphi_1))$.

On the other hand, for a longest word x in $\varphi_1^{-1}(P_2)$ we have $x \in P_1A^*$, since φ_1 is defined everywhere on $\varphi_1^{-1}(P_2)$. Therefore, $x = p_1w$ for some $p_1 \in P_1, w \in A^*$. So, $\varphi_1(x) = \varphi_1(p_1)\, w$, hence $|w| \le |\varphi_1(x)|$; and since $\varphi_1(x) \in P_2$ we have $|\varphi_1(x)| \le \ell(\varphi_2)$; so, $|w| \le \ell(\varphi_2)$. Thus, $|x| = |p_1| + |w| \le \ell(\varphi_1) + \ell(\varphi_2)$. $\square$

For elements of $Inv_{k,1}$ the image code has the same size as the domain code, which is also the table size. Thus Theorem 4.5 implies:

Corollary 4.6. *For all $\varphi, \psi \in Inv_{k,1}$: $\|\psi \circ \varphi\| \le \|\psi\| + \|\varphi\|$.* $\square$

Another immediate consequence of Theorem 4.5 is the following.

Corollary 4.7. *Let $\varphi_i : P_iA^* \to Q_iA^*$ be right-ideal homomorphisms for $i = 1, \ldots, n$, where $P_i, Q_i \subset A^*$ are finite prefix codes. Let c_1, c_2 be positive constants.*
(1) If $|Q_i| \le c_1$ for all i then $|\mathsf{imC}(\varphi_n \circ \cdots \circ \varphi_1)| \le c_1n$.
(2) If $\ell(\varphi_i) \le c_2$ for all i then $\ell(\varphi_n \circ \cdots \circ \varphi_1)| \le c_2n$.

So, if $|Q_i| \le c_1$ and $\ell(\varphi_i) \le c_2$ for all i then $\mathsf{imC}(\varphi_n \circ \cdots \circ \varphi_1)$ consists of a linearly bounded number ($\le c_1n$) of words, each of linearly bounded length ($\le c_2n$). $\square$

Since $|\mathsf{imC}(\tau_{i,j})| = k^j$, this implies:

Corollary 4.8. *The word-length of the position transposition $\tau_{i,j}$ over any finite generating set of $M_{k,1}$ is exponential.* $\square$

4.2. Some algorithmic problems about right-ideal homomorphisms

We consider several problems about right-ideal homomorphisms of A^* and show that they have deterministic polynomial-time algorithms. We also show that the word problem of $M_{k,1}$ over $\Gamma_{k,1} \cup \{\tau_{i,i+1} : 0 < i\}$ is coNP-complete, where $\Gamma_{k,1}$ is any finite generating set of $M_{k,1}$. This will help us with the complexity analysis of the word problem of the Thompson–Higman monoids $M_{k,1}$, and provide other corollaries of independent interest.

Lemma 4.9. *There are deterministic polynomial time algorithms for the following problems.*
Input*: Two finite prefix codes $P_1, P_2 \subset A^*$, given explicitly by lists of words;*
Output 1*: The finite prefix code $\Pi \subset A^*$ such that $\Pi A^* = P_1A^* \cap P_2A^*$, where Π is produced explicitly as a list of words.*
Question 2*: Is $P_1A^* \cap P_2A^*$ essential in P_1A^* (or in P_2A^*, or in both)?*

Proof. We saw already that Π exists and $\Pi \subseteq P_1 \cup P_2$ (see Lemmas A.1 and 3.3 of [4]).
Algorithm for Output 1: Since we know that $\Pi \subseteq P_1 \cup P_2$, we just need to search for the elements of Π within $P_1 \cup P_2$. For each $x \in P_1$ we check whether x also belongs to P_2A^* (by checking whether any element of P_2 is a prefix of x). Since P_1 and P_2 are explicitly given as lists, this takes polynomial time. Similarly, for each $x \in P_2$ we check whether x also belongs to P_1A^*. Thus, we have computed the set $\Pi_1 = (P_1 \cap P_2A^*) \cup (P_2 \cap P_1A^*)$. Now, Π is obtained from Π_1 by eliminating every word that has another word of Π_1 as a prefix. Since Π_1 is explicitly listed, this takes just polynomial time.
Algorithm for Question 2: We first compute Π by the previous algorithm. Next, we check whether every $p_1 \in P_1$ and every $p_2 \in P_2$ is a prefix of some $r \in \Pi$; since P_1, P_2, and Π are explicitly listed, this takes just polynomial time. $\square$

Notation: We denote the unique prefix code that generates a right ideal $R \subseteq A^*$ by $\mathsf{prefC}(R)$. We observe that if $\varphi_1 : P_1A^* \to Q_1A^*$ and $\varphi_2 : P_2A^* \to Q_2A^*$ are right-ideal homomorphisms, where $P_1, Q_1, P_2, Q_2 \subset A^*$ are finite prefix codes, then $\mathsf{imC}(\varphi_2 \circ \varphi_1(.)) = \mathsf{prefC}(\varphi_2(Q_1A^*))$.

Lemma 4.10. *The following input-output problem is in* P.

- Input*: A finite prefix code $S_0 \subset A^*$ (given explicitly by a list of words), and n right-ideal homomorphisms $\varphi_i : P_iA^* \to Q_iA^*$ for $i = 1, \ldots, n$ (given explicitly by finite tables); $P_i, Q_i \subset A^*$ are finite prefix codes.*
- Output*: The finite prefix code* $\mathsf{prefC}(\varphi_n \circ \cdots \circ \varphi_1(S_0A^*))$*, given explicitly by a list of words.*

Proof. We first prove the Lemma in case $n = 1$. We note that $\mathsf{prefC}(\varphi_1(S_0A^*)) = \varphi_1(\Pi)$, where Π is the prefix code that generates the right ideal $S_0A^* \cap PA^*$. By Lemma 4.9, Π is finite and can be explicitly found in deterministic polynomial time. From Π, an explicit list for $\varphi_1(\Pi)$ can be obtained in time polynomial in $|\Pi|$, $\ell(\Pi)$, $\|\varphi_1\|$ and $\ell(\varphi_1)$. By Theorem 4.5 applied to φ_1 and id_{S_0} we have: $|\mathsf{prefC}(\varphi_1(S_0A^*))| = |\mathsf{imC}(\varphi_1 \circ \mathsf{id}_{S_0})| \le |\mathsf{imC}(\varphi_1)| + |S_0|$, and $\ell(\varphi_1(S_0A^*)) = \ell(\varphi_1 \circ \mathsf{id}_{S_0}) \le \ell(\varphi_1) + \ell(S_0)$. So the total time for computing $\varphi_1(\Pi)$ is polynomial in terms of the size of the input. Let $p_1(\|\varphi_1\|, \ell(\varphi_1), |S_0|, \ell(S_0))$ be a polynomial upper bound for finding $\varphi_1(\Pi)$ from the input.

To prove the lemma in general, we compute the sequence of finite prefix codes $S_1, S_2, \ldots, S_n$, where $S_i = \mathsf{prefC}(\varphi_i(S_{i-1}A^*))$ for $i = 1, \ldots, n$. For this we repeatedly use the case $n = 1$ above, thus computing S_i from S_{i+1} and φ_i in deterministic time $\le p_i'(\|\varphi_i\|, \ell(\varphi_i), |S_{i-1}|, \ell(S_{i-1}))$, where p_i' is a polynomial. By Theorem 4.5 applied to $\varphi_{i-1}, \ldots, \varphi_1, \mathsf{id}_{S_0}$ we have:

$$|S_{i-1}| = \left|\mathsf{imC}(\varphi_{i-1} \circ \cdots \circ \varphi_1(S_0A^*))\right| \le |S_0| + \sum_{r=1}^{i-1} |\mathsf{imC}(\varphi_r)|,$$

$$\ell(S_{i-1}) = \ell(\mathsf{prefC}(\varphi_{i-1} \circ \cdots \circ \varphi_1(S_0A^*))) \le \ell(S_0) + \sum_{r=1}^{i-1} \ell(\varphi_i).$$

So, $|S_{i-1}|$ and $\ell(S_{i-1})$ are linearly bounded in terms of the input size, hence

$$p_i'(\|\varphi_i\|, \ell(\varphi_i), |S_{i-1}|, \ell(S_{i-1})) \le p_i(\|\varphi_i\|, \ell(\varphi_i), |S_0|, \ell(S_0)),$$

where p_i is a polynomial. Finally, the total time is at most $\sum_{i-1}^{n} p_i(\|\varphi_i\|, \ell(\varphi_i), |S_0|, \ell(S_0))$, which is a polynomial in terms of the input size. $\square$

Corollary 4.11. *The following input-output problem has a deterministic polynomial-time algorithm.*

- Input*: Right-ideal homomorphisms $\varphi_j : P_jA^* \to Q_jA^*$ (for $j = 1, \ldots, n$), where $P_j, Q_j \subset A^*$ are finite prefix codes; each φ_j is explicitly given by its table.*
- Output*: The set* $\mathsf{imC}(\varphi_n \circ \cdots \circ \varphi_1)$*, given explicitly by a list of words.*

Proof. This is a special case of Lemma 4.10 with $S_0 = \{\varepsilon\}$. $\square$

When we consider the word problem of $M_{k,1}$ over a finite generating set, we measure the input size by the length of input word (with each generator having length 1). But for the word problem of $M_{k,1}$ over the infinite generating set $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$ we count the length of the position transpositions $\tau_{i-1,i}$ as i. Indeed, at least $\log_2 i$ bits are needed to describe the subscript i of $\tau_{i-1,i}$. Moreover, in the connection between $M_{k,1}$ (over $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$) and circuits, $\tau_{i-1,i}$ is interpreted as the wire-crossing operation of wire number i and wire number $i - 1$; this suggests that viewing the size of $\tau_{i-1,i}$ as i is more natural. In any case, we will see next that the word problem of $M_{k,1}$ over $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$ is coNP-complete, even if the size of $\tau_{i-1,i}$ is more generously measured as i; this is a stronger result than if $\log_2 i$ were used.

Theorem 4.12 (*coNP-complete Word Problem*). *The word problem of $M_{k,1}$ over the infinite generating set $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$ is coNP-complete, where $\Gamma_{k,1}$ is any finite generating set of $M_{k,1}$.*

Proof. In [5] (see also [3]) it was shown that the word problem of the Thompson–Higman group $G_{k,1}$ over $\Gamma_{G_{k,1}} \cup \{\tau_{i-1,i} : i > 1\}$ is coNP-complete, where $\Gamma_{G_{k,1}}$ is any finite generating set of $G_{k,1}$. Hence, since the elements of the finite set $\Gamma_{G_{k,1}}$ can be expressed by a finite set of words over $\Gamma_{k,1}$, it follows that the word problem of $M_{k,1}$ over $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$ is also coNP-hard.

We will prove now that this word problem also belongs to coNP. The *input* consists of two words $(\rho_m, \ldots, \rho_1)$ and $(\sigma_n, \ldots, \sigma_1)$ over $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$. The *input size* is $\sum_{h=1}^{m} \ell(\rho_h) + \sum_{j=1}^{n} \ell(\sigma_j)$, where each generator in $\Gamma_{k,1}$ has length 1, and each generator of the form $\tau_{i-1,i}$ has length i. Since $\Gamma_{k,1}$ is finite there is a constant $c > 0$ such that $c \geq \ell(\gamma)$ for all $\gamma \in \Gamma_{k,1}$. By Theorem 4.5(2), the table of $\sigma_n \circ \cdots \circ \sigma_1$ (and more generally, the table of $\sigma_j \circ \cdots \circ \sigma_1$ for any j with $n \geq j \geq 1$) only contains words of length $\leq \sum_{j=1}^{n} \ell(\sigma_j)$, and similarly for $\rho_m \circ \cdots \circ \rho_1$ (and for $\rho_i \circ \cdots \circ \rho_1$, $m \geq i \geq 1$). So all the words in the tables for any $\sigma_j \circ \cdots \circ \sigma_1$ and any $\rho_i \circ \cdots \circ \rho_1$ have lengths that are linearly bounded by the size of the input $((\rho_m, \ldots, \rho_1), (\sigma_n, \ldots, \sigma_1))$.

Claim. Let $N = \max\{\sum_{i=1}^{m} \ell(\rho_i),\ \sum_{j=1}^{n} \ell(\sigma_j)\}$. Then $\rho_m \cdot \cdots \cdot \rho_1 \neq \sigma_n \cdot \cdots \cdot \sigma_1$ as elements of $M_{k,1}$ iff there exists $x \in A^N$ such that $\rho_m \circ \cdots \circ \rho_1(x) \neq \sigma_n \circ \cdots \circ \sigma_1(x)$.

Proof of the Claim: As we saw above, the tables of $\rho_m \circ \cdots \circ \rho_1$ and $\sigma_n \circ \cdots \circ \sigma_1$ only contain words of length $\leq N$. Thus, restricting $\rho_m \circ \cdots \circ \rho_1$ and $\sigma_n \circ \cdots \circ \sigma_1$ to $A^N A^*$ is an essential restriction, and the resulting tables have domain codes in A^N. Therefore, $\rho_m \cdot \cdots \cdot \rho_1 = \sigma_n \cdot \cdots \cdot \sigma_1$ (as elements of $M_{k,1}$) iff $\rho_m \circ \cdots \circ \rho_1$ and $\sigma_n \circ \cdots \circ \sigma_1$ are equal on A^N. □

The Claim yields a nondeterministic polynomial-time algorithm which decides whether there exists $x \in A^N$ such that $\rho_m \circ \cdots \circ \rho_1(x) \neq \sigma_n \circ \cdots \circ \sigma_1(x)$, as follows. The algorithm guesses $x \in A^N$, computes $\rho_m \circ \cdots \circ \rho_1(x)$ and $\sigma_n \circ \cdots \circ \sigma_1(x)$, and checks that they are different words ($\in A^*$) or that one is undefined and the other is a word. Applying Theorem 4.5(2) to $\rho_m \circ \cdots \circ \rho_1 \circ \mathrm{id}_{A^N}$ and to $\sigma_n \circ \cdots \circ \sigma_1 \circ \mathrm{id}_{A^N}$ shows that $|\rho_m \circ \cdots \circ \rho_1(x)| \leq 2N$ and $|\sigma_n \circ \cdots \circ \sigma_1(x)| \leq 2N$. Also by Theorem 4.5(2), all intermediate results (as we successively apply ρ_i for $i = 1, \ldots, m$, or σ_j for $j = 1, \ldots, n$) are words of length $\leq 2N$. These successive words are computed by applying the table of ρ_i or σ_j (when ρ_i or σ_j belong to $\Gamma_{k,1}$), or by directly applying the position permutation $\tau_{h,h-1}$ (if ρ_i or σ_j is $\tau_{h,h-1}$). Thus, the outputs $\rho_m \circ \cdots \circ \rho_1(x)$ and $\sigma_n \circ \cdots \circ \sigma_1(x)$ can be computed in polynomial time. The above is a nondeterministic polynomial-time algorithm for the negated word problem. Hence the word problem of $M_{k,1}$ over $\Gamma_{k,1} \cup \{\tau_{i-1,i} : i > 1\}$ is in coNP. □

4.3. The word problem of $M_{k,1}$ is in P

We now prove our main result.

Theorem 4.13 (*Word Problem in P*). *The word problem of the Thompson–Higman monoids $M_{k,1}$, over any finite generating set, can be decided in deterministic polynomial time.*

We assume that a fixed finite generating set $\Gamma_{k,1}$ of $M_{k,1}$ has been chosen. The input consists of two sequences $(\rho_m, \ldots, \rho_1)$ and $(\sigma_n, \ldots, \sigma_1)$ over $\Gamma_{k,1}$, and the input size is $m + n$. We want to decide whether, as elements of $M_{k,1}$, the products $\rho_m \cdot \cdots \cdot \rho_1$ and $\sigma_n \cdot \cdots \cdot \sigma_1$ are equal.

Overview of the proof:

- We compute the finite sets $\mathrm{imC}(\rho_m \circ \cdots \circ \rho_1)$, $\mathrm{imC}(\sigma_n \circ \cdots \circ \sigma_1) \subset A^*$ explicitly as lists of words. By Corollary 4.11 we can do this in polynomial time, and these sets have polynomial size. (However, by Proposition 4.1, the table sizes of $\rho_m \circ \cdots \circ \rho_1$ or $\sigma_n \circ \cdots \circ \sigma_1$ could be exponential in m or n.)
- We check whether $\mathrm{Im}(\rho_m \circ \cdots \circ \rho_1) \cap \mathrm{Im}(\sigma_n \circ \cdots \circ \sigma_1)$ is essential in both $\mathrm{Im}(\rho_m \circ \cdots \circ \rho_1)$ and $\mathrm{Im}(\sigma_n \circ \cdots \circ \sigma_1)$. By Lemma 4.9(Question 2) this can be done in polynomial time. If the answer is "no" then $\rho_m \cdot \cdots \cdot \rho_1 \neq \sigma_n \cdot \cdots \cdot \sigma_1$, since they don't have a common maximum essential extension. If "yes", we continue.
- We compute the finite prefix code $\Pi \subset A^*$ such that $\Pi A^* = \mathrm{Im}(\rho_m \circ \cdots \circ \rho_1) \cap \mathrm{Im}(\sigma_n \circ \cdots \circ \sigma_1)$. By Lemma 4.9 (Output 1) this can be done in polynomial time, and Π has polynomial size.
- For every $r \in \Pi$ we compute a deterministic finite automaton (DFA) accepting the finite set $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r) \subset A^*$, and a DFA accepting the finite set $(\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r) \subset A^*$. By Corollary 4.15 this can be done in polynomial time, and the DFAs have polynomial size. (Note that the finite sets themselves could have exponential size in m or n.)
- For every $r \in \Pi$ we check whether the DFA for $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r)$ and the DFA for $(\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r)$ are equivalent. By classical automata theory, this can be done in polynomial time.

These DFAs are equivalent for all $r \in \Pi$ iff $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r) = (\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r)$ for all $r \in \Pi$. Since ΠA^* is essential in $\mathrm{Im}(\rho_m \circ \cdots \circ \rho_1)$ and in $\mathrm{Im}(\sigma_n \circ \cdots \circ \sigma_1)$, this holds iff $\rho_m \cdot \cdots \cdot \rho_1 = \sigma_n \cdot \cdots \cdot \sigma_1$ (in $M_{k,1}$). □

Automata — notation and facts: "DFA" stands for *deterministic finite automaton*. The language accepted by a DFA $\mathcal{A}$ is denoted by $\mathcal{L}(\mathcal{A})$. A DFA is a structure (S, A, δ, s_0, F) where S is the set of states, A is the input alphabet, $s_0 \in S$ is the start state, $F \subseteq S$ is the set of accept states, and $\delta : S \times A \to S$ is the next-state function; in general, δ is a partial function. We extend the definition of δ to a function $S \times A^* \to S$ by defining $\delta(s, w)$ to be the state that the DFA reaches from s after reading w (for any $w \in A^*$ and $s \in S$). See [19,23] for background on finite automata. A DFA is called *acyclic* iff its underlying directed graph has no directed cycle. It is easy to prove that a language $L \subseteq A^*$ is finite iff L is accepted by an acyclic DFA. Moreover, L is a finite prefix code iff L is accepted by an acyclic DFA that has a single accept state. By the *size* of a DFA $\mathcal{A}$ we mean the number of states, $|S|$, of the DFA; we denote this by $\mathsf{size}(\mathcal{A})$. By the *min-depth* of a DFA $\mathcal{A}$ with single accept state we mean the length of the *shortest* path from the start state to the accept state; we denote this by $\mathsf{mindepth}(\mathcal{A})$. For a finite prefix code $P \subseteq A^*$ we denote the length of the longest word in P by $\ell(P)$, and we define the *total length* of P by $\|P\| = \sum_{x \in P} |x|$. For a language $L \subseteq A^*$ and a partial function $\Phi : A^* \to A^*$, we define the inverse image of L under Φ by $\Phi^{-1}(L) = \{x \in A^* : \Phi(x) \in L\}$. For $L \subseteq A^*$ we denote the set of all *strict* prefixes of the words in L by $\mathsf{spref}\ (L)$.

The reason why we use acyclic DFAs to describe finite sets is that a finite set can be exponentially larger than the number of states of a DFA that accepts it; e.g., A^n is accepted by an acyclic DFA with $n + 1$ states. This conciseness plays a crucial role in our polynomial-time algorithm for the word problem of $M_{k,1}$.

Lemma 4.14. *Let $\mathcal{A}$ be an acyclic DFA with a single accept state. Let $\varphi : PA^* \to QA^*$ be a right-ideal homomorphism, where $P, Q \subset A^*$ are finite prefix codes. We assume that $\ell(Q) \leq \mathsf{mindepth}(\mathcal{A})$, and that $\varphi^{-1}(\mathcal{L}(\mathcal{A})) \neq \varnothing$.*

Then $\varphi^{-1}(\mathcal{L}(\mathcal{A}))$ is accepted by a one-accept-state acyclic DFA whose size is $< \mathsf{size}(\mathcal{A}) + \|P\|$, and whose min-depth is $\geq \mathsf{mindepth}(\mathcal{A}) - \ell(Q)$. Moreover, the transition table of this DFA can be constructed deterministically in polynomial time, based on the transition table of $\mathcal{A}$ and the table of φ.

Proof. Let $\mathcal{A} = (S, A, \delta, s_0, \{s_A\})$ where s_A is the single accept state; s_A has no out-going edges (they would be useless). For any set $X \subseteq A^*$ and any $s \in S$ we denote $\{\delta(s, x) : x \in X\}$ by $\delta(s, X)$. Since $\mathcal{A}$ is acyclic, its state set S can be partitioned into the following two sets: $\delta(s_0, \mathsf{spref}(Q))$, and $\delta(s_0, QA^*)$. The block $\delta(s_0, \mathsf{spref}(Q))$ is non-empty since it contains s_0; the block $\delta(s_0, QA^*)$ is non-empty because of the assumption $\varphi^{-1}(\mathcal{L}(\mathcal{A})) \neq \varnothing$.

Since $\mathcal{L}(\mathcal{A})$ is a prefix code and φ is a right-ideal homomorphism, $\varphi^{-1}(\mathcal{L}(\mathcal{A}))$ is a prefix code. To accept $\varphi^{-1}(\mathcal{L}(\mathcal{A}))$ we introduce an acyclic DFA with single accept state, called $\varphi^{-1}(\mathcal{A})$, constructed as follows:

- State set of $\varphi^{-1}(\mathcal{A})$: $\mathsf{spref}(P) \cup \delta(s_0, QA^*)$.
  The start state is ε, i.e. the root of the prefix tree of P.
  The accept state is the accept state s_A of $\mathcal{A}$.
- State-transition function δ_1 of $\varphi^{-1}(\mathcal{A})$:
  For every $r \in \mathsf{spref}(P)$ and $a \in A$ such that $ra \in \mathsf{spref}(P)$: $\delta_1(r, a) = ra$.
  For every $r \in \mathsf{spref}(P)$ and $a \in A$ such that $ra \in P$: $\delta_1(r, a) = \delta(s_0, \varphi(ra))$.
  For every $s \in \delta(s_0, QA^*)$ and $a \in A$: $\delta_1(s, a) = \delta(s, a)$.

It follows immediately from this definition that for all $p \in P : \delta_1(\varepsilon, p) = \delta(s_0, \varphi(p))$.

The DFA $\varphi^{-1}(\mathcal{A})$ can be pictured as being constructed as follows: The DFA has two parts. The first part is the prefix tree of P, but with the leaves left out (and with the leaf edges dangling). The second part is the DFA $\mathcal{A}$ restricted to the state subset $\delta(s_0, QA^*)$. The two parts are connected together by gluing each (hypothetical) leaf $p \in P$ to the state $\delta(s_0, \varphi(p)) \in \delta(s_0, QA^*)$.

The description of $\varphi^{-1}(\mathcal{A})$ constitutes a deterministic polynomial time algorithm for constructing the transition table of $\varphi^{-1}(\mathcal{A})$, based on the transition table of $\mathcal{A}$ and on the table of φ.

We will prove now that the DFA $\varphi^{-1}(\mathcal{A})$ accepts exactly $\varphi^{-1}(\mathcal{L}(\mathcal{A}))$; i.e., $\varphi^{-1}(\mathcal{L}(\mathcal{A})) = \mathcal{L}(\varphi^{-1}(\mathcal{A}))$.

[$\subseteq$] Consider any $y \in \mathcal{L}(\mathcal{A})$ such that $\varphi^{-1}(y) \neq \varnothing$. We want to show that $\varphi^{-1}(\mathcal{A})$ accepts all the words in $\varphi^{-1}(y)$. Since $\varphi^{-1}(y) \neq \varnothing$ we have $y = qw$ for some strings $q \in Q = \mathsf{imC}(\varphi)$ and $w \in A^*$. Since Q is a prefix code, q and w are uniquely determined by y. Moreover, since $y \in \mathcal{L}(\mathcal{A})$ it follows that $y = qw$ has an accepting path in $\mathcal{A}$ of the form

$$s_0 \xrightarrow{q} \delta(s_0, q) \xrightarrow{w} s_A.$$

Then for every $x \in \varphi^{-1}(y)$ we have $x = pv$ for some strings $p \in P$ and $v \in A^*$, so $\varphi(x) = \varphi(p)\, v$; we also have $\varphi(x) = qw$, hence $\varphi(p)$ and q are prefix-comparable. Therefore, $\varphi(p) = q$, since Q is a prefix code, and hence $v = w$. Thus every $x \in \varphi^{-1}(qw)$ has the form pw for some string $p \in \varphi^{-1}(q)$. Now in $\varphi^{-1}(\mathcal{A})$ there is the following accepting path on input $x = pw \in \varphi^{-1}(qw) = \varphi^{-1}(q)\, w$:

$$\varepsilon \xrightarrow{p} \delta_1(\varepsilon, p) = \delta(s_0, \varphi(p)) \xrightarrow{w} s_A.$$

Thus $\varphi^{-1}(\mathcal{A})$ accepts $x = pw$.

[$\supseteq$] Suppose $\varphi^{-1}(\mathcal{A})$ accepts x. Then, because the prefix tree of P forms the beginning of $\varphi^{-1}(\mathcal{A})$, x has the form $x = pw$ for some $p \in P$ and $w \in A^*$. The accepting path in $\varphi^{-1}(\mathcal{A})$ on input pw has the form

$$s_0 \xrightarrow{p} \delta_1(\varepsilon, p) = \delta(s_0, \varphi(p)) \xrightarrow{w} s_A.$$

Also, $\varphi(x) = qw$ where $q = \varphi(p) \in Q$. Hence $\mathcal{A}$ has the following path on input qw:

$$s_0 \xrightarrow{q} \delta(s_0, q) = \delta(s_0, \varphi(p)) \xrightarrow{w} s_A.$$

So, $qw \in \mathcal{L}(\mathcal{A})$. Hence, $x \in \varphi^{-1}(qw) \subseteq \varphi^{-1}(\mathcal{L}(\mathcal{A}))$. Thus $\mathcal{L}(\varphi^{-1}(\mathcal{A})) \subseteq \varphi^{-1}(\mathcal{L}(\mathcal{A}))$. $\square$

Corollary 4.15. *Let $\mathcal{A}$ be an acyclic DFA with a single accept state. For $i = 1, \ldots, n$, let $P_i, Q_i \subset A^*$ be finite prefix codes, and let $\varphi_i : P_iA^* \to Q_iA^*$ be right-ideal homomorphisms. We assume that $\sum_{i=1}^{n} \ell(Q_i) \le \mathsf{mindepth}(\mathcal{A})$, and that $(\varphi_n \circ \cdots \circ \varphi_1)^{-1}(\mathcal{L}(\mathcal{A})) \neq \varnothing$,*

Then $(\varphi_n \circ \cdots \circ \varphi_1)^{-1}(\mathcal{L}(\mathcal{A}))$ is accepted by a one-accept-state acyclic DFA whose size is $< \mathsf{size}(\mathcal{A}) + \sum_{i=1}^{n} \|P_i\|$, and whose min-depth is $\ge \mathsf{mindepth}(\mathcal{A}) - \sum_{i=1}^{n} \ell(Q_i)$.

Moreover, the transition table of this DFA can be constructed deterministically in polynomial time, based on the transition table of $\mathcal{A}$ and the tables of φ_i $(i = 1, \ldots, n)$.

Proof. This follows from Lemma 4.14 by induction on n. See [1] for details. □

Proof of Theorem 4.13. Let $(\rho_m, \ldots, \rho_1)$ and $(\sigma_n, \ldots, \sigma_1)$ be two sequences of generators from $\Gamma_{k,1}$. We want to decide in deterministic polynomial time whether the products $\rho_m \cdot \cdots \cdot \rho_1$ and $\sigma_n \cdot \cdots \cdot \sigma_1$ are the same (as elements of $M_{k,1}$).

First, by Corollary 4.11, we can compute the sets $\mathsf{imC}(\rho_m \circ \cdots \circ \rho_1)$ and $\mathsf{imC}(\sigma_n \circ \cdots \circ \sigma_1)$ explicitly as lists of words, in polynomial time. By Lemma 4.9 we can check in polynomial time whether the right ideal $\mathsf{Im}(\rho_m \circ \cdots \circ \rho_1) \cap \mathsf{Im}(\sigma_n \circ \cdots \circ \sigma_1)$ is essential in $\mathsf{Im}(\rho_m \circ \cdots \circ \rho_1)$ and in $\mathsf{Im}(\sigma_n \circ \cdots \circ \sigma_1)$. If not, we immediately conclude that $\rho_m \cdot \cdots \cdot \rho_1 \neq \sigma_n \cdot \cdots \cdot \sigma_1$. Otherwise, Lemma 4.9 lets us compute a generating set Π for the right ideal $\mathsf{Im}(\rho_m \circ \cdots \circ \rho_1) \cap \mathsf{Im}(\sigma_n \circ \cdots \circ \sigma_1)$, in deterministic polynomial time; this generating set Π will be a finite prefix code, given explicitly by a list of words. By Corollary 4.7 and because $\Pi \subseteq \mathsf{imC}(\rho_m \circ \cdots \circ \rho_1) \cup \mathsf{imC}(\sigma_n \circ \cdots \circ \sigma_1)$, Π has linearly bounded cardinality and the length of the longest words in Π is linearly bounded.

To find out whether $\rho_m \cdot \cdots \cdot \rho_1 = \sigma_n \cdot \cdots \cdot \sigma_1$, it is sufficient to check whether $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r) = (\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r)$ for every $r \in \Pi$, since ΠA^* is essential in both $\mathsf{Im}(\rho_m \circ \cdots \circ \rho_1)$ and $\mathsf{Im}(\sigma_n \circ \cdots \circ \sigma_1)$. Let $\lambda = \max\{\sum_{i=1}^{m} \ell(\mathsf{imC}(\rho_i)), \sum_{j=1}^{n} \ell(\mathsf{imC}(\sigma_j))\}$. For every $r \in \Pi$ we have $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r) \neq \varnothing$ and $(\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r) \neq \varnothing$, because $\Pi \subset \mathsf{Im}(\rho_m \circ \cdots \circ \rho_1) \cap \mathsf{Im}(\sigma_n \circ \cdots \circ \sigma_1)$.

If $|r| \ge \lambda$ then Corollary 4.15 implies that $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r)$ is accepted by an acyclic one-accept-state DFA $\mathcal{A}_\rho$, constructed deterministically in polynomial time; similarly, we construct an acyclic one-accept-state DFA $\mathcal{A}_\sigma$ which accepts $(\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r)$.

If $|r| < \lambda$, we replace r by the set $rA^{\lambda-|r|}$. It is easy to see that $rA^{\lambda-|r|}$ is accepted by an acyclic single-accept-state DFA with $\lambda + 1$ states. By Corollary 4.15, $(\rho_m \circ \cdots \circ \rho_1)^{-1}(rA^{\lambda-|r|})$ is accepted by an acyclic one-accept-state DFA $\mathcal{A}_\rho$, constructed deterministically in polynomial time. Similarly, we construct an acyclic one-accept-state DFA $\mathcal{A}_\sigma$ which accepts $(\sigma_n \circ \cdots \circ \sigma_1)^{-1}(rA^{\lambda-|r|})$.

Obviously, $(\rho_m \circ \cdots \circ \rho_1)^{-1}(rA^{\lambda-|r|}) = (\sigma_n \circ \cdots \circ \sigma_1)^{-1}(rA^{\lambda-|r|})$ (or, in case $|r| \ge \lambda$, $(\rho_m \circ \cdots \circ \rho_1)^{-1}(r) = (\sigma_n \circ \cdots \circ \sigma_1)^{-1}(r)$) if and only if $\mathcal{A}_\rho$ and $\mathcal{A}_\sigma$ accept the same language, i.e. they are equivalent DFAs. It is well known (see e.g., [19], or [23] pp. 103–104) that the equivalence problem for DFAs that are given explicitly by transition tables, is decidable deterministically in polynomial time. This proves Theorem 4.13. □

Acknowledgements

I would like to thank John Meakin for many discussions over the years concerning the Thompson groups and generalizations to inverse monoids. The author was supported by NSF grant CCR-0310793. The first version of this paper appeared in http://arxiv.org/abs/0704.0189 (2 April 2007).

References

[1] J.C. Birget, Monoid generalizations of the Richard Thompson groups, Mathematics ArXiv: math.GR/0704.0189 v.1 (2 April 2007), v.2 (28 April 2008) (Preprint of the present paper, with more detailed proofs).
[2] J.C. Birget, One-way permutations, computational asymmetry and distortion, J. Algebra (in press) Mathematics ArXiv: http://arxiv.org/abs/0704.1569 (12 April 2007).
[3] J.C. Birget, Factorizations of the Thompson–Higman groups, and circuit complexity, Internat. J. Algebra Comput. 18.2 (2008) 285–320. Preprint: Mathematics ArXiv: math.GR/0607349, July 2006.
[4] J.C. Birget, The groups of Richard Thompson and complexity, Internat. J. Algebra Comput. 14 (5,6) (2004) 569–626. Preprint: Mathematics ArXiv: math.GR/0204292, April 2002.
[5] J.C. Birget, Circuits, coNP-completeness, and the groups of Richard Thompson, Internat. J. Algebra Comput. 16 (1) (2006) 35–90. Preprint: Mathematics ArXiv: http://arXiv.org/abs/math.GR/0310335, October 2003.
[6] M. Brin, C. Squier, Groups of piecewise linear homeomorphisms of the real line, Invent. Math. 79 (1985) 485–498.
[7] M. Brin, The Chameleon groups of Richard J. Thompson: Automorphisms and dynamics, Publ. Math. de l'IHES 84 (1997) 5–33.
[8] K. Brown, R. Geoghegan, An infinite-dimensional torsion-free FP_∞ group, Invent. Math. 77 (1984) 367–381.
[9] J. Burillo, S. Cleary, M. Stein, J. Taback, Combinatorial and metric properties of Thompson's group T, Trans. AMS (in press). Mathematics ArXiv: math.GR/0503670.
[10] J.W. Cannon, W.J. Floyd, W.R. Parry, Introductory notes on Richard Thompson's groups, L'Enseignement Math. 42 (1996) 215–256.
[11] A.H. Clifford, G.B. Preston, The Algebraic Theory of Semigroups, Vol. 1, in: Mathematical Survey, No 7 (I), American Mathematical Society, Providence, 1961.
[12] J. Cuntz, Simple C^*-algebras, Commun. Math. Phys. 57 (1977) 173–185.
[13] P. Dehornoy, Geometric presentations for Thompson's groups, J. Pure Appl. Algebra 203 (2005) 1–44.
[14] J. Dixmier, Traces sur les C^*-algèbres II, Bull. Sci. Math. 88 (1964) 39–57.
[15] E. Ghys, V. Sergiescu, Sur un groupe remarquable de difféomorphismes du cercle, Comment. Math. Helv. 62 (2) (1987) 185–239.
[16] V. Guba, M.V. Sapir, Diagram groups, Mem. Amer. Math. Soc. 130 (620) (1997) viii+117 pages.
[17] P.A. Grillet, Semigroups, An Introduction to the Structure Theory, Marcel Dekker, New York, 1995.

[18] G. Higman, Finitely presented infinite simple groups, in: Notes on Pure Mathematics, vol. 8, The Australian National University, Canberra, 1974.
[19] J. Hopcroft, J. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 1979.
[20] B. Hughes, Trees, ultrametrics, and noncommutative geometry (revised June 2007), 60 pages; PDF http://www.math.vanderbilt.edu/~hughescb.
[21] M.V. Lawson, Orthogonal completions of the polycyclic monoids, Comm. Algebra 35 (2007) 1651–1660.
[22] J. Lehnert, P. Schweitzer, The co-word problem for the Higman–Thompson group is context-free, Bull. London Math. Soc. 39 (2007) 235–241.
[23] H. Lewis, Ch. Papadimitriou, Elements of the Theory of Computation, 2nd ed., Prentice Hall, 1998.
[24] R. McKenzie, R.J. Thompson, An elementary construction of unsolvable word problems in group theory, in: W.W. Boone, F.B. Cannonito, R.C. Lyndon (Eds.), Word Problems, North-Holland, 1973, pp. 457–478.
[25] V.V. Nekrashevych, Cuntz–Pimsner algebras of group actions, J. Operator Theory 52 (2) (2004) 223–249.
[26] Elizabeth A. Scott, A construction which can be used to produce finitely presented infinite simple groups, J. Algebra 90 (1984) 294–322.
[27] Richard J. Thompson, Manuscript (1960s).
[28] Richard J. Thompson, Embeddings into finitely generated simple groups which preserve the word problem, in: S. Adian, W. Boone, G. Higman (Eds.), Word Problems II, North-Holland, 1980, pp. 401–441.