Noname manuscript No.
(will be inserted by the editor)

Independent Finite Automata on Cayley Graphs

Ville Salo · Ilkka Törmä

Received: date / Accepted: date

Abstract In the setting of symbolic dynamics on discrete finitely generated infinite groups, we define a model of finite automata with multiple independent heads that walk on Cayley graphs, called group-walking automata, and use it to define subshifts. We characterize the torsion groups (also known as periodic groups) as those on which the group-walking automata are strictly weaker than Turing machines, and those on which the head hierarchy is infinite.

Mathematics Subject Classification (2000) 68Q45 · 37B50 · 20E99

Keywords group-walking automaton · torsion group · periodic group · multi-headed automaton · subshift

1 Introduction

One of the central objects in symbolic dynamics is the dynamical system S^G (where G is a discrete group and S a finite alphabet), called the *full shift*, where G acts by translations [4]. In particular, one studies its subsystems, usually called *subshifts*, and classes of such subsystems. Some of the important classes studied are the SFTs (subshifts defined by a finite set of forbidden patterns), sofic shifts

Research supported by FONDECYT grant 3150552 (first author)

Ville Salo
Center of Mathematical Modeling
University of Chile, Santiago, Chile

Ilkka Törmä
Department of Computer Science
Boston University
Boston, Massachusetts, USA

Ville Salo · Ilkka Törmä
Department of Mathematics and Statistics
University of Turku, Turku, Finland
E-mail: vosalo@utu.fi, iatorm@utu.fi

(the factors of SFTs) and the effective, or Π_1^0 subshifts (defined by a recursively enumerable set of forbidden patterns). SFTs and sofic shifts are natural objects to study on all groups, and a robust notion of effectiveness of subshifts on arbitrary groups, which we call *intrinsically* Π_1^0, is given in [2].

In this paper we define some new families of subshifts on an arbitrary (discrete finitely generated infinite) group G. Namely, we discuss the class of subshifts defined by certain multi-headed automata that walk on the Cayley graph of the group G. Our model is inspired by various flavors of two-dimensional finite automata, and is also strongly related to the *pebble automata* used in [3,5]. We believe that in the context of symbolic dynamics on groups, multi-headed finite automata provide an interesting notion to study that combines the geometric and computational properties of the underlying group. We have previously studied the case $G = \mathbb{Z}^d$ in [11], the main result being that three-headed finite-state automata define the same subshifts as general Turing machines.[1] It turns out that up to notational complications and a few simple tricks, the same result can be shown on all groups containing a copy of $\mathbb{Z}$. We show this in Theorem 1 and Theorem 2: on any group containing an isomorphic copy of $\mathbb{Z}$ as a subgroup, all Π_1^0 subshifts can be recognized by three-headed automata, and four-headed automata recognize exactly the intrinsically Π_1^0 subshifts. Furthermore, we prove that five or more heads provide no additional power to the model, so the hierarchy of heads collapses to the fourth level. If the word problem of the group is also decidable, the classes of Π_1^0 and intrinsically Π_1^0 subshifts coincide, so the head hierarchy further collapses to the third level.

Most finitely generated groups of practical interest contain a copy of $\mathbb{Z}$. For example, in addition to infinite (finitely generated) abelian groups, this is true for free groups, Baumslag-Solitar groups, the Heisenberg group, the Thompson groups F, T and V, and the general linear groups $GL(n, \mathbb{Z})$. In fact, infinite finitely generated groups without a copy of $\mathbb{Z}$, known as torsion groups, are quite rare and hard to construct. Nevertheless, many examples exist in the literature. The question is particularly hard in the case that the torsion is bounded, that is, there exists $n \in \mathbb{N}$ such that every element of the group generates a subgroup of order at most n. See [1] for a discussion of groups with bounded torsion. In the case of unbounded torsion, there are examples that are relatively simple to define, and simple to prove torsion. We mention in particular [6,7].

Given that such groups exist, an obvious question is whether we can extend Theorem 1 and Theorem 2 to this case. It turns out that we cannot: in Theorem 3 we show that a subshift on a torsion group accepted by a multi-headed automaton 'cannot be too sparse', and as a further result we obtain Theorem 6, which characterizes the torsion groups as those on which multi-headed automata are strictly weaker than Turing machines. Furthermore, in Theorem 5 we show that the hierarchy of heads is infinite on every (infinite) torsion group. The proof of this result is by a reduction to one-dimensional finite automata and diagonalization.

Finally, we prove in Proposition 2 that conditioned on the existence of certain 'unpredictable' torsion groups, there exist non-torsion groups on which four-headed automata recognize a strictly larger set of configurations than three-headed ones. This (conditionally) shows that Theorem 2 is optimal in the number of heads.

[1] In the earlier article [5], essentially the same observation is made in a slightly different setting on the group $\mathbb{Z}^2$.

This article is an extended version of the conference paper [12], with the two last mentioned results being new to this version. The paper [12] is completely replaced by this one, but we note that this paper is not an extended version of [11], whose only intersection with the present paper is that Theorem 5 of [11] is essentially Theorem 1 of this paper.

2 Definitions and examples

2.1 Subshifts

In this section, we define some basic notions of of symbolic dynamics and computability. Some references on symbolic dynamics on general groups are [4, 2], and a standard reference on $\mathbb{Z}$ is [9].

Let G be a group with identity element $1_G \in G$. Our groups are always infinite (the finite case being trivial) and finitely generated (since the notions we consider are local). For convenience, if G is finitely generated, we fix a symmetric finite set $\mathcal{G}(G) \subset G$ of generators for it. The sets $\mathcal{G}(G)^*$ and $\mathcal{G}(G)^{\leq n}$ consist of all finite words over $\mathcal{G}(G)$ and those of length at most n, and for $v, w \in \mathcal{G}(G)^*$, we denote $v \sim w$ and $v \sim g$ if the words correspond to the same element $g \in G$. We denote by $B_G(n)$ the ball of radius n with respect to the fixed set of generators: $B_G(n) = \{g \in G \mid w \in \mathcal{G}(G)^{\leq n}, w \sim g\}$. The *(right) G-distance* $d_G(g, h)$ of two elements $g, h \in G$ is the smallest $n \geq 0$ for which $g \in h \cdot B_G(n)$. Equivalently, it is the length of the shortest word $w \in \mathcal{G}(G)^*$ such that $g \sim h \cdot w$. Equivalently, it is the smallest $n \in \mathbb{N}$ such that $h^{-1}g \in B_G(n)$.

A *torsion element* of a group G is an element $g \in G$ that satisfies $g^n = 1_G$ for some $n \geq 1$. If all elements of G are torsion, then G is a *torsion group*. To each torsion element $g \in G$ we associate its *order* $t_G(g) = \min\{n \geq 1 \mid g^n = 1_G\}$, and to each finitely generated torsion group we associate the *torsion function* $T_G : \mathbb{N} \to \mathbb{N}$, defined by $T_G(n) = \max\{t_G(g) \mid g \in B_G(n)\}$. A non-torsion group, conversely, is one containing an isomorphic copy of $\mathbb{Z}$.

Both *alphabet* and *state set* mean any finite set. The symbol S always means an alphabet, and the set S^G is the *full G-shift over S*. Its elements, usually denoted by x, y, z, are called *configurations*. We define both a left and a right action of G on S^G, called the *left and right shifts*. The left action is given by $(g \cdot x)_h = x_{g^{-1}h}$. The right action is given by $\sigma^R_g(x)_h = x_{hg}$.

We give S^G the product topology induced by the discrete topology on S, making it a compact metrizable space. It is easy to show that both actions are continuous in this topology. A *subshift* of S^G is a topologically closed subset of S^G closed under the left action of G. The *subshift generated by a configuration* $x \in S^G$ is the smallest subshift of S^G containing x. Equivalently, it is the topological closure of the orbit of x in the left action on S^G. For a set of configurations $A \subset S^G$, similarly the *subshift generated by* A is the smallest subshift containing A.

A *cellular automaton* on a subshift $X \subset S^G$ is a continuous map $f : X \to X$ that commutes with the left shifts in the sense that $g \cdot f(x) = f(g \cdot x)$ holds for all

$x \in S^G$ and $g \in G$. We denote by $\mathrm{Aut}(X)$ the group of bijective cellular automata on X under composition.[2]

A *pattern (on G)* is a function $P \in S^D$, where $D = D(P)$ is a finite subset of G, called the *domain* of P. Each pattern P defines a *cylinder set* $[P] = \{x \in S^G \mid x|_D = P\}$. The clopen (topologically closed and open) sets in S^G are precisely the finite unions of cylinders, and form a basis for the topology.

Subshifts can be characterized as sets $X \subset S^G$ for which there exists a set of *forbidden patterns* $\mathcal{F}$ such that

$$X = \{x \in S^G \mid \forall P \in \mathcal{F} : \forall g \in G : g \cdot x \notin [P]\}.$$

We also need to discuss patterns that appear in the subshift, so for $A \subset G$ let $\mathcal{L}_A(X) = \{P : A \to S \mid \exists x \in X : x|_A = P\}$. Each cellular automaton on X has a *radius* $r \in \mathbb{N}$ and a *local rule* $F : S^{B_G(r)} \to S$ such that $f(x)_g = F((g^{-1} \cdot x)|_{B_G(r)})$ holds for all $x \in X$ and $g \in G$.

The *Cayley graph* of a finitely generated group G with respect to a symmetric set of generators $\mathcal{G}(G)$ is the labeled graph (V, E, L) where the set of edge labels is $L = \mathcal{G}(G)$, the vertices are $V = G$ and $(g, h, \lambda) \in E$ (that is, there is an edge from g to h with label λ) if and only if $g \cdot \lambda = h$. The right distance of a group translates to the obvious metric in its Cayley graph, namely the length of the shortest connecting path. Note that a subshift is invariant under the left action, but the Cayley graph is drawn with respect to right actions. When the local rule of a cellular automaton looks for the neighbors of a cell g, it looks at the cells with index gh, where h comes from a finite set. This means that it is looking at the neighbors of g in the Cayley graph. We will not explicitly discuss Cayley graphs, but whenever discussing the movement of a finite-state automaton on a group, we have this graph in mind, giving meaning to intuitive statements like 'the head moves one step in direction h' (meaning that it moves from its original position g to the position gh).

Example 1 Let G be the free group generated by $h_1, h_2 \in G$, and $X \subset S^G$ the set

$$\{x \in S^G \mid \forall g \in G : \forall n \in \mathbb{Z} : x_{gh_1^n} = x_g\}.$$

We show that X is a subshift. It is clear that it is topologically closed. Let then $x \in X$ and $f \in G$. We need to show $f \cdot x \in X$. Given $g \in G$ and $n \in \mathbb{Z}$, we have

$$(f \cdot x)_{gh_1^n} = x_{f^{-1}gh_1^n} = x_{f^{-1}g} = (f \cdot x)_g$$

by the definition of the left action, and the fact $x \in X$. This is also clear from writing X in terms of the action:

$$X = \{x \in S^G \mid \forall g \in G : \forall n \in \mathbb{Z} : (g^{-1} \cdot x)_{h_1^n} = (g^{-1} \cdot x)_{1_G}\}$$

Definition 1 If $S \ni 0$ is a finite alphabet, then the *one-S subshift* on a group G is the subshift $X^1_{S,G} \subset S^G$ where a finite pattern $P \in S^D$ is forbidden if and only if there exist $d \neq e \in D$ with $P_e \neq 0$ and $P_d \neq 0$. If $0 \notin S$, we write $X^1_{S,G} = X^1_{S \cup \{0\},G}$.

[2] This is a group because by the compactness of X, the inverse of a CA is continuous, and the inverse of a bijection commuting with a group action is easily seen to commute with the group action.

More concretely, the one-S subshift contains those configurations of S^G where at most one coordinate contains a nonzero symbol. The group G is usually clear from context, and we write X^1_S for $X^1_{S,G}$.

Definition 2 Let $S \ni 0$ be a finite alphabet. A configuration $x \in S^G$ is *k-sparse* if it satisfies $|\{g \in G \mid x_g \neq 0\}| \leq k$. A subshift is *$k$-sparse* if each of its configurations is k-sparse, and *sparse* if it is k-sparse for some $k \in \mathbb{N}$.

For a configuration $x \in S^G$ and $a \in S$, define the *a-support* of x by $\mathrm{supp}_a(x) = \{g \in G \mid x_g \neq a\}$. The *support* of x is its 0-support.

The one-S subshift X^1_S is of course a 1-sparse subshift on any group, and all 1-sparse subshifts are of this form. Note that in a sparse subshift, there is a global bound on the number of nonzero symbols. The *sum* $x + y$ of sparse configurations $x, y \in S^G$ with disjoint supports (no $g \in G$ satisfies $x_g \neq 0$ and $y_g \neq 0$) is defined by

$$(x+y)_g = \begin{cases} x_g & \text{if } x_g \neq 0, \\ y_g & \text{otherwise.} \end{cases}$$

A finite pattern is represented computationally as a finite list of word-symbol pairs $(w, d) \in \mathcal{G}(G)^* \times S$. Such a list is *inconsistent* if it contains two pairs (v, d) and (w, e) with $v \sim w$ and $d \neq e$ (in this case, it does not actually encode a pattern), and otherwise *consistent*.

Definition 3 Let G be a finitely generated group. The *word problem of* G is the set $E = \{w \in \mathcal{G}(G)^* \mid w \sim 1_G\}$ of words that represent the identity element of G. Whether the word problem is decidable is independent of the chosen generator set. We say G is *recursively presented* if $G \cong \langle g_1, \ldots, g_k \mid w_1, w_2, \ldots \rangle$, where $(w_i)_{i \in \mathbb{N}}$ is a recursively enumerable sequence of words in $\mathcal{G}(G)^*$ that represent the identity element of G.[3] This is equivalent to the set E being recursively enumerable.

A subshift on G is Π^0_1 if there exists a Turing machine enumerating a list of (possibly inconsistent) forbidden patterns for it, with the interpretation that an inconsistent pattern defines an empty cylinder. A subshift X on G is *intrinsically* Π^0_1 if there exists an oracle Turing machine that, given an oracle for the word problem of G, enumerates a list of consistent forbidden patterns for X.

If the group G has a decidable word problem, then the classes of Π^0_1 and intrinsically Π^0_1 subshifts coincide; in general, the latter class contains the former.

In [2], what we call Π^0_1 is called *effectively closed*, and what we call intrinsically Π^0_1 is called *G-effective*. The latter notion was first defined and studied there. They also show that 'group-walking Turing machines' define precisely the intrinsically Π^0_1 subshifts on every group.[4]

2.2 Automata

We now define group-walking automata and the subshifts they recognize. Here and henceforth, by π_i we mean the projection to the ith coordinate of a finite Cartesian product.

[3] The term 'recursively presented' comes from the fact that one may assume $\{w_i \mid i \in \mathbb{N}\}$ to be a recursive set of words.

[4] Turing machines can do this on any group, as they can modify a configuration and create a tape for themselves. Our finite-state automata fail in this task on torsion grops, by Theorem 6.

Definition 4 A *k-headed group-walking automaton* on the full shift S^G is a tuple $A = (\prod_{i=1}^k Q_i, f, I, F, S)$, where $Q_1, Q_2, \ldots, Q_k$ are state sets not containing the symbol 0, I and F are finite clopen subsets of the product subshift $Y = \prod_{i=1}^k X^1_{Q_i}$, and $f : S^G \times Y \to S^G \times Y$ is a CA satisfying $\pi_1 \circ f = \pi_1$ and

$$\pi_i(\pi_2(f(x, y))) = 0^G \iff \pi_i(y) = 0^G$$

for all $x \in S^G$, $y \in Y$ and $i \in \{1, \ldots, k\}$.

For a k-headed automaton A as above, we denote by $\mathcal{S}(A) \subset S^G$ the subshift

$$\{x \in S^G \mid \forall g, h \in G, y \in I, n \in \mathbb{N} : h \cdot \pi_2(f^n(g \cdot x, y)) \notin F\}.$$

We denote by $\mathcal{S}(G, k)$ the class of all subshifts $\mathcal{S}(A)$ for k-headed automata A. We also write $\mathcal{S}(G) = \bigcup_{k \geq 1} \mathcal{S}(G, k)$.

The intuition for these definitions is the following. A configuration $y \in Y = \prod_{i=1}^k X^1_{Q_i}$ consists of k layers $\pi_i(y)$, each of which contains at most one nonzero symbol $q_i \in Q_i$, representing the i'th head of the automaton in state q_i. The cellular automaton f is the update function of the heads: since f has a finite radius, the heads can only move at a bounded speed, and interact over bounded distances. Also, the condition $\pi_1 \circ f = \pi_1$ ensures that the automaton cannot alter the configuration of S^G that it runs on; in other words, our model is "read-only". The second condition on f guarantees that the heads will not disappear under the dynamics f: $\pi_i(\pi_2(f(x, y))) = 0^G \iff \pi_i(y) = 0^G$ means that the i'th head is present in $\pi_2(f(x, y)) \in Y$ if and only if it is present in $y \in Y$, for any configuration $x \in S^G$. In one step, each head can move at most a distance of r on the Cayley graph, where $r \geq 0$ is the radius of the local rule of the cellular automaton f, and one can also define model-specific local rules for these automata that explicitly describe the movement of the heads, which is the approach taken in [11].

The clopen sets $I, F \subset Y$ are the *initial and final states* of the automaton. Each of them is a finite union of cylinder sets $[P]$, and since they are also finite as sets, each of the patterns P necessarily contains all k heads of the automaton. Thus, an initial or finite state specifies the position and internal state for each head, and we translate them by every element of G in the definition of $\mathcal{S}(G)$.

We make a brief note on our definitions. The reader may rightfully wonder why, for example, the heads always start together, and must gather together in the end to reject the configuration. We refer the interested reader to [11] for a discussion of these choices, as the models are essentially equivalent. The short justification is simply that we need to fix some model to prove results, and we feel that the characterizations of torsion groups we obtain are more interesting than the details of the model, since many of them could be changed without affecting the results. However, the fact that our heads cannot communicate over arbitrarily large distances turns out to be very important for proving that the head hierarchy is infinite. We discuss this in Section 8.

Example 2 Let G be again the free group generated by the two elements $h_1, h_2 \in G$, and let $S = \{0, 1\}$. We define a two-headed group-walking automaton $A = (Q_1 \times Q_2, f, I, F, S)$ on G as follows. The local state sets are $Q_1 = \{q(h_1), q(h_1^{-1})\}$ and $Q_2 = \{q(h_2), q(h_2^{-1})\}$, the set of initial states I contains only the cylinder set $\{x \in (Q_1 \times Q_2)^G \mid x_{1_G} = (q(h_1), q(h_2))\}$, and the set of final states F contains the cylinder $\{x \in (Q_1 \times Q_2)^G \mid x_{1_G} = (q(h_1^{-1}), q(h_2^{-1}))\}$. This means that the heads of the automaton are initialized at the same coordinate in states $q(h_1)$ and $q(h_2)$, and a configuration is rejected if they ever return to the same coordinate in states $q(h_1^{-1})$ and $q(h_2^{-1})$. The CA f moves each head by the step indicated in its state, and if a head encounters a symbol 1 in state $q(h_1)$ or $q(h_2)$, it assumes the respective inverse state $q(h_1^{-1})$ or $q(h_2^{-1})$.

In a run of the automaton, the heads start moving in the directions h_1 and h_2 until they encounter symbols 1, and then turn back. If both of them turn at the same time, they will meet again where they started, in the states $q(h_1^{-1})$ and $q(h_2^{-1})$, so the configuration is rejected. If not, the configuration is not rejected. Thus the automaton A defines the subshift $X \subset S^G$ with the forbidden patterns

$$\{1_G \mapsto 0, h_1 \mapsto 0, \ldots, h_1^{n-1} \mapsto 0, h_1^n \mapsto 1, h_2 \mapsto 0, \ldots, h_2^{n-1} \mapsto 0, h_2^n \mapsto 1\}$$

for all $n \geq 1$. It is not an SFT.

Naturally, Turing machines are stronger than multi-headed finite automata.

Lemma 1 *If G is a finitely generated group and $X \in \mathcal{S}(G)$, then X is intrinsically Π_1^0.*

Proof Let A be a group-walking automaton that defines X. We construct an oracle Turing machine T_A that outputs its forbidden patterns. The machine T_A enumerates all consistent patterns over G (using the oracle for the word problem of G), and simulates a run of the automaton A on each of them, from every initial state. If one of the heads exits the pattern during such a simulation, or every head enters an infinite loop, that simulation is simply discarded. If one of the runs enters a rejecting state on the pattern P before exiting it (from any initial configuration and initial position on the domain $D(P)$), the machine T_A outputs the pattern P. It is clear that T_A defines the same subshift as A. □

3 Non-torsion groups

On non-torsion groups, there are essentially no restrictions on the types of computation a multi-headed finite state automaton can do, apart from the inherent limits of computation. In fact, we will implement all Π_1^0-subshifts on such groups, using just three heads. The construction is similar to that in [5] and [11].

Theorem 1 *If G is finitely generated and non-torsion, then $\mathcal{S}(G, 3)$ contains every Π_1^0-subshift on G.*

Proof Let $X \subset S^G$ be a Π_1^0-subshift, and let $h \in G$ be an element of infinite order. Given a Turing machine T enumerating a list of forbidden patterns for X, we construct an automaton A_T with three heads, the *pointer head*, the *zig-zag head* and the *counter head*. The relative positions of these heads store a number, which we increment, decrement, multiply and divide by suitable constants, and test for equivalence and divisibility by constants, in order to perform arbitrary computation: such a model is Turing-complete by the results of [13].

More precisely, all heads are initialized on the same element of G, which we may assume to be 1_G. The run of the automaton proceeds in *sweeps*, each of which either corresponds to an arithmetical operation as described above, or moves the heads in some direction. Between these sweeps, the location of both the pointer head and the zig-zag head is some $g \in G$, and the position of the counter head is gh^p. The number $p \in \mathbb{N}$ is the *counter value*. Changes in the counter value are used to perform computation, and changes to the value g allow us to read the contents of every cell in the configuration.

The role of the zig-zag head is to make 'sweeps' between the pointer and counter heads, and coordinate their movement. First, to increment the counter value, the zig-zag head moves to the counter head along the progression $g, gh, gh^2, \ldots$. When the counter head is reached at the coordinate gh^p, it makes a single step to the coordinate gh^{p+1}, and the zig-zag head returns to the pointer head. The procedure for decrementing the counter is analogous.

Next, we explain how to multiply the counter value by a rational number $0 < \frac{m}{n} < 1$ assuming it is divisible by n; to multiply by a rational number greater than 1, one essentially performs the same steps in reverse. The zig-zag head again moves to the counter head, which is at gh^p, along the progression $g, gh, gh^2, \ldots$. The two heads then perform a coordinated move along the infinite path $g, gh, gh^2, \ldots$ so that they meet exactly at $gh^{\frac{m}{n}p}$. The zig-zag head then returns to the pointer head, and computation continues. We have much freedom in performing these moves, but we fix a particular scheme that works: After the zig-zag head and the counter head meet, the counter head starts moving in steps of h towards the pointer head (so that from the cell gh^j, it moves to the cell gh^{j-1} in one step), until it meets the zig-zag head again. The zig-zag head moves towards the pointer head by h^n every step, until it meets the pointer head. Note that n divides p, so that the zig-zag head indeed reaches exactly the cell g. After this, the zig-zag head starts moving back towards the counter head at speed $\frac{m}{n-m-1}$. More precisely, the zig-zag head carries a modular counter, starting at 0, and at each step it increments this counter. When the modular counter reaches $n-m-1$, the zig-zag head resets it to 0 and moves by h^m. When the zig-zag head reaches the counter head, it turns back, and returns to the pointer head. It is a simple calculation to check that the heads meet exactly at $gh^{\frac{m}{n}p}$, as required, so the counter value has been changed correctly.

Now that we can do arbitrary computation in the counter value, we give the algorithm we simulate in it. In the algorithm, objects related to the group are stored as they are output by the Turing machine: group elements are finite words over $\mathcal{G}(G)$, and patterns $P \in S^D$ are lists of pairs $(w, s) \in \mathcal{G}(G)^* \times S$ meaning $P_w = s$. We assume the Turing machine T outputs an infinite list of (possibly inconsistent) forbidden patterns, and enters the state q_{out} every time it outputs a new pattern.

The function ReadSymbol gives the symbol currently under the pointer head. The procedure MoveBy(a) causes the three heads to assume new positions: if the pointer head and zig-zag head are at g and the counter head is at gh^p, they are moved to ga and gah^p, respectively. This step is explained below. We note that there are only finitely many different messages sent between the abstract computation and A_T, namely the exchange related to ReadSymbol and the commands MoveBy(a) for finitely many $a \in G$. This information exchange can easily be performed by storing the state of the Turing machine T directly in the finite state of the pointer head.

Algorithm 1 The algorithm that the three-headed automaton A_T simulates.

$c \leftarrow c_0$ ▷ A configuration of T, set to the initial configuration
$u \leftarrow 1_G \in G$ ▷ The position of the pointer head relative to the initial position
$P : \emptyset \to S$ ▷ A finite pattern at the initial position
loop
  repeat
    $c \leftarrow \textsc{NextConf}_T(c)$ ▷ Simulate one step of T
  until $\textsc{State}(c) = q_{\text{out}}$ ▷ T outputs something
  $P' \leftarrow \textsc{OutputOf}(c)$ ▷ A forbidden pattern
  while $D(P') \not\subseteq D(P)$ **do**
    $w \leftarrow \textsc{LexMin}(D(P) \setminus D(P'))$ ▷ The lexicographically minimal element
    for $a = (u^{-1})_1, (u^{-1})_2, \ldots, (u^{-1})_{|u|}, w_1, w_2, \ldots, w_{|w|}$ **do**
      $\textsc{MoveBy}(a)$ ▷ Move all heads of A_T by group element a
    $u \leftarrow w$ ▷ New position of the pointer is w
    $b \leftarrow \textsc{ReadSymbol}$ ▷ Read the symbol of x under the pointer head
    $P \leftarrow P \cup \{u \mapsto b\}$ ▷ Expand P by one coordinate
  if $P|_{D(P')} = P'$ **then halt** ▷ The forbidden pattern P' was found

It is easy to see that this algorithm does what we want: whenever the Turing machine T enumerates a forbidden pattern P', we expand the stored pattern P by reading the configuration until its domain contains that of P'. If P' occurs in the configuration, it is eventually found by the algorithm from some starting position, and conversely, if the automaton halts, this is because it found a forbidden pattern.

To finish the proof, we explain how to perform $\textsc{MoveBy}(a)$ using the same trick we used to perform multiplications. First, the zig-zag head moves to the counter head in steps of h. Then, both heads start moving toward the pointer head. The counter head moves in steps of h^{-1}, computing the parity of p on the way, and the zig-zag head moves in steps of h^{-2}. If p is even, then the zig-zag head reaches the pointer head exactly, moves to ga, and starts moving along the sequence $ga, gah, gah^2, \ldots$ in steps of h. If p is odd, then the zig-zag head reaches the cell gh^{-1} instead, moves to gah, and starts moving in steps of h as before. The counter head performs the same task, but with the speeds reversed: after reaching the pointer head with speed h^{-1}, it starts moving from ga in steps of h^2 if p is even, and from gah in steps of h^2 if p is odd. When the counter head reaches the pointer head, the pointer head also moves to ga. It is easy to check that the counter head and the zig-zag head meet at the cell gah^p. The counter head stops, and the zig-zag head returns to the pointer head. ◻

From this theorem, Lemma 1 and the fact that all intrinsically Π^0_1 subshifts are Π^0_1 on groups with decidable word problem, we infer the following result.

Corollary 1 *If G is finitely generated and non-torsion, and has decidable group problem, then $\mathcal{S}(G,3) = \mathcal{S}(G,k)$ for all $k \geq 3$.*

Next, we give a characterization of the class of intrinsically Π^0_1 subshifts in terms of our model. The proof is essentially identical to that of Theorem 1, with only minor modifications.

Theorem 2 *If G is finitely generated and non-torsion, then the class $\mathcal{S}(G,4) = \bigcup_{k\geq 1} \mathcal{S}(G,k)$ contains exactly the intrinsically Π^0_1 subshifts on G.*

Proof All subshifts in $\bigcup_{k \geq 1} \mathcal{S}(G,k)$ are intrinsically Π^0_1, by Lemma 1. The proof that $\mathcal{S}(G,4)$ contains the intrinsically Π^0_1 subshifts is similar to that of Theorem 1, except that we must simulate a Turing machine with access to an oracle for the word problem of G. Thus, we only need to describe how one can use four heads to check whether the identity $1 \sim w$ holds for an arbitrary $w \in \mathcal{G}(G)^*$. For this, we use three heads to move by the letters of w, and leave the fourth head as a marker in the cell we started from. We return back on top of the fourth head if and only if $1 \sim w$. We can then move back by w^{-1} and pick up the fourth head. □

4 Walking on torsion groups

A torsion group is one where every element generates a finite subgroup. In this section, we show that on such groups, non-trivial sparse subshifts cannot be recognized by multi-headed automata. We also show two results about cellular automata and automorphism groups of sparse subshifts on torsion groups. These follow from a curious property, Lemma 4, of CA on sparse subshifts on torsion groups. The lemma states, intuitively, that finitely many heads can only travel a (uniformly) bounded distance on the all-zero configuration on a torsion group.

In the proof of the lemma, we use the following lemma about finite pseudometric spaces. A *pseudometric* on a set X is a function $d : X \times X \to \mathbb{R}_{\geq 0}$ that is symmetric, satisfies the triangle inequality and $d(x,x) = 0$ for all $x \in X$, and a *pseudometric space* is a topological (not necessarily Hausdorff) space with a basis given by open balls defined by a pseudometric. The diameter of a subset $Y \subset X$ is denoted $\mathrm{diam}(Y)$.

Lemma 2 *Let (X,d) be a finite pseudometric space with $|X| = k \geq 2$, and let $y \neq z \in X$. For all $c < d(y,z)/(k-1)$, there exists a partition $X = Y \cup Z$ with $y \in Y$, $z \in Z$, and $d(Y,Z) > c$.*

In particular, this can be applied to two elements $y, z \in X$ such that $d(y,z) = \mathrm{diam}(X)$.

Proof For a set $E \subset X$, write $B_E(r)$ for the closed ball of radius $r \geq 0$ around E. Let $X_1 = \{y\}$, and inductively define $X_{i+1} = B_{X_i}(c)$. For all $i \geq 1$ we have either $|X_{i+1}| > |X_i|$ or $X_{i+1} = X_i$, and in the latter case we have $X_j = X_i$ for all $j \geq i$. It follows that $X_i = X_{i+1}$ holds for some $i \leq k$.

If we have $z \in X_i$, then $d(y,z) \leq (k-1)c$, since z is in the ball $B_y((i-1)c) \subset B_y((k-1)c)$. This is a contradiction, so it must be the case that $z \notin X_i$. Then $Y = X_i$ and $Z = X \setminus X_i$ give the desired partition. □

Lemma 3 *Let $x, y \in S^G$ and let f be a CA on S^G with radius $r \in \mathbb{N}$ such that $f(0^G) = 0^G$. Suppose further that for all $n \in \mathbb{N}$, the distance between the supports of $f^n(x)$ and $f^n(y)$ is strictly more than $2r$. Then $f^n(x+y) = f^n(x) + f^n(y)$ for all $n \in \mathbb{N}$.*

Proof This follows by induction: $f^{n+1}(x+y) = f(f^n(x) + f^n(y)) = f^{n+1}(x) + f^{n+1}(y)$ since by the assumption, the supports of $f^n(x)$ and $f^n(y)$ have distance stricly more than $2r$, so every application of the local rule sees only one of the supports. □

Lemma 4 *For all torsion groups G, there exists a function $d : \mathbb{N}^3 \to \mathbb{N}$ with the following property. For all k-sparse subshifts $X \subset S^G$ over all alphabets $S \ni 0$ with $|S| = q + 1$, all cellular automata $f : X \to X$ with radius $r \in \mathbb{N}$, and all $x \in X$, we have*

$$(\exists n \in \mathbb{N} : f^n(x)_g \neq 0) \implies \exists h \in B_G(d(k, q, r)) : x_{gh} \neq 0.$$

Proof We prove the existence of such a function d by induction. We define the function so that it is monotone in all the three parameters. Let t_G be the order function and T_G the torsion function of G.

Case 1: $k = 1$

First, let $k = 1$, and let $f : X \to X$ be a CA. It is easy to show that if $x_{gh} = 0$ for all $h \in B_G(r)$, then $f(x)_g = 0$. Intuitively, this means that nonzero symbols can 'spread' by at most r per time step, and one cannot appear from nowhere. Since X is a k-sparse subshift and $k = 1$, every point $x \in X$ contains at most one nonzero coordinate $x_g \neq 0$. Intuitively, we want to give an upper bound on how far the nonzero symbol can travel from its initial position g.

By shift-commutation, it is enough to analyze the case $x_{1_G} \neq 0$. Combining the previous observations and the fact $|S| = q + 1$, it follows from the pigeonhole principle that $f^{n+m}(x) = h \cdot f^n(x)$ for some $0 \leq n < n + m \leq q + 1$ and $h \in B_G((q + 1)r)$.[5] Since f commutes with the shift, we have $f^{n+\ell m}(x) = h^\ell \cdot f^n(x)$ for all $\ell \in \mathbb{N}$. Since $h^{t_G(h)} = 1_G$, we have $f^{n+t_G(h)m}(x) = f^n(x)$. We have shown that $f^j(x)_{h'} \neq 0$ for some $j \in \mathbb{N}$ implies $h' \in B_G((q + 1)r(1 + t_G(h)))$. Since $h \in B_G((q + 1)r)$, we can define

$$d(1, q, r) = (q + 1)r(1 + T_G((q + 1)r)).$$

Case 2: $k > 1$

Next, consider the case $k > 1$. To each configuration $x \in X$, we associate the metric space $A(x)$ whose points are the nonzero coordinates of x, and whose distances are those induced by the natural (right) distance in G. We will split the analysis of the dynamics of f on the point x into two cases, depending on whether the 'diameter' of the configuration $f^n(x)$ stays bounded (by an explicit constant) as n grows.

Intuitively, the idea is that as long as the diameter stays small, we can shrink all the information in x into a single symbol, reducing to the case $d(1, \cdot, \cdot)$, and if the configuration starts expanding, then it splits into two pieces that can never again communicate, and we apply induction to these smaller pieces.

More precisely, consider the metric space $A(x) = (\{g \in G \mid x_g \neq 0\}, d_G)$. Define $c = 2d(k - 1, q, r) + 2r$, and note that since d is monotone, we in particular have

$$c \geq \max_{1 \leq \ell < k} d(\ell, q, r) + d(k - \ell, q, r) + 2r.$$

We say that a configuration $x \in X$ is *clustered* if $\operatorname{diam}(A(x)) \leq (k - 1)c$ holds, and *scattered* otherwise.

[5] Note that here we use the fact that $x_{1_G} \neq 0$. In general, we can only conclude $f^{n+m}(x) = \sigma^R_h(f^n(x))$ for some $h \in B_G((q + 1)r)$.

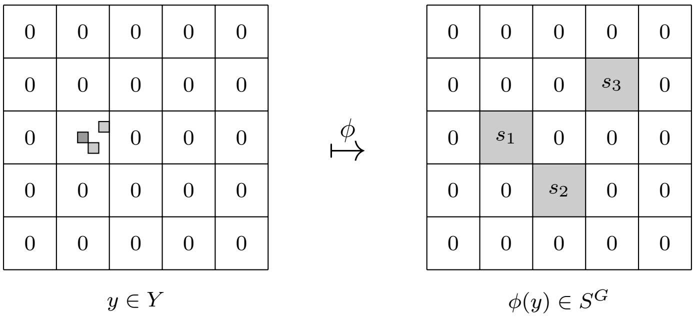

Fig. 1 The decompression function ϕ applied to a configuration $y \in Y$. We have chosen $G = \mathbb{Z}^2$ here for simplicity, even though it is not a torsion group. Note that the alphabet of Y consists of certain patterns of X and the symbol 0.

Suppose $x \in X$ and $N \in \mathbb{N}$ are such that $f^n(x)$ is clustered for all $n \leq N$. We will give an upper bound on how far nonzero symbols can travel from their original positions in these N steps. Let $Z \subset X$ be the subshift generated by the configurations $f^n(x)$ for $n \leq N$. It is easy to see that every configuration of Z is clustered. Note that the subshift Z may not be closed under f.

Let $Y = X^1_{\{0\} \cup K}$, where $K \subset \mathcal{L}_{B_G((k-1)c)}(Z)$ is the set of patterns P of shape $B_G((k-1)c)$ occurring in Z such that $P_{1_G} \neq 0$. Clearly, Y is a 1-sparse subshift, and it should be thought of as a 'compressed' version of Z, where all the nonzero symbols have been encoded into a single coordinate. The idea is to simulate the CA f on the compressed subshift Y, and reduce back to the $k = 1$ case. Let $\phi : Y \to S^G$ be the 'decompression function' defined by

$$\phi(y)_h = \begin{cases} (y_g)_{g^{-1}h}, & \text{if } \exists g \in G : y_g \neq 0 \wedge g^{-1}h \in B_G((k-1)c), \\ 0, & \text{otherwise.} \end{cases}$$

Let $Y' = \phi^{-1}(Z)$, so that $\phi : Y' \to Z$ is a surjective block map.[6] A visualization of ϕ is shown in Figure 1.

Claim There exists a (not necessarily unique) cellular automaton $f_\phi : Y' \to Y'$ such that $\phi(f_\phi(y)) = f(\phi(y))$ holds for all $y \in Y'$ such that $f(\phi(y)) \in Z$.

Intuitively, the CA f_ϕ simulates f on the compressed configurations of Y', as long as their ϕ-images are clustered.

Proof (of claim) Observe that for each $z \in Z$ and $g \in G$ there is at most one configuration $y \in Y'$ such that $y_g \neq 0$ and $\phi(y) = z$. To define a map f_ϕ with the desired properties, we need a way to pick a group element in every subset of the group contained in a ball of radius $(k-1)c + r$. For this, one can use any choice function

$$\psi : \{D \mid \emptyset \subsetneq D \subset B_G((k-1)c + r)\} \to B_G((k-1)c + r)$$

[6] The fact that G is torsion prevents us, in general, from defining a bijective version of ϕ. Also, the subshift Y' may be strictly smaller than Y.

satisfying $\psi(D) \in D$ for all $D \subset B_G((k-1)c+r)$. For this purpose, we use a total order $h_1 < h_2 < h_3 < \cdots$ on G, not necessarily in any way compatible with its algebraic structure, and take $\psi(A)$ to be the least element of A.

First, for the all-0 configuration $0^G \in Y'$, we define $f_\phi(0^G) = 0^G$, and for all $y \in Y'$ such that $f(\phi(y)) \notin Z$, we also define $f'(y) = 0^G$. For all other $y \in Y'$, let $g \in G$ be the unique element with $y_g \neq 0$, and let $W \subset Y'$ be the set of configurations $y' \in Y'$ with $\phi(y') = f(\phi(y))$. The set W is nonempty since $\phi : Y' \to Z$ is surjective, and it is finite because the unique nonzero coordinate of each $y' \in W$ is among the coordinates gh where $h \in B_G((k-1)c+r)$, since we assumed $P_{1_G} \neq 0$ for each $P \in K$. Now, we choose $f_\phi(y)$ to be the unique configuration $y' \in W$ with $y'_{gh} \neq 0$, where $h \in G$ is minimal in the ordering $h_1 < h_2 < \cdots$. It is easy to check that f_ϕ is then continuous and shift-commuting. In fact, from the way we defined it, we see that its radius is at most $(k-1)c+r$. □

Recall our assumption that $x \in X$ and $f^n(x)$ is clustered for all $n \leq N$. We have $x \in Z$ by the definition of Z, so there exists a configuration $y \in Y'$ such that $\phi(y) = x$. By the above claim, we have $\phi(f_\phi^n(y)) = f^n(x)$ for all $n \leq N$. Since Y is a 1-sparse subshift with alphabet of size $|K|+1$ and f_ϕ is a CA on it with radius at most $(k-1)c+r$, we have

$$(\exists n : f_\phi^n(y)_g \neq 0) \implies \exists h \in B_G(d(1,|K|,(k-1)c+r)) : y_{gh} \neq 0 \tag{1}$$

by Case 1 of this proof. We also remark that if we have $N > |K|$, then the configuration $f^n(x)$ is clustered for all $n \in \mathbb{N}$, since there exist $i < j \leq N$ such that $f_\phi^i(\phi(y))$ is a translated version of $f_\phi^j(\phi(y))$.

Subcase 2.1: clustered configurations

Now, consider the case when $f^n(x)$ is clustered for all $n \in \mathbb{N}$. Suppose that $f^n(x)_g \neq 0$ for some $g \in G$. Since the block map ϕ has radius $(k-1)c$, we have $\phi(f^n(x))_{gh'} = f_\phi^n(y)_{gh'} \neq 0$ for some $h' \in B_G((k-1)c)$. Equation (1) implies that $y_{gh'h} \neq 0$ for some $h \in B_G(d(1,|K|,(k-1)c+r))$, and from the definition of ϕ it follows that $x_{gh'h} \neq 0$ as well, since $(y_{gh'h})_{1_G} \neq 0$. We have shown that if $f^n(x)$ contains a nonzero symbol in some coordinate, then there is a nonzero coordinate in x at distance at most $d(1,|K|,(k-1)c+r)+(k-1)c$. Note that the cardinality of K is at most doubly exponential in $(k-1)c$ and $c = 2d(k-1,q,r)+2r$, so this formula only depends on k, q and r.

Subcase 2.2: scattered configurations

Suppose finally that the configuration $f^n(x)$ is scattered for some $n \in \mathbb{N}$, which we assume to be minimal. As remarked above, we must have $n \leq |K|$, as otherwise there exist $i < j \leq n$ such that $f_\phi^i(\phi(y))$ is a translated version of $f_\phi^j(\phi(y))$. We can then apply Lemma 2 to the metric space $A(f^n(x))$, and obtain a partition for it into sets $C, D \subset G$ with distance more than c.

Denote $y = f^n(x)$. We define a partition of the configuration y by $y = y_C + y_D$, where $(y_C)_g = y_g$ when $g \in C$ and $(y_C)_g = 0$ otherwise, and y_D is defined analogously. By the definition of c, we have $c \geq d(|C|,q,r)+d(|D|,q,r)+2r$. Then by Lemma 3 we have $f^j(y) = f^j(y_C)+f^j(y_D)$ for all $j \in \mathbb{N}$. In particular, if we have $f^j(y)_g \neq 0$ for some $j \in \mathbb{N}$ and $g \in G$, then $y_{gh} \neq 0$ for some $h \in B_G(\max_{\ell<k} d(\ell,q,r)) \subset B_G(d(k-1,q,r))$ by the induction hypothesis. Since we

have $n \leq |K|$ and the CA f has radius r, this implies that $x_{ghh'} \neq 0$ for some $h' \in B_G(r|K|)$, which implies $hh' \in B_G(r|K| + d(k-1,q,r))$.

Putting all three cases together, we can define the function d recursively by

$$d(k,q,r) = d(1,|K|,(k-1)c+r) + (k-1)c + r|K| + d(k-1,q,r)$$

for all $k > 1$. □

The bounds we give are not very strong, but at least one can check that if the torsion function T_G is primitive recursive (computable, respectively), then so is the function d. For our model, Lemma 4 implies the following result about infinite torsion groups.

Theorem 3 *If G is finitely generated, infinite and torsion, and $X \subset S^G$ is sparse and nontrivial, then $X \notin \mathcal{S}(G)$.*

Proof Let A be a group-walking automaton and Y its associated subshift, and let $X' = \{x + y \mid x, y \in X, \forall g \in G : x_g = 0 \vee y_g = 0\}$. Since $X' \times Y$ is sparse, Lemma 4 implies that any head of A can only travel a bounded distance on any configuration of $X' \times Y$. Then, for all $x \in X$ and all but finitely many $g \in G$, the configuration $x + (g \cdot x)$ is rejected by A if and only if x is. If the support of x is maximal, this configuration is not in X. Thus A does not define X. □

As an aside, we mention that Lemma 4 also has a corollary for automorphism groups of sparse subshifts on torsion groups.

Theorem 4 *If G is torsion and $X \subset S^G$ is sparse, then $\mathrm{Aut}(X)$ is also torsion.*

The last theorem has a converse: If G is not torsion and $X \subset S^G$ is sparse and nontrivial, then there is an automorphism of X that shifts an isolated point of X along a copy of $\mathbb{Z}$. If the right shifts σ_g^R are well-defined on X is suffices to take σ_g^R for g of infinite order.

5 Infinite Hierarchy

We showed in Theorem 2 that the hierarchy $\mathcal{S}(G,k)_{k \geq 1}$ collapses to the fourth level whenever G is not a torsion group. We now prove the converse of this, obtaining a characterization of torsion groups.

Theorem 5 *Let G be a finitely generated infinite torsion group. Then the hierarchy $\mathcal{S}(G,k)_{k \geq 1}$ is infinite.*

It was proved already in [3] that the head hierarchy of finite automata on two-dimensional finite grids is infinite. The main idea of the proof is diagonalization: an automaton with many heads can simulate all automata with few heads and behave differently from them on at least one input. Our proof is also based in diagonalization, but in order to apply it, we need to prove some additional restrictions that apply to automata on torsion groups.

We begin by proving a strengthening of Lemma 4, and for that, we need some auxiliary definitions. For two nonempty sets $A, B \subset G$, we denote $d_G(A,B) = \min\{d_G(a,b) \mid a \in A, b \in B\}$, and similarly $d_G(g,B) = \min\{d_G(g,b) \mid b \in B\}$ for

$g \in G$. Note that this does not define a pseudometric on subsets of G, as the triangle inequality does not hold in general. For two configurations $x, y \in S^G$ and $a \in S$, denote by

$$d_{\max}(y \xrightarrow{a} x) = \max_{g \in \mathrm{supp}_a(y)} d_G(g, \mathrm{supp}_a(x))$$

the maximum distance between a non-a coordinate of y and the a-support of x. Also, fix a function $\phi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $\phi(1, N) = 4N$ and $\phi(k, N) > k\phi(k-1, N) + (3k+4)N$ for $k > 1$ and $N \in \mathbb{N}$.

Lemma 5 *Let G be a torsion group, and let $0 \in S$. Let $A = (Q, f, I, F, S)$ be a k-headed automaton with radius $r \geq 0$, and let $N \geq d(k, |Q|, r) \geq r$ as defined in Lemma 4. Let $x \in S^G$ be an arbitrary configuration, and let $y \in Y$ be a configuration in the associated subshift of A. Then for all $n \in \mathbb{N}$, for the configuration $y^n =: \pi_2(f^n(x, y))$ it holds that $d_{\max}(y^n \xrightarrow{0} x) \leq M =: \max(\phi(k, N), d_{\max}(y \xrightarrow{0} x) + N)$.*

The intuition behind this lemma is that the heads of a group-walking automaton cannot travel arbitrarily far away from the region of non-0 coordinates (unless some heads are far away from this region in the first place). Note that $d_{\max}(y^n \xrightarrow{0} x)$ is the maximum distance of a head of A from a non-0 coordinate of x at time n, so $d_{\max}(y^n \xrightarrow{0} x) \leq M$ means that no head of A is further than M steps away from a non-0 coordinate. We will later use this result to simulate a given automaton by one that never steps from a non-0 coordinate to one containing a 0.

Proof Suppose first that $d_G(\mathrm{supp}_0(x), \mathrm{supp}_0(y)) > 2N$, so that the heads of A are already far from the 0-support of x. We define $y^m =: \pi_2(f^m(x, y))$, and note that $y^m = \pi_2(f^m(0^G, y))$ for all $m \geq 1$ such that $d_G(\mathrm{supp}_0(x), \mathrm{supp}_0(y^{m-1})) > N$, since the radius of f is $r \leq N$. By an inductive argument with Lemma 4 applied to the k-sparse subshift $0^G \times Y$, this holds for all $m \geq 1$, and we have $d_{\max}(y^m \xrightarrow{0} x) \leq d_{\max}(y \xrightarrow{0} x) + d(k, |Q|, r) \leq d_{\max}(y \xrightarrow{0} x) + N$ for all $m \geq 0$. Thus the claim holds in this case.

Suppose then that $d_G(\mathrm{supp}_0(x), \mathrm{supp}_0(y)) \leq 2N$, and assume for contradiction that $d_{\max}(y^n \xrightarrow{0} x) > M$ holds for some $n \in \mathbb{N}$. We proceed by induction on k, the number of heads of A. If $k = 1$, then there exists $m < n$ with $2N < d_{\max}(y^m \xrightarrow{0} x) \leq 3N$, and applying the argument of the first paragraph to y^m gives $d_{\max}(y^n \xrightarrow{0} x) \leq 4N = M$.

Next, suppose that $k > 1$, and let $m < n$ be such that $M - 2N < d_{\max}(y^m \xrightarrow{0} x) \leq M - N$. The time step m exists, since $d_{\max}(y^{m+1} \xrightarrow{0} x) \leq d_{\max}(y^m \xrightarrow{0} x) + r$ for all $m \in \mathbb{N}$ and $d_{\max}(y \xrightarrow{0} x) \leq M - N$. If we have $d_G(\mathrm{supp}_0(x), \mathrm{supp}_0(y^m)) > 2N$, then the first paragraph applied to y^m again gives $d_{\max}(y^n \xrightarrow{0} x) \leq \phi(k, N)$.

Suppose then that $d_G(g, \mathrm{supp}_0(x)) \leq 2N$ for some $g \in \mathrm{supp}_0(y^m)$, and let $X = \mathrm{supp}_0(y^m)$. Then (X, d_G) is a pseudometric space with $|X| \leq k$, and Lemma 2 applied to the coordinate g and the coordinate $h \in \mathrm{supp}_0(y^m)$ with $d_G(h, \mathrm{supp}_0(x)) > M - 2N \geq \phi(k, N) - 2N$ gives a partition of X into two components X^g, X^h such that $g \in X^g$, $h \in X^h$ and $d_G(X^g, X^h) \geq (\phi(k, N) - 4N)/k > \phi(k-1, N) + 3N$. This gives a partition $y^m = y^{m,g} + y^{m,h}$ for the configuration y^m in the obvious way.

We claim that if X^h is chosen to be minimal, then $d_G(\mathrm{supp}_0(x), X^h) > 2N$. Namely, if X^h is minimal, then for every $h' \in X^h$ there exists a sequence $h =$

$h_1, h_2, \ldots, h_s = h'$ such that $s < k$ and $d_G(h_i, h_{i+1}) < (\phi(k,N) - 4N)/k$ for every i. Then $d_G(h, h') < \phi(k,N) - 4N$, which implies $d_G(h', g') > d_G(h, g') - d_G(h', h) > \phi(k,N) - 2N - \phi(k,N) + 4N = 2N$ for all $g' \in \mathrm{supp}_0(x)$.

Denote $y^{m+p,g} = \pi_2(f^p(x, y^{m,g}))$ for $p \geq 0$, and similarly for h. By Lemma 4, we have $\mathrm{supp}_0(y^{m+p,h}) \subset \mathrm{supp}_0(y^{m,h}) \cdot B_G(N)$, and in particular $d_{\max}(y^{m+p,h} \xrightarrow{0} x) \leq M$ holds for all $p \geq 0$. Next, we apply the induction hypothesis to the configuration $y^{m,g}$, and obtain

$$d_{\max}(y^{m+p,g} \xrightarrow{0} x) \leq \max(\phi(k-1,N), d_{\max}(y^{m,g} \xrightarrow{0} x) + N) < M$$

for all $p \geq 0$. By induction on p, we can now show

$$d_G(\mathrm{supp}_0(y^{g,m+p}), \mathrm{supp}_0(y^{h,m+p})) > N$$

for all $p \geq 0$, so that $y^{m+p} = y^{g,m+p} + y^{h,m+p}$. Since $d_{\max}(y^{m+p} \xrightarrow{0} x) \leq \max(d_{\max}(y^{g,m+p} \xrightarrow{0} x), d_{\max}(y^{h,m+p} \xrightarrow{0} x))$, this finishes the proof. □

Next, we define geodesics of finitely generated groups, which we use to embed finite one-dimensional languages into subshifts on torsion groups.

Definition 5 Let G be a finitely generated group, and let $p \in G^*$ be a word of length n over G. We say that p is a *path*, if $p_i^{-1} p_{i+1} \in \mathcal{G}(G)$ holds for all $i \in \{0, \ldots, n-2\}$, and it is a *geodesic*, if $d_G(p_0, p_{n-1}) = n - 1$. The set of geodesics of G is denoted $\mathrm{Geo}(G)$.

Fix an alphabet S and some symbol $\# \notin S$, and denote $S_{(G)} = S \times \mathcal{G}(G) \cup \{\#\}$. For a path $p \in G^n$ that has no repeated elements and $w \in S^n$, define the configuration $x(p,w) \in S_{(G)}^G$ by $x_{p_i} = (w_i, p_i^{-1} p_{i+1})$ for all $i \in \{0, \ldots, n-2\}$, $x_{p_{n-1}} = (w_{n-1}, 1_G)$, and $x_g = \#$ for all other $g \in G$. For a language $L \subset S^*$, the *G-geodesic subshift of* L, denoted X_G^L, is the topological closure of

$$\{x(p,w) \mid p \in \mathrm{Geo}(G), w \in L, |p| = |w|\} \tag{2}$$

which is a shift-invariant set.

We note some properties of geodesics and X_G^L. First, every contiguous subsequence of a geodesic is itself a geodesic, and arbitrarily long geodesics can be found in all infinite groups. Second, the only elements of X_G^L with finitely many non-$\#$ coordinates are exactly those in (2). The following observation simplifies the proof of the following lemma.

Lemma 6 *Let G be an infinite finitely generated group. If some configuration in X_G^L does not contain arbitrarily large balls containing only $\#$, then G is virtually $\mathbb{Z}$.*

Proof Suppose X_G^L contains a configuration not containing arbitrarily large balls of all $\#$. Then in G, there is an infinite geodesic p such that every $g \in G$ is at distance at most $m \in \mathbb{N}$ from some element in p. It is easy to see that the number of elements at distance $n \in \mathbb{N}$ from the origin of G is at most $(2(n+m)+1)|B_G(m)|$. In particular, the group has linear volume growth in the sense of [14], and thus is virtually $\mathbb{Z}$. □

Lemma 7 *Let G be an infinite finitely generated torsion group and S a finite alphabet. For each k-headed automaton $A = (Q, f, I, F, S_{(G)})$, there exists a k-headed automaton $\hat{A} = (\hat{Q}, \hat{f}, \hat{I}, \hat{F}, S_{(G)})$ with $X_G^{S^*} \cap \mathcal{S}(A) = X_G^{S^*} \cap \mathcal{S}(\hat{A})$ such that the following conditions hold.*

1. *Every initial configuration $y \in \hat{I}$ of $\hat{A}$ has all heads at 1_G, and if $x \in X_G^{S^*}$ is such that $x_{1_G} = \#$, then $\pi_2(\hat{f}^n(x, y)) \notin G \cdot \hat{F}$ for all $n \in \mathbb{N}$.*
2. *For any configuration $x \in X_G^{S^*}$ such that $x_{1_G} \neq \#$ and any $y \in \hat{I}$, we have $\mathrm{supp}_0(\pi_2(\hat{f}^n(x, y))) \subset \mathrm{supp}_\#(x)$ for all $n \in \mathbb{N}$.*

The essence of this lemma is that, on a torsion group, every automaton A can be turned into an automaton $\hat{A}$ whose heads never leave the support of the configuration, so that on the configurations we are interested in, namely $X_G^{S^*}$, the automaton $\hat{A}$ simulates the behavior of A.

Proof Let $N \geq d(k, |Q|, r) \geq r$ be given by Lemma 4 for A, and assume without loss of generality that every initial configuration $y \in I$ satisfies $\mathrm{supp}_0(y) \subset B_G(N)$. We may also assume that $X_G^{S^*} \cap \mathcal{S}(A) \neq \emptyset$.

We first construct an automaton $\tilde{A} = (\tilde{Q}, \tilde{f}, \tilde{I}, \tilde{F}, S_{(G)})$ that satisfies condition 1. For that, define the state sets of $\tilde{A}$ as $\tilde{Q}_i = Q_i \cup \{q_i^{y,g} \mid y \in I, g \in B_G(3N)\}$, and the initial set $\tilde{I}$ as the collection of clopen sets of configurations with $(q_i^{y,g})_{i=1}^k$ at the origin, where $y \in I$ and $g \in B_G(3N)$ are the same for all coordinates $i \in \{1, \ldots, k\}$. The update function $\tilde{f}$ behaves exactly as f on the states Q_i. Suppose then that head i is in one of the new states $q_i^{y,g}$ at coordinate $h \in G$, where $g \in B_G(3N)$ and $y \in I$ contains head i at coordinate $g_i \in B_G(N)$. If $x_h \neq \#$, then the head will step to the coordinate hgg_i in state q_i, and if $x_h = \#$, it will retain its state and position. This behavior is visualized in Figure 2. Finally, the final set $\tilde{F}$ is exactly F.

It is clear by construction that $\tilde{A}$ satisfies condition 1. Let us show that $X_G^{S^*} \cap \mathcal{S}(A) = X_G^{S^*} \cap \mathcal{S}(\tilde{A})$ holds, and for that, let $x \in X_G^{S^*}$ be arbitrary. If $x \in \mathcal{S}(A)$, then clearly $x \in \mathcal{S}(\tilde{A})$, since every accepting initial configuration $y \in I$ and $g \in B_G(3N)$ give rise to an accepting initial configuration of $\tilde{A}$: the run of $\tilde{A}$ either simulates the accepting run of A, or the heads never leave the states $q_i^{y,g}$. Now, let $y \in I$ and $g \in G$ be such that the run $\pi_2(f^n(x, g \cdot y)) \in G \cdot F$ for some $n \geq 0$. If $d_G(g, \mathrm{supp}_\#(x)) \leq 3N$, then the initial configuration $\tilde{y} \in \tilde{I}$ containing the states $q_i^{y,h^{-1}}$, where $h \in B_G(3N)$ is such that $x_{gh} \neq \#$, satisfies $f(x, gh \cdot \tilde{y}) = (x, g \cdot y)$, and thus leads to a rejecting run of $\tilde{A}$. On the other hand, if we have $d_G(g, \mathrm{supp}_\#(x)) > 3N$, then $d_G(gg_i, \mathrm{supp}_\#(x)) > 2N$ for all heads i, and Lemma 4 implies that none of the heads of A ever come within distance N of $\mathrm{supp}_\#(x)$. By Lemma 6, every configuration of $X_G^{S^*}$ contains arbitrarily large areas of #s, so since the run of A is rejecting, we have $X_G^{S^*} \cap \mathcal{S}(A) = \emptyset$, contradicting our earlier assumption. Thus the case $d_G(g, \mathrm{supp}_\#(x)) > 3N$ is impossible, and we have shown $X_G^{S^*} \cap \mathcal{S}(A) = X_G^{S^*} \cap \mathcal{S}(\tilde{A})$.

Let us now define the automaton $\hat{A}$. We define the state sets by $\hat{Q}_i = (\tilde{Q}_i \times B_G(M)) \cup \{e\}$, where $M = \phi(k, N) + N$ and e is a new *error state*. The idea is that a head of $\hat{A}$ at $g \in G$ in state (q, h) represents the corresponding head of $\tilde{A}$ at gh in state q. The initial configurations are defined from those of $\tilde{A}$ by having each head in state $(q, 1_G)$ instead of q, and the final configurations are defined by having the simulated heads in some final configuration of $\tilde{A}$.

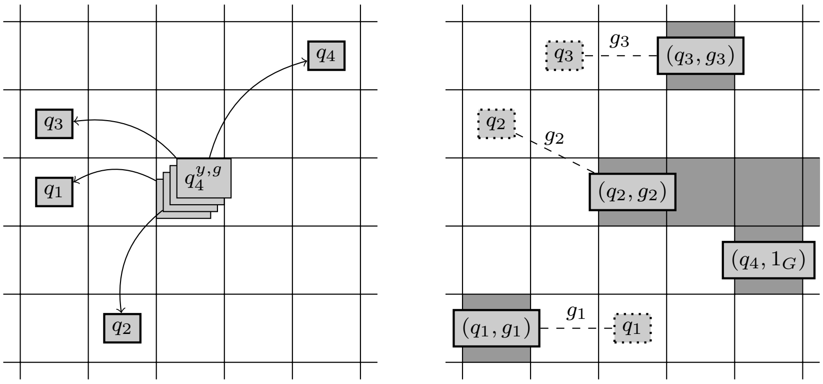

Fig. 2 A visualization of the proof of Lemma 7, with automaton $\tilde{A}$ on the left and $\hat{A}$ on the right. The figures do not depict the same configuration.

We do not define the update function $\hat{f}$ explicitly, but instead give an intuitive explanation of how it behaves on a configuration $z = (x, y) \in X_G^{S^*} \times Y$. From the simulated states and positions of the heads of $\tilde{A}$ in y, one can compute the simulated states q_i and positions g_i of the heads of $\tilde{A}$ on the next time step. For each i, the function $\hat{f}$ places the actual head at the coordinate $h_i \in g_i \cdot B_G(M)$ for which the distance $d_G(g_i, h_i)$ is minimal (with ties being broken in some consistent way). The head will be in state $(q_i, h_i^{-1} g_i)$, so it is correctly simulating the head at g_i. If such a coordinate does not exist, the head i retains its position and enters the error state e in the image $f(z)$.

Lemma 5 implies that for an initial configuration $g \cdot y$, where $g \in \mathrm{supp}_\#(x)$ and $y \in \tilde{I}$, we have $d_0(\tilde{f}^n(x, g \cdot y)) \leq M$ for all $n \geq 0$. This means that the head of $\hat{A}$ will always stay within distance M of the simulated head of $\tilde{A}$, if they are initialized on the same non-# coordinate, and thus no head of $\hat{A}$ will enter the error state e. For initial configurations $g \cdot y$ where $g \notin \mathrm{supp}_\#(x)$, the simulated automaton $\tilde{A}$ never enters a final configuration by construction, and neither does $\hat{A}$, as its heads either correctly simulate $\tilde{A}$ or enter the error state e. Since the final set $\hat{F}$ is defined in terms of the simulated heads, this implies $\mathcal{S}(\tilde{A}) = \mathcal{S}(\hat{A})$. The second condition now holds by construction of $\hat{A}$, and the first condition is inherited from $\tilde{A}$. □

We note that Lemma 5 and Lemma 7 depend on the fact that the heads of our automata cannot communicate over arbitrarily long distances, and our proofs fail if this is allowed. See Section 8 for an extended discussion. The proofs of the lemmas also rely on Lemma 4, which requires G to be a torsion group.

Finally, we are ready to prove Theorem 5. Our proof uses the same diagonalization technique as the main result of [3], that the head hierarchy of multi-headed finite automata on one-dimensional finite words is infinite. The main idea is that Lemma 7 allows us to consider group-walking automata on geodesic subshifts X_G^L essentially as one-dimensional finite automata, where $L \subset S^*$ is an arbitrary one-dimensional language. We choose L as a language that describes arbitrary k-headed group-walking automata, and then define an n-headed automaton (for some $n > k$) that 'interprets' this language, simulates the k-headed automaton A_w that a word

$w \in L$ represents, and rejects the configuration containing w if and only if A_w does not reject it.

Proof (of Theorem 5) Fix $k \geq 1$. We begin by defining a one-dimensional language $L_k \subset S^*$ that describes k-headed group-walking automata on S^G with arbitrarily large state sets. The alphabet of L_k is $S = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\} \cup \mathcal{G}(G)$, where $\mathcal{G}(G)$ is the finite symmetric generating set of G. Pick an arbitrary k-headed radius-r automaton $A = (Q, f, I, F, S_{(G)})$ on $S^G_{(G)}$. Since we are interested in the set $\mathcal{S}(A) \cap X_G^{S^*}$ of geodesic configurations that A does not reject, we may assume that the state sets of its heads are $Q_i = \{1, \ldots, |Q_i|\}$, and that A satisfies the conditions of Lemma 7. In particular, the heads of A never leave the geodesic path on a configuration of $X_G^{S^*}$, and are always initialized at the same coordinate.

We now inductively define the encoding $\xi(A) \in S^*$ of the automaton A. Let $T = \prod_{i=1}^k (Q_i \cup \{0\})$ be the alphabet of the associated subshift Y. A state $q \in Q_i \cup \{0\}$ is encoded as the word $\xi(q) = \mathtt{a}^q\mathtt{b}$, a word $w = w_1 \cdots w_n \in \mathcal{G}(G)^n$ is encoded as $\xi(w) = w_1\mathtt{a}w_2\mathtt{a} \cdots \mathtt{a}w_n\mathtt{b}$, a symbol $(e, q) \in S_{(G)} \times T$ is encoded as $\xi(e, q) = sg\xi(q_1) \cdots \xi(q_k)$ if $e = (s, g) \in S \times \mathcal{G}(G)$ and $\xi(e) = \xi(q_1) \cdots \xi(q_k)$ if $e = \#$, and a pattern $P \in (S_{(G)} \times T)^{B_G(r)}$ is encoded as the concatenation of $\xi(w)\xi(P_w)$ for all words $w \in \mathcal{G}(G)^{\leq r}$. The complete encoding $\xi(A)$ of A consists of the string $\mathtt{a}^{\max_i |Q_i|}$, the encoding of every transition $(P, f(P))$ for every input pattern $P \in (S \times T)^{B_G(r)}$ and every possible way of writing each coordinate $g \in B_G(r)$ as a geodesic word $w \in \mathcal{G}(G)^*$, and the encoding of every tuple $(q_i, g_i)_{i=1}^k$ of states and translations in the initial and final sets I and F, all delimited by the symbols $\mathtt{b}$ in some unambiguous way. The language L_k consists of the strings $\xi(A)\mathtt{c}^n$ for all k-head automata A and all $n \geq 0$.

Let us describe a $(3k+2+C)$-headed automaton $\hat{A}$, where C is a finite constant left undetermined, with the property that for all k-head automata A, there exists some configuration $x = x(p, w) \in X_G^{L_k}$ with a finite-length geodesic path p such that exactly one of A and $\hat{A}$ rejects x. We will give a high-level description of $\hat{A}$, as the detailed state set and local rule would be very complicated. However, it should not be hard to see that the described behavior can be implemented by finite state sets and local interactions.

The idea is that each word $w = \xi(A)\mathtt{c}^n \in L_k$ for $n \geq 0$ contains the encoding of the automaton A, and $\hat{A}$ uses this encoding to simulate A from different initial configurations until it can decide whether $x \in \mathcal{S}(A)$. The automaton $\hat{A}$ has k *simulation heads*, k *state heads*, $k + 1$ *clock heads*, an *initial head*, and C *auxiliary heads*. The heads will always be initialized on the same coordinate, and every coordinate except the beginning $p_0 \in G$ of the path p leads to an accepting run.

If the heads are initialized on the coordinate p_0, then they perform as follows. First, only the auxiliary heads can move independently, and the other heads are controlled by them. Each simulation head corresponds directly to a simulated head of A, in the sense that the position of the simulation head matches that of the simulated head. The state heads stay on the first $\max_i |Q_i|$ symbols of the path. Each of them corresponds to the internal state of one simulated head, in the sense that the ith state head on the qth $\mathtt{a}$-symbol corresponds to the state $q \in Q_i$.

The C auxiliary heads perform the computations required by the simulation. One of the auxiliary heads is called the *neighborhood head*. In the beginning of a simulation step, the neighborhood head is situated on the first coordinate p_0 of the path. Other auxiliary heads scan the r-ball around p_0, and check whether its contents match a pattern $P \in (S_{(G)} \times T)^{B_G(r)}$ encoded in the word w. Note that we only need to consider the intersection of said ball with the path p, since the complement of p contains only #-symbols in the configuration x, and the simulated heads never leave the path. This can be implemented with six auxiliary heads as follows. One head reads the generator symbols from the encoding of P, while a second head walks along them on the path p, guided by a third head that travels back and forth between them. If the second head encounters a simulation head that is also present in the pattern P, three additional heads can check whether its state matches the one in P: a fourth head advances along the sequence of $\mathtt{a}$-symbols in the encoding of P and a fifth one along the sequence at the beginning of p where the corresponding state head lies, synchronized by a sixth head that travels between them, verifying that the sequences have the same length. Since the coordinates of the pattern P are encoded as a geodesic sequence of generators in every possible way, there exists one encoding of P in w such that the second head will not have to leave the path in order to reach a coordinate that contains a non-# symbol in P. When a matching encoding is found, if the transition result $f(P) \in S \times T$ contains any heads of A, then the corresponding simulation heads are moved to coordinate of the neighborhood head, and the corresponding state heads are moved to the correct positions on the initial string of $\mathtt{a}$-symbols. We may assume that updating the heads one at a time will not cause any problems; for example, we may modify A so that on even-numbered steps, every head just stores its surroundings in its internal state, and the head positions are updated on odd-numbered turns.

The above process is repeated for every coordinate of the path; when the correct pattern P has been found and all required simulation and state heads have been moved to their new positions, the neighborhood head advances one step along the path. When the last coordinate of the path has been processed, the neighborhood head returns to p_0. Next, the auxiliary heads check whether a final (rejecting) configuration is reached, using the same technique as above. If this is the case, $\hat{A}$ accepts the configuration x by entering simple looping state.

If a rejecting configuration is not reached, then the $k + 1$ clock heads are updated. After each simulated step of A, the first clock head advances one step along p, guided by an auxiliary head. If it is the last coordinate of p, it immediately returns to the beginning, after which the second head advances one step. When the second clock head reaches the end of p, it also returns to the beginning, and the third head advances one step. The remaining clock heads behave similarly. After this, a new simulation step begins.

Now, notice that the clock heads will return to their original positions after $|p|^{k+1}$ simulated steps, where $|p|$ is the length of the path p. If the simulation goes on for the $|p|^{k+1}$ steps without reaching a final configuration, it is terminated, the initial head is advanced by one step along p, and a new simulation is initialized from that coordinate. If the initial head reaches the end of the path and cannot advance further, then $\hat{A}$ rejects the configuration x. This concludes the description of $\hat{A}$.

We claim that if n (and thus $|p|$) is large enough, then $x \in \mathcal{S}(\hat{A})$ if and only if $x \notin \mathcal{S}(A)$. First, if we have $x \in \mathcal{S}(A)$, then A does not reject the configuration, every simulated run of A goes on for $|p|^{k+1}$ steps, and $\hat{A}$ eventually rejects, so $x \notin \mathcal{S}(\hat{A})$. Suppose then that $x \notin \mathcal{S}(A)$, so there is a rejecting run of A on x starting from some coordinate $g \in G$. By our assumption on A, we have $g = p_i$

for some $i \in \{0, \dots, |p| - 1\}$. Since A has k heads that never leave the path p, any rejecting run of A contains at most $\prod_{i=1}^{k} |Q_i| \cdot |p| = |p|^k \prod_{i=1}^{k} |Q_i|$ different configurations, and in particular enters a rejecting configuration after at most this many steps. If $|p| > \prod_{i=1}^{k} |Q_i|$, then $\hat{A}$ will simulate A from the coordinate p_i (or some other rejecting coordinate) long enough for A to reject x, and will itself accept it, which implies $x \in \mathcal{S}(\hat{A})$. Since n can be arbitrarily large, this finishes the proof. □

6 Characterizations of torsion groups

Combining the results of the previous sections, we obtain some characterizations of torsion groups.

Lemma 8 *The X_S^1 subshift is intrinsically Π_1^0 on every group.*

Theorem 6 *Let G be a finitely generated infinite group. The following are equivalent*

- *G is torsion,*
- *the X_S^1 subshift is not in $\mathcal{S}(G, 4)$,*
- *$\mathcal{S}(G, 4)$ is not equal to the class of all intrinsically Π_1^0 subshifts, and*
- *the hierarchy $\mathcal{S}(G, k)_{k \geq 1}$ is infinite.*

Proof This follows from Lemma 8, Theorem 2, Theorem 3 and Theorem 5. □

Finally, we note that Lemma 8 requires the intrinsic notion of computability, as shown by the following corollary of [2, Proposition 2.3] (also proved in [8]).

Proposition 1 *Let G be a recursively presented and finitely generated group, and let S be a nontrivial finite alphabet. The subshift $X_{S,G}^1$ is Π_1^0 if and only if G has a decidable word problem.*

7 Three- and four-headed automata on non-torsion groups

In this section, we give a sufficient condition for a torsion group G to satisfy $\mathcal{S}(G \times \mathbb{Z}, 3) \subsetneq \mathcal{S}(G \times \mathbb{Z}, 4)$. We note that we do not have examples of such groups. Intuitively, the property required is that the word problem of G is highly non-computable, in the specific sense that there is a recursively enumerable set of 'long' identities in the group whose validity cannot be checked using an oracle for the set of all 'short' identities. We also require an additional quantitative property related to the torsion function to give meaning to 'short' and 'long'.

The proof idea is that we consider $G \times \mathbb{Z}$-configurations over $S = \{0, 1\}$ where the $\mathbb{Z}$-cosets all contain the same configuration $x \in S^{\mathbb{Z}}$. The configuration x is p-periodic, where $p = \phi(w) \geq 1$ is a computable compression of a word $w \in \mathcal{G}(G)^*$ over the generators of G, and we take the subshift corresponding to those periods p that encode the identity element 1_G. We show that three heads can only explore a small number of identities of G (though they can explore any amount of *positions* of G), so if we assume suitable 'unpredictability properties' for G, then three heads cannot explore the word problem for long enough words to determine whether the word coded by p is the identity map.

We proceed to the details. In this section, fix G to be a finitely generated torsion group with generating set $\mathcal{G}(G)$, and define $T : \mathbb{N} \to \mathbb{N}$ by $T(n) = \max(T_G(n), n)$, where T_G is the torsion function of G. This definition ensures that our formulas are accurate even if T_G is sublinear. For each $p \in \mathbb{N}$, let $x^p \in \{0,1\}^{G \times \mathbb{Z}}$ be the configuration where $x^p_{(g,n)} = 1$ if and only if $n \equiv 0 \bmod p$. For a Turing machine M and numbers $p \in \mathbb{N}$ and $r \in \mathbb{N}$, we write $M(p, B_G(r))$ for the result of running M on the input p using an oracle for the word problem of G for words of length at most r (the result being either '$M(p, B_G(r))$ eventually halts' or '$M(p, B_G(r))$ never halts').

For a set $B \subset \mathbb{N}$, let $X_B \subset \{0,1\}^{G \times \mathbb{Z}}$ be the smallest subshift containing the configurations x^p with $p \in B$. The subshift X_B contains precisely the orbits of the configurations x^p such that $p \in B$, and if B is infinite, also the configuration where each $\mathbb{Z}$-coset contains the configuration $\cdots 0001000 \cdots$ (with the 1 in the same position in each coset) and the all-zero configuration. These limiting configurations are not relevant for the proof, but we mention them for completeness.

We now show that the heads of a three-headed automaton cannot travel arbitrarily far from each other on the periodic configurations x^p.

Lemma 9 *With the definitions above, let $\zeta : \mathbb{N} \to \mathbb{N}$ be any function with superexponential growth and let $A = (Q, f, I, F, \{0,1\})$ be a 3-headed automaton on $G \times \mathbb{Z}$ with associated subshift Y. For all large enough $p \in \mathbb{N}$ (depending on A), if $y \in Y$ contains all heads of A at the same coordinate and $n \in \mathbb{N}$ is arbitrary, then $\pi_2(f^n(x^p, y))$ contains the heads at some coordinates (g_0, n_0), (g_1, n_1) and (g_2, n_2) where*

$$\operatorname{diam}(\{g_0, g_1, g_2\}) \leq (\zeta \circ T \circ \zeta \circ T \circ \zeta)(p).$$

Proof The proof is very similar to that of Lemma 4, so we give only a sketch. Since we are dealing with a Cartesian product $G \times \mathbb{Z}$, the distance between two coordinates $(g,n), (h,m) \in G \times \mathbb{Z}$ is the sum of the coordinate-wise distances, or $d_G(g,h) + |n - m|$. We call these coordinate-wise metrics the *G-distance* and the *$\mathbb{Z}$-distance*. Let $r \in \mathbb{N}$ be the radius of f, and denote $q = |Q|$.

Consider the run of a single head of A on a configuration from the orbit of x^p, starting from some coordinate $(g,n) \in G \times \mathbb{Z}$. The head can travel a G-distance of at most $T(pqr)$ from its initial position. Namely, in at most pq steps, it must simultaneously repeat its internal state and its position modulo p on the $\mathbb{Z}$-coset. At this point, it has traveled to some coordinate $(h, n + mp) \in G \times \mathbb{Z}$, where $d_G(g,h) \leq pqr \leq T(pqr)$. The head is now necessarily in a loop, moving on the $\mathbb{Z}$-component at a linear rate. Using the torsion function T_G, we obtain as in Lemma 4 that the G-distance between the head and (g,n) never exceeds $T(pqr)$.

Next, consider the run of two heads of A, started from the same position. If the heads at some point get separated by a $\mathbb{Z}$-distance of at least $3pqr$, then the heads must be in a loop, with their $\mathbb{Z}$-distance increasing at a linear rate. The heads will never meet again, and can each travel a G-distance of at most $T(pqr)$ from that point on.

Similarly, if the two heads separate by a G-distance of more than $2T(pqr) + r$ at some point, then they will never meet again, and can each travel an additional G-distance of at most $T(pqr)$.

Finally, if the heads always stay within G-distance $2T(pqr) + r$ and $\mathbb{Z}$-distance $3pqr$ from each other, then we can consider them to be a single head with $q' = pq^2 \cdot 3pqr|B_G(2T(pqr) + r)|$ states. By the first part of the proof, after q' steps the heads either have not separated, and thus will never travel a G-distance of more than $T(pq'r)$, or alternatively separate before this, and then travel at most an additional G-distance of $2T(pqr)$ steps. All in all, two heads can travel at most a G-distance of

$$T(3p^3q^3r^2|B_G(2T(pqr)+r)|) + 2T(pqr) \leq C(T \circ \zeta \circ T \circ \zeta)(p)$$

for a constant $C > 0$ and all large enough p (depending on A), since T grows at least linearly and $|B_G(\cdot)|$ grows at most exponentially.

Now, consider the movement of 3 heads. If p is large enough, and one of the heads is not within G-distance of $3C(T \circ \zeta \circ T \circ \zeta)(p)$ from any other head, then the heads will not travel an additional G-distance of more than $C(T \circ \zeta \circ T \circ \zeta)(p)$ by the previous argument. Thus, the three heads will forever stay within G-distance $5C(T \circ \zeta \circ T \circ \zeta)(p)$ from each other. In particular they stay within G-distance $(\zeta \circ T \circ \zeta \circ T \circ \zeta)(p)$ from each other. □

We now present a formal definition for the property that we require from G.

Definition 6 Let $\phi : \mathbb{N} \to \mathbb{N}$ be a function. We say that a group G is *ϕ-unpredictable* if there exists a partial computable function $\psi : \mathbb{N} \to \mathcal{G}(G)^*$ such that for any Turing machine M, there exist infinitely many $p \in \mathbb{N}$ such that $\psi(p)$ is defined and

$$\psi(p) \sim_G 1_G \iff M(p, B_G(\phi(p))) \text{ never halts.}$$

Proposition 2 *Let G and T be as before, and suppose that there exists a superexponential function $\zeta : \mathbb{N} \to \mathbb{N}$ such that G is $(\zeta \circ T \circ \zeta \circ T \circ \zeta)$-unpredictable. Then*

$$\mathcal{S}(G \times \mathbb{Z}, 3) \subsetneq \mathcal{S}(G \times \mathbb{Z}, 4).$$

Proof Let $\phi = \zeta \circ T \circ \zeta \circ T \circ \zeta$, let $\psi : \mathbb{N} \to S^*$ be the function in the definition of ϕ-unpredictability, and let

$$B = \{p \in \mathbb{N} \mid \psi(p) \text{ is undefined or } \psi(p) \not\sim_G 1_G\}.$$

We claim that the subshift $X_B \subset \{0,1\}^{G \times \mathbb{Z}}$ is intrinsically Π^0_1, and thus belongs to $\mathcal{S}(G \times \mathbb{Z}, 4)$ by Theorem 2. Namely, B can be defined by the following forbidden patterns:

– Every two-element pattern $\{(g,n) \mapsto 0, (h,n) \mapsto 1\}$, which forbid two $\mathbb{Z}$-cosets from having distinct configurations.
– The patterns corresponding to the words 10^a10^{a+1} and $0^{a+1}10^a1$ on a $\mathbb{Z}$-coset for all $a \in \mathbb{N}$, which force the periodic structure of the configurations x^p.
– For all $p \in \mathbb{N}$ such that $\psi(n)$ is defined and $\psi(n) \sim_G 1_G$, the pattern corresponding to the word $10^{p-1}1$ on a $\mathbb{Z}$-coset, which remove the incorrect configurations x^p from X_B.

The above patterns can clearly be generated by a Turing machine using the word problem of G as an oracle.

We claim that $X_B \notin \mathcal{S}(G \times \mathbb{Z}, 3)$, so suppose for contradiction that there exists a three-headed automaton A that defines X_B. Now, consider the following Turing machine M. Given $p \in \mathbb{N}$ and an oracle for the $\phi(p)$-ball of G, the machine M starts simulating A on the finitely many configurations in the orbit of x^p, starting from initial patterns of A placed at the origin. Since the heads stay within G-distance $\phi(p)$ from each other, this simulation is possible: Since each configuration in X_B is G-invariant, we can fix the first head to stay at the origin and move the others relatively to it. More precisely, the moves of the second and third head are performed as indicated by the local rule of A, while a move of the first fixed head from $(1_G, n)$ to (h, n') is simulated by a movement of the first head from $(1_G, n)$ to $(1_G, n')$, and for $i = 2, 3$ moving the ith head from (g_i, n_i) to $(h^{-1}g_i, n_i)$. If the G-distance of the heads exceeds $\phi(p)$, or the simulation of A never rejects the configuration, then $M(p, B_G(\phi(p)))$ never halts. If A rejects the configuration from some initial state before the G-distance of the heads exceeds $\phi(p)$, then $M(p, B_G(\phi(p)))$ eventually halts.

We claim that for all large enough p, the machine $M(p, B_G(\phi(p)))$ halts if and only if A rejects the configuration x^p. Namely, Lemma 9 implies that for all large enough p, the heads of A always stay within G-distance of $\phi(p)$ from the first head, and since the first head is fixed at the origin and the other heads move relatively to it in the simulation, the machine $M(p, B_G(\phi(p)))$ will either halt or keep simulating A forever.

Now, by the assumption that A defines the subshift X_B, for all large enough $p \in \mathbb{N}$ the machine $M(p, B_G(\phi(p)))$ eventually halts if and only if $p \notin B$, which is equivalent to $\psi(p)$ being defined and $\psi(p) \sim_G 1_G$. Since G is ϕ-unpredictable, there exist arbitrarily large $p \in \mathbb{N}$ such that $\psi(p)$ is defined and

$$\psi(p) \sim_G 1_G \iff M(p, B_G(\phi(p))) \text{ never halts.}$$

This is a contradiction, so A cannot define the subshift X_B. $\square$

8 Future work and open questions

We believe a group with the property required in Proposition 2 exists, which leads to the following conjecture.

Conjecture 1 There exists a finitely generated non-torsion group G with the property $\mathcal{S}(G, 3) \subsetneq \mathcal{S}(G, 4)$. In particular, $\mathcal{S}(G, 3)$ is not always equal to the class of intrinsically Π^0_1 subshifts.

We know that if G is not a torsion group, then the hierarchy $\mathcal{S}(G, k)_{k \geq 1}$ collapses to the fourth level (if not earlier), and $\mathcal{S}(G, 4)$ is exactly the class of intrinsically Π^0_1 subshifts. On torsion groups, the hierarchy does not contain all intrinsically Π^0_1 subshifts, and we have shown in Theorem 5 that it is infinite. We have not tried to optimize the proof with respect to the number of heads needed for a strict inclusion (as is evident from the unknown constant C), but we nevertheless believe that the hierarchy is strict on all torsion groups. Namely, this holds for many different types of automata on finite words [15, 10], and our results show that in some sense, torsion groups behave similarly to them. The main difficulty in converting these results to our formalism is that the proofs usually involve languages with precise combinatorial structure, and the geodesics of torsion groups might in principle contain some additional information that allows a k-head group-walking automaton to recognize geodesic subshifts of languages that no k-head one-dimensional automaton can recognize.

Conjecture 2 If G is an infinite finitely generated torsion group, then $\mathcal{S}(G,k) \subsetneq \mathcal{S}(G,k+1)$ for all $k \geq 1$.

Some very basic questions about the abelian cases were left open in [11]. We have no progress on these questions.

Question 1 Do we have $\mathcal{S}(\mathbb{Z},2) = \mathcal{S}(\mathbb{Z},3)$ or $\mathcal{S}(\mathbb{Z}^2,2) = \mathcal{S}(\mathbb{Z}^2,3)$?

We note that in [5], a slightly different model of multi-headed group-walking automaton is studied on the group $\mathbb{Z}^2$, and it is shown that in this model, two-headed machines are strictly weaker than three-headed ones. It seems that the question is harder in our model. In [11], we only showed that $\mathcal{S}(\mathbb{Z}^d,2) \subsetneq \mathcal{S}(\mathbb{Z}^d,3)$ holds for $d \geq 3$. It is natural to ask what happens on general groups, and we make the following conjecture.

Conjecture 3 Let G be any finitely generated infinite group that is not virtually $\mathbb{Z}$ or virtually $\mathbb{Z}^2$. Then $\mathcal{S}(G,2) \subsetneq \mathcal{S}(G,3)$.

Finally, we briefly discuss how the results of the paper (to the best of our knowledge) depend on our precise model of finite-state automata. Namely, we discuss an open problem related to a detail of the definition that did not play a major role in the conference version [12] of this paper, but in the present paper becomes very important: our model does not allow the heads to share a state over large distances.

Suppose that we modify our model in a natural way to allow instant communication over distances in the form of a shared state, and call this new capability *telepathy*. Allowing telepathy, Lemma 5 becomes false: we can have a two-headed machine where the first head moves along a geodesic, reading movement instructions, and have the second head follow these instructions, moving arbitrarily far from the geodesic. Thus our proof of Lemma 7 fails, and perhaps even the lemma itself is not true, as the second head could theoretically explore identities of the group that are not visible to an automaton that is not allowed to leave the geodesic. In particular, our proof of Theorem 5 fails. We do not know whether the theorem itself is true.

Question 2 In a model where telepathy is allowed, is the hierarchy $\mathcal{S}(G,k)_{k\geq 1}$ still infinite for all finitely generated infinite torsion groups G?

References

1. S.I. Adian. The Burnside problem on periodic groups and related questions. *Proceedings of the Steklov Institute of Mathematics*, 272(2):2–12, 2011.
2. N. Aubrun, S. Barbieri, and M. Sablik. A notion of effectiveness for subshifts on finitely generated groups. *ArXiv e-prints*, December 2014.
3. Manuel Blum and Carl Hewitt. Automata on a 2-dimensional tape. In *Switching and Automata Theory, 1967. SWAT 1967. IEEE Conference Record of the Eighth Annual Symposium on*, pages 155 –160, oct. 1967.
4. T. Ceccherini-Silberstein and M. Coornaert. *Cellular Automata and Groups*. Springer Monographs in Mathematics. Springer-Verlag, 2010.
5. Marianne Delorme and Jacques Mazoyer. Pebble automata. figures families recognition and universality. *Fundam. Inf.*, 52(1-3):81–132, January 2002.

6. R. I. Grigorčuk. On Burnside's problem on periodic groups. *Funktsional. Anal. i Prilozhen.*, 14(1):53–54, 1980.
7. Narain Gupta and Said Sidki. On the burnside problem for periodic groups. *Mathematische Zeitschrift*, 182(3):385–388, 1983.
8. E. Jeandel. Some Notes about Subshifts on Groups. *ArXiv e-prints*, January 2015.
9. Douglas Lind and Brian Marcus. *An introduction to symbolic dynamics and coding.* Cambridge University Press, Cambridge, 1995.
10. Burkhard Monien. Two-way multihead automata over a one-letter alphabet. *RAIRO Inform. Théor.*, 14(1):67–82, 1980.
11. Ville Salo and Ilkka Törmä. Plane-walking automata. *CoRR*, abs/1408.6701, 2014.
12. Ville Salo and Ilkka Törmä. Group-walking automata. In Jarkko Kari, editor, *Cellular Automata and Discrete Complex Systems*, volume 9099 of *Lecture Notes in Computer Science*, pages 224–237. Springer Berlin Heidelberg, 2015.
13. Rich Schroeppel. A two counter machine cannot calculate 2^N. 1972.
14. Matthew C.H. Tointon and Ariel Yadin. Horofunctions on graphs of linear growth. *Comptes Rendus Mathematique*, 354(12):1151 – 1154, 2016.
15. Andrew C. Yao and Ronald L. Rivest. $k+1$ heads are better than k. *J. Assoc. Comput. Mach.*, 25(2):337–340, 1978.