THE GROUP OF REVERSIBLE TURING MACHINES: SUBGROUPS, GENERATORS AND COMPUTABILITY

SEBASTIÁN BARBIERI, JARKKO KARI AND VILLE SALO

Abstract. We study an abstract group of reversible Turing machines. In our model, each machine is interpreted as a homeomorphism over a space which represents a tape filled with symbols and a head carrying a state. These homeomorphisms can only modify the tape at a bounded distance around the head, change the state and move the head in a bounded way. We study three natural subgroups arising in this model: the group of finite-state automata, which generalizes the topological full groups studied in topological dynamics and the theory of orbit-equivalence; the group of oblivious Turing machines whose movement is independent of tape contents, which generalizes lamplighter groups and has connections to the study of universal reversible logical gates; and the group of elementary Turing machines, which are the machines which are obtained by composing finite-state automata and oblivious Turing machines. We show that both the group of oblivious Turing machines and that of elementary Turing machines are finitely generated, while the group of finite-state automata and the group of reversible Turing machines are not. We show that the group of elementary Turing machines has undecidable torsion problem. From this, we also obtain that the group of cellular automata (more generally, the automorphism group of any uncountable one-dimensional sofic subshift) contains a finitely-generated subgroup with undecidable torsion problem. We also show that the torsion problem is undecidable for the topological full group of a full $\mathbb{Z}^d$-shift on a non-trivial alphabet if and only if $d \geqslant 2$.

Keywords: Reversible Turing machine, cellular automata, torsion problem, topological full group.

MSC2020: *Primary:* 68Q04, 68Q80, 37B15. *Secondary:* 37B10, 37B51. 20F10,

1. Introduction

This article is the extended version of the conference paper [4]. The main new results that did not appear in the conference paper are that elementary Turing machines are finitely generated, and that automorphism groups of uncountable sofic $\mathbb{Z}$-shifts have finitely-generated subgroups with undecidable torsion problem.

1.1. Turing machines and their generalization.

Turing machines have been studied since the 30s as the standard formalization of the abstract concept of computation. However, more recently, Turing machines have also been studied in the context of dynamical systems. In [27], two dynamical systems were associated to a Turing machine, one with a "moving tape" and one with a "moving head". After that, there has been a lot of study of dynamics of Turing machines, see for example [12, 7, 25, 16, 15, 21, 17]. Another connection between Turing machines and dynamics is that they can be used to describe "effectively closed" zero-dimensional dynamical systems. A particularly interesting case is that of subshifts whose forbidden patterns are enumerated by a Turing machine. These subshifts are called effectively closed, or Π_1^0 subshifts, and especially in multiple dimensions, they are central to the topic due to the strong links known between SFTs, sofic shifts and Π_1^0-subshifts, see for example [20, 13, 3]. An intrinsic notion of Turing machine computation for these subshifts on general groups was proposed in [2], and a similar study was performed with finite state machines in [38, 37].

In all these papers, the definition of a Turing machine is (up to mostly notational differences and switching between the moving tape and moving head model) the following: A Turing machine is a

function $T\colon \Sigma^{\mathbb{Z}} \times Q \to \Sigma^{\mathbb{Z}} \times Q$ defined by a local rule $f_T\colon \Sigma \times Q \to \Sigma \times Q \times \{-1, 0, 1\}$ by the formula

$$T(x, q) = (\sigma^{-d}(\tilde{x}), q') \text{ if } f_T(x_0, q) = (a, q', d),$$

where $\sigma\colon \Sigma^{\mathbb{Z}} \to \Sigma^{\mathbb{Z}}$ is the shift action given by $\sigma^d(x)_z = x_{z-d}$, $\tilde{x}_0 = a$ and $\tilde{x}|_{\mathbb{Z}\setminus\{0\}} = x|_{\mathbb{Z}\setminus\{0\}}$. In this paper, such Turing machines are called **classical Turing machines**. This definition (as far as we know) certainly suffices to capture all computational and dynamical properties of interest, but it also has some undesirable properties: The composition of two classical Turing machines – and even the square of a classical Turing machine – is typically not a classical Turing machine, and the reverse of a reversible classical Turing machine is not always a classical Turing machine.

In this paper, we give a more general definition of a Turing machine, by allowing it to move the head and modify cells at an arbitrary (but bounded) distance on each timestep. With the new definition, we get rid of both issues: With our definition,

- Turing machines are closed under composition, forming a monoid, and
- reversible Turing machines are closed under inversion, forming a group.

We also characterize reversibility of classical Turing machines in combinatorial terms, and show what their inverses look like. Our definition of a Turing machine originated in the yet unpublished work of M. Schraudner and the last author, where the group of such machines was studied on general subshifts (with somewhat different objectives). The same definition was given in 1991 by Moore [33] under the name "generalized shifts".

These benefits of the definition should be compared to the benefits of allowing arbitrary radii in the definition of a cellular automaton: If we define cellular automata as having a fixed radius of, say, 3, then the inverse map of a reversible cellular automaton is not always a cellular automaton, as the inverse of a cellular automaton may have a much larger radius, see [11]. Similarly, with a fixed radius, the composition of two cellular automata is not necessarily a cellular automaton.

We give our Turing machine definitions in two ways, with a moving tape and with a moving head, as done in [27]. The moving tape point of view is often the more useful one when studying one-step behavior and invariant measures, whereas we find the moving head point of view easier for constructing examples, and when we need to track the movement of multiple heads. The moving head Turing machines are in fact a subset of cellular automata on a particular kind of subshift. The moving tape machine on the other hand is a generalization of the topological full group of a subshift, which is an important concept in particular in topological dynamics and the theory of orbit equivalence. For the study of topological full groups of minimal subshifts and their interesting group-theoretic properties, see for example [18, 19, 22]. The (one-sided) SFT case is studied in [32]. We shall show that our two Turing machine models yield isomorphic monoids, and isomorphic groups in the case of reversible Turing machines.

1.2. **Our results and comparisons with other groups.** In Section 2, we define our models and prove basic results about them. In Section 2.3, we define a natural uniform measure on these spaces and use it to show that injectivity and surjectivity are both equal to reversibility in our model.

Our results have interesting counterparts in the theory of cellular automata: One of the main theorems in the theory of cellular automata is that on a large class of groups (the **surjunctive groups**, see for instance [9, Chapter 3]) injectivity implies surjectivity, and (global) bijectivity is equivalent to having a cellular automaton inverse map. Furthermore, one can attach to a reversible one- or two-dimensional cellular automaton its "average drift", that is, the speed at which information moves when the map is applied, and this is a homomorphism from the group of cellular automata to a subgroup

of $\mathbb{Q}^d$ under multiplication (where d is the corresponding dimension), see [23]. In Section 3 we use the uniform measure to define an analog, the "average movement" homomorphism for Turing machines.

In Section 3, we define some interesting subgroups of the group of reversible Turing machines. First, we define the local permutations – Turing machines that never move the head at all –, and their generalization to oblivious Turing machines where movement is allowed, but is independent of the tape contents. The group of oblivious Turing machines can be seen as a kind of generalization of lamplighter groups. It turns out that the group of oblivious Turing machines is finitely generated. Our proof relies strongly on the existence of universal reversible logical gates, see [1].

We also define the group of (reversible) finite-state machines – Turing machines that never modify the tape. This group is not finitely generated, but we give a natural infinite generating set for it. Finite-state machines with a single state exactly correspond to the topological full groups of full shifts, and in this sense our definition of a reversible finite-state machine can be seen as a generalization of the topological full group on a full $\mathbb{Z}^d$-shift.

Our original motivation for defining these subgroups – finite-state machines and local permutations – was to study the question of whether they generate all reversible Turing machines. Namely, a reversible Turing machine changes the tape contents at the position of the head and then moves, in a globally reversible way. Thus, it is a natural question whether every reversible Turing machine can actually be split into reversible tape changes (actions by local permutations) and reversible moves (finite-state automata).

We call the join of finite-state machines and local permutations elementary Turing machines, and show that not all Turing machines are elementary, as Turing machines can have arbitrarily small average movement, but elementary ones have only a discrete sublattice of possible average movements. However, we show that every reversible classical Turing machine (and thus all its iterates) is elementary.

A reason why elementary Turing machines are so interesting is that, while we show that the group of reversible Turing machines is not finitely generated (Theorem 3.11), we have that the group of elementary Turing machines on a one-dimensional tape is finitely generated, for any fixed alphabet size and number of states.

Theorem 1.1. *The group of elementary Turing machines on n symbols, k states and dimension 1 is finitely generated for every $n, k \geqslant 1$.*

In Section 4, we show that the group of Turing machines is recursively presented and has a decidable word problem, but that its torsion problem (the problem of deciding if a given element has finite order) is undecidable in all dimensions. In fact, we show that the finitely-generated group of elementary Turing machines has an undecidable torsion problem in the purely group-theoretic sense. The proof is based on simulating classical Turing machines with elementary ones on a fixed alphabet and state set, with a simulation that preserves finite orbits. The result follows because the periodicity of reversible Turing machines is undecidable [25].

Theorem 1.2. *The torsion problem of the group of elementary Turing machines on n symbols, k states and dimension d is undecidable for every $n \geqslant 2$, $k \geqslant 1$ and $d \geqslant 1$.*

As an application of the former result, we obtain that there is a finitely-generated group of reversible cellular automata on a full shift (thus on any uncountable sofic shift), which has undecidable torsion problem. This follows from an "almost-embedding" of the group of Turing machines into the group of cellular automata – there is no actual embedding for group-theoretic reasons, but we show that the almost-embedding we construct preserves the decidability of the torsion problem.

Corollary 1.3. *Let X be an uncountable sofic $\mathbb{Z}$-subshift. Then there is a finitely generated subgroup of* $\mathrm{Aut}(X)$ *which has undecidable torsion problem.*

For finite-state machines, we show that the torsion problem is decidable in dimension one, but is undecidable in higher dimensions; again we construct a finitely-generated subgroup where the problem is undecidable. The proof in the one-dimensional case is based on a simple pigeonhole argument, while in the higher-dimensional case undecidability is a corollary of the undecidability of the snake tiling problem [24].

Theorem 1.4. *Consider the group G of finite-state machines on $n \geqslant 2$ symbols, $k \geqslant 1$ states and dimension $d \geqslant 1$,*

(1) The torsion problem of G is decidable if $d = 1$.

(2) G contains a finitely generated subgroup with undecidable torsion problem when $d \geqslant 2$.

Furthermore, we show that the decidability result holds for finite-state machines running on an arbitrary sofic subshift and that in fact their finiteness problem is decidable meaning we can decide whether a given finitely-generated subgroup is finite or not (which implies the decidability of their torsion problem). As a special case of our results, we obtain the following statement about topological full groups.

Corollary 1.5. *Let $d \geqslant 2$. The topological full group of a full $\mathbb{Z}^d$-shift on at least two symbols contains a finitely generated subgroup with undecidable torsion problem.*

We note that our group is very similar to the Brin-Thompson group 2V [6, 5], the main difference being that elements of 2V can erase and add symbols to the tape; indeed, the group of reversible Turing machines on an alphabet with 2 symbols and 1 state can be seen naturally as a subgroup of 2V, consisting of elements that do not use this addional functionality. Our group is not isomorphic to any of the Brin-Thompson groups nV, as they are finitely generated. Thompson's V has a decidable torsion problem, but that of 2V is not, also due to the undecidability of periodicity of reversible Turing machines.

1.3. **Preliminaries.** In this section we present general definitions and fix the notation which is used throughout the article. The review of these concepts will be brief and focused on the dynamical aspects. For a more complete introduction the reader may refer to [30]. Throughout this article we shall define several monoids and groups that come from Turing machines. In order to make it easier for the reader to recall these notations, we have added a dictionary in Appendix A, where these definitions can be found along with short descriptions and references to where they are defined.

We restrict our setting to finitely generated and torsion-free abelian groups $\mathbb{Z}^d$, although, just like with cellular automata [9], the idea of a Turing machine directly generalizes to general groups. All of the results of this and the following section directly generalize for arbitrary countable groups G in place of $\mathbb{Z}^d$.

Let $\mathcal{A}$ be a finite alphabet. The set $\mathcal{A}^{\mathbb{Z}^d} = \{x \colon \mathbb{Z}^d \to \mathcal{A}\}$ equipped with the shift action $\sigma \colon \mathbb{Z}^d \times \mathcal{A}^{\mathbb{Z}^d} \to \mathcal{A}^{\mathbb{Z}^d}$ defined by $(\sigma^{\vec{v}}(x))_{\vec{u}} = x_{\vec{u}-\vec{v}}$ is a **full shift**. Thus $\sigma^{\vec{v}}$ shifts cell contents in direction $\vec{v}$, or equivalently moves the origin of x to $-\vec{v}$. The elements $a \in \mathcal{A}$ and $x \in \mathcal{A}^{\mathbb{Z}^d}$ are called **symbols** and **configurations** respectively. Configuration $x \in \mathcal{A}^{\mathbb{Z}^d}$ is **periodic** if $\sigma^{\vec{v}}(x) = x$ for some non-zero $\vec{v} \in \mathbb{Z}^d$, and it is **eventually periodic** if there exists a periodic configuration $y \in \mathcal{A}^{\mathbb{Z}^d}$ that differs from x only in finitely many positions. The set of configurations $\mathcal{A}^{\mathbb{Z}^d}$ endowed with the prodiscrete topology (that is, the product topology obtained from taking the discrete topology on $\mathcal{A}$) is a compact metrizable space, a generating metric being given by $d(x, y) = 2^{-\inf\{|\vec{v}| \,:\, \vec{v} \in \mathbb{Z}^d,\ x_{\vec{v}} \neq y_{\vec{v}}\}}$, where $|\vec{v}|$ is the taxicab norm $|\vec{v}| = \sum_{i=1}^{d} |\vec{v}_i|$.

This topology has the sets $[a]_{\vec{v}} = \{x \in \mathcal{A}^{\mathbb{Z}^d} : x_{\vec{v}} = a \in \mathcal{A}\}$ as a subbasis. A **support** is a finite subset $F \subset \mathbb{Z}^d$. Given a support F, a **pattern with support** F is an element p of $\mathcal{A}^F$. The **cylinder** generated by p in position $\vec{v}$ is $[p]_{\vec{v}} = \bigcap_{\vec{u}\in F} [p_{\vec{u}}]_{\vec{v}+\vec{u}}$. For simplicity, we write $[p] = [p]_{\vec{0}}$.

Definition 1.6. *A subset X of $\mathcal{A}^{\mathbb{Z}^d}$ is a **subshift** if it is topologically closed and σ-invariant, that is, for every $\vec{v} \in \mathbb{Z}^d$ we have $\sigma^{\vec{v}}(X) \subset X$.*

Equivalently, X is a subshift if and only if there exists a set of patterns $\mathcal{F}$ that defines it:

$$X = \bigcap_{p\in\mathcal{F},\vec{v}\in\mathbb{Z}^d} \mathcal{A}^{\mathbb{Z}^d} \setminus [p]_{\vec{v}}.$$

Any such $\mathcal{F}$ which defines X is called a set of **forbidden patterns** for X.

For a subshift $X \subset \mathcal{A}^{\mathbb{Z}^d}$ and a finite support $F \subset \mathbb{Z}^d$ we define the **language** $L_F(X)$ **of support** F of X as the set of patterns $p \in \mathcal{A}^F$ such that $[p] \cap X \neq \varnothing$. The language of X is the union $L(X)$ of $L_F(X)$ over all finite $F \subset \mathbb{Z}^d$. We denote $p \sqsubset X$ iff $p \in L(X)$. For an individual configuration $x \in \mathcal{A}^{\mathbb{Z}^d}$ we denote $p \sqsubset x$ iff $x \in [p]_{\vec{v}}$ for some $\vec{v} \in \mathbb{Z}^d$, and we say that pattern p **occurs** in x.

Let X, Y be subshifts over alphabets $\mathcal{A}$ and $\mathcal{B}$ respectively. A continuous $\mathbb{Z}^d$-equivariant (i.e. shift-commuting) map $\phi\colon X \to Y$ between subshifts is called a morphism. A well-known theorem of Curtis, Hedlund and Lyndon which can be found in full generality in [9] asserts that morphisms are equivalent to maps defined by local rules as follows: There exists a finite $F \subset \mathbb{Z}^d$ and $\Phi\colon \mathcal{A}^F \to \mathcal{B}$ such that for every $x \in X$, $\phi(x)_{\vec{v}} = \Phi(\sigma^{-\vec{v}}(x)|_F)$. If ϕ is an endomorphism (that is, $X = Y$) then we refer to it as a cellular automaton. A cellular automaton is said to be reversible if there exists a cellular automaton ϕ^{-1} such that $\phi \circ \phi^{-1} = \phi^{-1} \circ \phi = \mathrm{id}$. It is well known that reversibility is equivalent to bijectivity, see Section 1.10 of [9].

Throughout this article we use the following notation inspired by Turing machines. We denote by $\Sigma = \{0, \dots, n-1\}$ the set of tape symbols and $Q = \{1, \dots, k\}$ the set of states. We also use the symbols $n = |\Sigma|$ for the size of the alphabet and $k = |Q|$ for the number of states. Given a function of the form $f\colon \Omega \to A_1 \times \ldots \times A_m$ we denote by $f_i\colon \Omega \to A_i$ the projection of f to the i-th coordinate.

2. Two models for Turing machine groups

In this section we define our generalized Turing machine model, and the group of Turing machines. In fact, we give two definitions for this group, one with a moving head and one with a moving tape as in [27]. We show that – except in the case of a trivial alphabet – these groups are isomorphic.[1] Furthermore, both can be defined both by local rules and "dynamically", that is, in terms of continuity and the shift action. In the moving tape model we characterize reversibility as preservation of a suitably defined measure.

2.1. The moving head model.

In the moving head model, we will represent our space as $\Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$. That is, the product of the set of configurations $\Sigma^{\mathbb{Z}^d}$, a set of states Q and the possible positions of a head $\mathbb{Z}^d$. The objects of this space are therefore 3-tuples $(x, q, \vec{v})$. In order to write this in a shorter manner, we use the notation $x_q^{\vec{v}}$ instead of $(x, q, \vec{v})$.

Given a function

$$f\colon \Sigma^{F_{\mathrm{in}}} \times Q \to \Sigma^{F_{\mathrm{out}}} \times Q \times \mathbb{Z}^d,$$

where $F_{\mathrm{in}}, F_{\mathrm{out}}$ are finite subsets of $\mathbb{Z}^d$, we can define a map $T_f\colon \Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d \to \Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$ as follows: given $x_q^{\vec{v}} \in \Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$ let $p = \sigma^{-\vec{v}}(x)|_{F_{\mathrm{in}}}$ and $f(p, q) = (\tilde{p}, \tilde{q}, \vec{w})$. Then we define $T_f(x_q^{\vec{v}}) = \tilde{x}_{\tilde{q}}^{\vec{v}+\vec{w}}$ where for $\vec{u} \in \mathbb{Z}^d$:

[1] Note that the **dynamics** obtained from these two definitions are in fact quite different, as shown in [27, 28].

$$\tilde{x}_{\vec{u}} = \begin{cases} x_{\vec{u}} & \text{if } \vec{u} - \vec{v} \notin F_{\text{out}}, \\ \tilde{p}_{\vec{u}-\vec{v}} & \text{if } \vec{u} - \vec{v} \in F_{\text{out}}. \end{cases}$$

Definition 2.1. *A function T for which there is an $f\colon \Sigma^{F_{in}} \times Q \to \Sigma^{F_{out}} \times Q \times \mathbb{Z}^d$ such that $T = T_f$ is called a* ***moving head*** *$(\mathbb{Z}^d, n, k)$-**Turing machine**, and f is its* ***local rule****. If there exists a $(\mathbb{Z}^d, n, k)$-Turing machine T^{-1} such that $T \circ T^{-1} = T^{-1} \circ T = \mathrm{id}$, we say T is* ***reversible****.*

This definition corresponds to classical Turing machines with the moving head model when $d = 1$, $F_{\text{in}} = F_{\text{out}} = \{\vec{0}\}$ and the third component of $f(x, q)$ is in $\{-1, 0, 1\}$ for all x, q. An illustration of how a moving head Turing machine acts can be seen in Figure 1. Note that $\sigma^{-\vec{v}}(x)|_F$ is the F-shaped pattern "at $\vec{v}$". We do not write $x|_{\vec{v}+F}$ because we want the pattern we read from x to have F as its domain.

Note that one of these machines could be defined by several different local functions f, and that the projection to the third coordinate f_3 of f has finite range. Also, given $f\colon \Sigma^{F_{\text{in}}} \times Q \to \Sigma^{F_{\text{out}}} \times Q \times \mathbb{Z}^d$ we can define $F = F_{\text{in}} \cup F_{\text{out}} \cup f_3(\Sigma^{F_{\text{in}}} \times Q)$ and $f'\colon \Sigma^F \times Q \to \Sigma^F \times Q \times F$ such that $T_f = T_{f'}$. This motivates the following definition: The **minimal neighborhood** of T is the minimum set for inclusion F for which there is $f\colon \Sigma^F \times Q \to \Sigma^F \times Q \times F$ such that $T = T_f$. This minimum always exists as the set of finite subsets of $\mathbb{Z}^d$ which work for this definition is closed under intersections.

As $\mathbb{Z}^d$ is finitely generated, we can also use a numerical definition of radius in place of the neighborhood: Let $B(\vec{v}, r)$ be the set of $\vec{u} \in \mathbb{Z}^d$ such that $|\vec{u} - \vec{v}| \leqslant r$. By possibly changing the local rule f, we can always choose $F_{\text{in}} = B(\vec{0}, r_i)$ and $F_{\text{out}} = B(\vec{0}, r_o)$ for some $r_i, r_o \in \mathbb{N}$, without changing the Turing machine T_f it defines. The minimal such r_i is called the **in-radius** of T, and the minimal r_o is called the **out-radius** of T. We say the in-radius of a Turing machine is -1 if there is no dependence on input, that is, the neighborhood $B(0^d, r_i)$ can be replaced by the empty set. The maximum value of $|\vec{v}|$ for all $\vec{v} \in f_3(\Sigma^F \times Q)$ is called the **move-radius** of T. Finally, the maximum of all these three radii is called the **radius** of T. In this terminology, classical Turing machines are those with in- and out-radius 0, and move-radius 1.

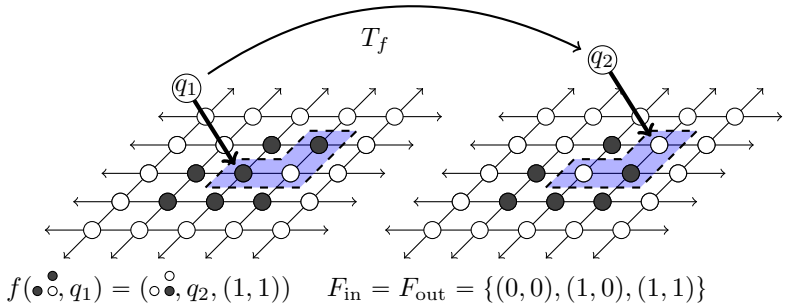


FIGURE 1. The action of a moving head machine T_f.

Definition 2.2. *We denote by $\mathrm{TM}(\mathbb{Z}^d, n, k)$ the set of $(\mathbb{Z}^d, n, k)$-Turing machines and $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ the set of reversible $(\mathbb{Z}^d, n, k)$-Turing machines.*

In some parts of this article we just consider $d = 1$. In this case we simplify the notation and just write $\mathrm{RTM}(n, k) = \mathrm{RTM}(\mathbb{Z}, n, k)$ and $\mathrm{TM}(n, k) = \mathrm{TM}(\mathbb{Z}, n, k)$. Of course, we want $\mathrm{TM}(\mathbb{Z}^d, n, k)$ to be a monoid and $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ a group under function composition. This is indeed the case, and one can prove this directly by constructing local rules for the inverse of a reversible Turing machine

and composition of two Turing machines. However, it is easier to extract this from the following characterization of Turing machines as a particular kind of cellular automaton.

Let X_k be the subshift with alphabet $Q \cup \{0\}$ such that in each configuration the number of non-zero symbols is at most one.

$$X_k = \{x \in \{0, 1, \dots, k\}^{\mathbb{Z}^d} : 0 \notin \{x_{\vec{u}}, x_{\vec{v}}\} \implies \vec{u} = \vec{v}\}.$$

In the case where $k = 1$ this subshift is often called the **sunny-side-up subshift**. The name comes from [34] as a reference to the shape of an egg prepared in that manner. We note that X_1 is also called the "at-most-one-one-subshift" in [10]. Notice that $X_0 = \{0^{\mathbb{Z}^d}\}$ consists of a single configuration and for non-negative integers $i < j$ we have $X_i \subsetneq X_j$. Let also $X_{n,k} = \Sigma^{\mathbb{Z}^d} \times X_k$, where recall that we always set $\Sigma = \{0, \dots, n-1\}$. For the case $d = 1$, configurations in $X_{n,k}$ represent a bi-infinite tape filled with symbols in Σ possibly containing a head that has a state in Q. Note that there might be no head in a configuration.

More precisely, we interpret $X_{n,k}$ as the compactification of $\Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$ by identifying $x_q^{\vec{v}} = (x, q, \vec{v}) \in \Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$ with the point $(x, y) \in X_{n,k}$, where $y_{\vec{v}} = q$ and $y_{\vec{u}} = 0$ for $\vec{u} \neq \vec{v}$. In other words, we add the possibility of having no head. We can now interpret Turing machines as functions on $X_{n,k}$ in the following way: For $(x, y) \in X_{n,k}$, if there is no $\vec{v} \in \mathbb{Z}^d$ such that $y_{\vec{v}} \neq 0$ then $T(x, y) = (x, y)$. Otherwise apply T through the natural bijection.

For a subshift X, we denote by $\mathrm{End}(X)$ the monoid of endomorphisms of X and $\mathrm{Aut}(X)$ the group of automorphisms of X. Next we will argue that Turing machines with n symbols and k states can be seen as endomorphisms of $X_{n,k}$. Continuity follows essentially from the fact that Turing machines only modify configurations near the head based on the symbols around it. Furthermore, since the shift moves both the configuration and the head by the same amount, it follows that Turing machines commute with the shift. For the convenience of the reader, we provide a formal proof in the following lemma.

Lemma 2.3. *Every $T \in \mathrm{TM}(\mathbb{Z}^d, n, k)$ induces a continuous and shift-invariant map on $X_{n,k}$.*

Proof. Let $T \in \mathrm{TM}(\mathbb{Z}^d, n, k)$ and let $f \colon \Sigma^F \times Q \to \Sigma^F \times Q \times F$ be such that $T = T_f$. Clearly T commutes with the shift on $X_{n,0}$ since it acts as the identity map. Let then $x_q^{\vec{v}} \in X_{n,k} \backslash X_{n,0}$.

Let $p = \sigma^{-\vec{v}}(x)|_F$ and $f(p, q) = (\tilde{p}, \tilde{q}, \tilde{\vec{u}})$. Then by the definition of T_f we have $T_f(x_q^{\vec{v}}) = \tilde{x}_{\tilde{q}}^{\vec{v}+\tilde{\vec{u}}}$ where

$$\tilde{x}_{\vec{u}} = \begin{cases} x_{\vec{u}} & \text{if } \vec{u} - \vec{v} \notin F \\ \tilde{p}_{\vec{u}-\vec{v}} & \text{if } \vec{u} - \vec{v} \in F \end{cases}$$

Now, let us consider the application of T_f to $\sigma^{\vec{w}}(x_q^{\vec{v}})$. By the definition of σ we have $\sigma^{\vec{w}}(x_q^{\vec{v}}) = \sigma^{\vec{w}}(x)_q^{\vec{v}+\vec{w}}$. Let $y = \sigma^{\vec{w}}(x)$ and $\vec{v}' = \vec{v} + \vec{w}$ so that $T_f(\sigma^{\vec{w}}(x_q^{\vec{v}})) = T_f(y_q^{\vec{v}'})$. Let $p' = \sigma^{-\vec{v}'}(y)|_F$ and note that

$$p'_{\vec{u}} = \sigma^{-\vec{v}'}(y)_{\vec{u}} = y_{\vec{u}+\vec{v}'} = \sigma^{\vec{w}}(x)_{\vec{u}+\vec{v}'} = x_{\vec{u}+\vec{v}'-\vec{w}} = x_{\vec{u}+\vec{v}} = \sigma^{-\vec{v}}(x)_{\vec{u}}$$

for all $\vec{u} \in F$ so $p' = p$. We have thus $f(p', q) = f(p, q) = (\tilde{p}, \tilde{q}, \tilde{\vec{u}})$. Then by definition we have $T_f(y_q^{\vec{v}'}) = \tilde{y}_{\tilde{q}}^{\vec{v}'+\tilde{\vec{u}}}$ where

$$\tilde{y}_{\vec{u}} = \begin{cases} y_{\vec{u}} & \text{if } \vec{u} - \vec{v}' \notin F \\ \tilde{p}_{\vec{u}-\vec{v}'} & \text{if } \vec{u} - \vec{v}' \in F \end{cases}$$

We have to show that

$$\tilde{y}_{\tilde{q}}^{\vec{v}'+\tilde{\vec{u}}} = \sigma^{\vec{w}}(\tilde{x}_{\tilde{q}}^{\vec{v}+\tilde{\vec{u}}}) = \sigma^{\vec{w}}(\tilde{x})_{\tilde{q}}^{\vec{v}+\tilde{\vec{u}}+\vec{w}}.$$

By definition, $\vec{v}' + \tilde{\vec{u}} = \vec{v} + \vec{w} + \tilde{\vec{u}}$. We now show $\tilde{y} = \sigma^{\vec{w}}(\tilde{x})$, that is, $\tilde{y}_{\vec{u}} = \tilde{x}_{\vec{u}-\vec{w}}$ for all $\vec{u} \in \mathbb{Z}^d$.

Let $\vec{u} \in \mathbb{Z}^d$. First suppose $\vec{u} - \vec{v}' \notin F$. Then $\tilde{y}_{\vec{u}} = y_{\vec{u}} = x_{\vec{u}-\vec{w}}$. On the other hand $\sigma^{\vec{w}}(\tilde{x})_{\vec{u}} = \tilde{x}_{\vec{u}-\vec{w}} = x_{\vec{u}-\vec{w}}$ where the last equality follows from the definition of $\tilde{x}$ because $\vec{u} - \vec{w} - \vec{v} = \vec{u} - \vec{v}' \notin F$.

Suppose then that $\vec{u} - \vec{v}' \in F$. Then $\tilde{y}_{\vec{u}} = \tilde{p}_{\vec{u}-\vec{v}'}$. As above, by the definition of $\tilde{x}$ we have $\vec{u} - \vec{w} - \vec{v} = \vec{u} - \vec{v}' \in F$ so $\tilde{x}_{\vec{u}-\vec{w}} = \tilde{p}_{\vec{u}-\vec{w}-\vec{v}} = \tilde{p}_{\vec{u}-\vec{v}'}$. This concludes the proof that T_f commutes with the shift.

Take $m \geqslant 1$ and choose some r such that $F \subset [-r, r]^d$. Given any $(x, c) \in X_{n,k}$, we have by definition of T_f that the values $T_f(x, c)|_{[-m,m]^d}$ depend exclusively on the values of (x, c) in $[-m, m]^d + F \subset [-m - r, m + r]^d$, from where it follows that $d(T_f(x, c), T_f(x', c')) \leqslant 2^{-m}$ whenever $d((x, c), (x', c')) \leqslant 2^{-(m+r)}$, thus T_f is continuous. □

From the previous argument it follows that Turing machines are cellular automata on $X_{n,k}$. We can make this identification precise if we add the property that "the configurations are only modified near a head". This is the content of the following proposition.

Proposition 2.4. *Let n, k be positive integers and $Y = X_{n,0}$. Then:*

$$\mathrm{TM}(\mathbb{Z}^d, n, k) = \{\phi \in \mathrm{End}(X_{n,k}) : \phi|_Y = \mathrm{id}, \phi^{-1}(Y) = Y\}$$
$$\mathrm{RTM}(\mathbb{Z}^d, n, k) = \{\phi \in \mathrm{Aut}(X_{n,k}) : \phi|_Y = \mathrm{id}\}.$$

Proof. Consider a Turing machine $T \in \mathrm{TM}(\mathbb{Z}^d, n, k)$ seen as a function on $X_{n,k}$. By Lemma 2.3 we have that $T \in \mathrm{End}(X_{n,k})$. Also, T acts trivially on $X_{n,0}$ so $T|_Y = \mathrm{id}$ and if a configuration has a head, it can only be shifted but not disappear, thus $T^{-1}(Y) = Y$. Moreover, if $T \in \mathrm{RTM}(\mathbb{Z}^d, n, k)$, then T has a Turing machine inverse, thus a cellular automaton inverse, and it follows that $T \in \mathrm{Aut}(X_{n,k})$.

Conversely, let $\phi \in \mathrm{End}(X_{n,k})$, so that $\phi(x, y)_{\vec{v}} = \Phi(\sigma^{-\vec{v}}(x, y)|_F)$ for some local rule $\Phi\colon (\Sigma \times \{0, \dots, k\})^F \to \Sigma \times \{0, \dots, k\}$ and F a finite subset of $\mathbb{Z}^d$, where we may suppose $\vec{0} \in F$.

As $\phi|_Y = \mathrm{id}$, we can deduce that $\Phi(u, v) = (u, v)_{\vec{0}}$ if $v = 0^F$. Therefore if $(x, y) \in X_{n,k}$, $y_{\vec{v}} \neq 0$ and we define $W_{\vec{v}} = \{\vec{u} : \vec{v} \in \vec{u} + F\} = \vec{v} - F$ we get that $\phi(x, y)|_{\mathbb{Z}^d \setminus W_{\vec{v}}} = (x, y)|_{\mathbb{Z}^d \setminus W_{\vec{v}}}$. We extend Φ to $\widetilde{\Phi}\colon (\Sigma \times \{0, \dots, k\})^{W_{\vec{0}}+F} \to (\Sigma \times \{0, \dots, k\})^{W_{\vec{0}}}$ by pointwise application of Φ. Note that $W_{\vec{0}} = -F$.

We can then define $f_\phi\colon \Sigma^{F-F} \times Q \to \Sigma^{-F} \times Q \times \mathbb{Z}^d$ by using $\widetilde{\Phi}$ as follows: We set $f_\phi(p, q) = (p', q', \vec{u})$ if, after defining $r \in \{0, \dots, k\}^{F-F}$ such that $r_{\vec{0}} = q$ and 0 elsewhere, we have $\widetilde{\Phi}(p, r) = (p', r')$ and $r' \in \{0, \dots, k\}^{-F}$ contains the symbol $q' \neq 0$ in position $\vec{u}$ (there is always a unique such position $\vec{u}$ as $\phi^{-1}(Y) = Y$). It can be verified that the Turing machine T_{f_ϕ} is precisely ϕ, therefore $\phi \in \mathrm{TM}(G, n, k)$.

If $\phi \in \mathrm{Aut}(X_{n,k})$ then $\phi^{-1}(Y) \supset Y$ is implied by $\phi|_Y = \mathrm{id}$, and since the inverse automorphism ϕ^{-1} satisfies $\phi^{-1}|_Y = \mathrm{id}$ as well, we also have $\phi^{-1}(Y) \subset Y$. Thus, ϕ is a Turing machine. Similarly, the inverse map ϕ^{-1} is a Turing machine. Thus, in this case $\phi \in \mathrm{RTM}(\mathbb{Z}^d, n, k)$. □

Alternatively, in the previous statement we can just write $\mathrm{TM}(\mathbb{Z}^d, n, k) = \{\phi \in \mathrm{End}(X_{n,k}) \mid \phi|_Y = \mathrm{id}, \phi^{-1}(Y) \subset Y\}$, since $\phi^{-1}(Y) \supset Y$ is implied by $\phi|_Y = \mathrm{id}$.

Corollary 2.5. *Let $\phi \in \mathrm{TM}(\mathbb{Z}^d, n, k)$. We have that $\phi \in \mathrm{RTM}(\mathbb{Z}^d, n, k)$ if and only if ϕ is bijective.*

Readers familiar with the theory of cellular automata may wonder if injectivity is enough, since injective cellular automata on full shifts are surjective. This is not a priori clear since cellular automata on nontransitive sofic shifts (such as $X_{n,k}$) can be injective without being surjective. We will, however, later prove the stronger result that both injectivity and surjectivity are equivalent to bijectivity.

Clearly, the conditions of Proposition 2.4 are preserved under function composition and inversion. Thus:

Corollary 2.6. *Under function composition, $(\mathrm{TM}(\mathbb{Z}^d, n, k), \circ)$ is a monoid and $(\mathrm{RTM}(\mathbb{Z}^d, n, k), \circ)$ is a group.*

We usually omit the function composition symbol, and use the notations $\mathrm{TM}(\mathbb{Z}^d, n, k)$ and $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ to refer to the corresponding monoids and groups.

An important corollary of Proposition 2.4 is that every result we prove about Turing machine groups says something about cellular automata. In particular, if a group H embeds into $\mathrm{RTM}(\mathbb{Z}, n, k)$, then H also embeds into $\mathrm{Aut}(X_{n,k})$, which implies that there exists a one-dimensional sofic subshift whose automorphism group contains H.[2]

Before defining the second model for Turing machines, we introduce an extended model which will be occasionally used in what follows. Given a subshift $X \subset \Sigma^{\mathbb{Z}^d}$ we denote the set of moving-head $(\mathbb{Z}^d, n, k)$-Turing machines f which satisfy $f(X \times X_k) \subset X \times X_k$ by $\mathrm{TM}(X, k)$ (where we omit the group from the notation, since it is determined by X). And the set of reversible ones by $\mathrm{RTM}(X, k)$. This change basically amounts to replacing the full shift $\Sigma^{\mathbb{Z}^d}$ in the definition by an arbitrary subshift X. This will be used to make explicit which properties from X are required for our results, though our focus is on the case where $X = \Sigma^{\mathbb{Z}^d}$ in which $\mathrm{RTM}(\Sigma^{\mathbb{Z}^d}, k) = \mathrm{RTM}(\mathbb{Z}^d, n, k)$ for $|\Sigma| = n$. Proposition 2.4 is still valid in this extended context.

One use of this generalization is that it allows us to study Turing machines on a more robust class of subshifts than just full shifts. For example, when studying Turing machines on transitive SFTs rather than full shifts only, states can be eliminated, due to the following lemma.

Definition 2.7. *Let $X \subset \Sigma^{\mathbb{Z}}$ be a subshift and $\# \notin \Sigma$. Write $\sqrt[n]{X}$ for the shift space with points*

$$\{x \in (\Sigma \cup \{\#\})^{\mathbb{Z}} : \text{there are } k \in \{0, \dots, n-1\} \text{ and } y \in X \text{ such that for every } m \in \mathbb{Z},$$
$$(x_{mn+k} = y_m \text{ and for every } j \in \{0, \dots, n-1\} \backslash \{k\} \text{ we have } x_{mn+j} = \#)\}.$$

The subshift $\sqrt[n]{X}$ is the space of configurations on $\mathbb{Z}$ where some configuration of X occurs on a coset of $n\mathbb{Z}$ and the rest of the positions are filled with a special symbol $\#$.

Lemma 2.8. *Let $X \subset A^{\mathbb{Z}}$ be a subshift. Then $\mathrm{TM}(X, k)$ is isomorphic to $\mathrm{TM}(\sqrt[k]{X}, 1)$ as a monoid and $\mathrm{RTM}(X, k)$ is isomorphic to $\mathrm{RTM}(\sqrt[k]{X}, 1)$ as a group.*

Proof. As the monoid of Turing machines can be seen as a submonoid of endomorphisms of a subshift, it follows that they commute with the shift. More precisely, for each machine T and $\vec{m} \in \mathbb{Z}$ if we interpret T acting on $X \times X_k$ by endomorphisms, then we have that $\sigma^{\vec{m}} \circ T \circ \sigma^{-\vec{m}} = T$. In particular, as non-# symbols appear at a bounded distance in each configuration of $\mathrm{TM}(\sqrt[k]{X}, 1)$, it suffices to define a Turing machine in $(\sqrt[k]{X} \backslash [\#]_0) \times \{1\} \times \mathbb{Z}$ to completely determine its action over $\sqrt[k]{X} \times \{1\} \times \mathbb{Z}$.

Let $\psi \colon X \to \sqrt[k]{X}$ be defined by

$$\psi(x)_i = \begin{cases} \#, & \text{if } i \not\equiv 0 \bmod k, \text{ and} \\ x_{i/k}, & \text{otherwise.} \end{cases}$$

Note that ψ defines a 1-to-1 map onto $\sqrt[k]{X} \backslash [\#]_0$.

Also, define $\eta \colon \mathbb{Z} \times Q \to \mathbb{Z}$ by $\eta(\vec{v}, q) = k\vec{v} + q - 1$ which is clearly bijective. We can thus finally define a bijection $\delta \colon X \times Q \times \mathbb{Z} \to (\sqrt[k]{X} \backslash [\#]_0) \times \{1\} \times \mathbb{Z}$ by $\delta(x_q^{\vec{v}}) = \psi(x)_1^{\eta(\vec{v}, q)}$.

Now, given a machine $T \in \mathrm{TM}(\sqrt[k]{X}, 1)$ we define $\varphi(T) \in \mathrm{TM}(X, k)$ by $\varphi(T) = \delta^{-1} \circ T \circ \delta$. Note that this is well-defined since $T \in \mathrm{TM}(\sqrt[k]{X}, 1)$ implies that $T(\sqrt[k]{X} \backslash [\#]_0 \times X_1) = \sqrt[k]{X} \backslash [\#]_0 \times X_1$ again because non-#-symbols appear with bounded gaps, and since T is the identity map on points where the head does not appear. By definition it is then clear that $\varphi(T \circ T') = \varphi(T) \circ \varphi(T')$ and that φ is 1-to-1. We only need to show that $\varphi(T) \in \mathrm{TM}(X, k)$ and that φ is onto.

[2] In Section 4.3, we show that $\mathrm{RTM}(\mathbb{Z}, n, k)$ also "almost" embeds into $\mathrm{Aut}(\Sigma^{\mathbb{Z}})$.

Firstly, it is clear that $\delta^{-1} \circ T \circ \delta$ maps $X \times Q \times \mathbb{Z}$ to itself, therefore the head cannot disappear. We have that $\delta \circ \sigma^{\vec{m}} = \sigma^{k\vec{m}} \circ \delta$ and thus $\varphi(T)$ is shift commuting.

$$\begin{aligned}
\varphi(T) \circ \sigma^{\vec{m}} &= \delta^{-1} \circ T \circ \delta \circ \sigma^{\vec{m}} \\
&= \delta^{-1} \circ T \circ \sigma^{k\vec{m}} \circ \delta \\
&= \delta^{-1} \circ \sigma^{k\vec{m}} \circ T \circ \delta \\
&= \sigma^{\vec{m}} \circ \delta^{-1} \circ T \circ \delta \\
&= \sigma^{\vec{m}} \circ \varphi(T).
\end{aligned}$$

Since δ and T are continuous, and δ^{-1} is continuous on the image of δ, we have that $\varphi(T)$ is continuous and shift invariant and therefore defines an endomorphism of $X \times X_k$ which is an element of $\mathrm{TM}(X, k)$. Conversely, an analogous argument shows that for each $T \in \mathrm{TM}(X, k)$ then the map T' defined as $T' = \delta \circ T \circ \delta^{-1}$ on $\sqrt[k]{X}\backslash[\#]_0 \times \{1\} \times \mathbb{Z}$ (and by conjugating with a suitable power of σ on other points) is in $\mathrm{TM}(\sqrt[k]{X}, 1)$ and thus $\varphi(T') = T$, showing that φ is onto.

Finally, if $T \in \mathrm{RTM}(\sqrt[k]{X}, 1)$ then $\varphi(T) \circ \varphi(T^{-1}) = \mathrm{id}$ and thus $\varphi(T) \in \mathrm{RTM}(X, k)$. □

2.2. **The moving tape model.** Even though the moving head model is helpful when building examples, it has a fundamental disadvantage: the space on which the machines act is not compact. From an intuitive point of view, it means that a sequence of Turing machines could potentially move the head to infinity and make it disappear. Or alternatively, in the point of view of seeing Turing machines as endomorphisms as in Proposition 2.4 (which usually, e.g. in Kůrka [27], is directly what the moving head model refers to), the space is compact, but there are uncountably many points that do not quite represent Turing machine configurations.

It's also possible to consider the position of the Turing machine as fixed at $\vec{0}$, and move the tape instead, to obtain the moving tape Turing machine model. In [27], where Turing machines are studied as dynamical systems, the moving head model and moving tape model give non-conjugate dynamical systems. However, the abstract monoids defined by the two points of view turn out to be equal, and we obtain an equivalent definition of the group of Turing machines.

As in the previous section, we begin with a definition using local rules.

Given a function $f\colon \Sigma^{F_{\mathrm{in}}} \times Q \to \Sigma^{F_{\mathrm{out}}} \times Q \times \mathbb{Z}^d$, where $F_{\mathrm{in}}, F_{\mathrm{out}}$ are finite subsets of $\mathbb{Z}^d$, we can define a map $T_f\colon \Sigma^{\mathbb{Z}^d} \times Q \to \Sigma^{\mathbb{Z}^d} \times Q$ as follows: If $f(x|_{F_{\mathrm{in}}}, q) = (p, q', \vec{v})$, then $T_f(x, q) = (\sigma^{-\vec{v}}(y), q')$ where

$$y_{\vec{u}} = \begin{cases} x_{\vec{u}}, & \text{if } \vec{u} \notin F_{\mathrm{out}} \\ p_{\vec{u}}, & \text{if } \vec{u} \in F_{\mathrm{out}}, \end{cases}$$

Definition 2.9. *Any function $T\colon \Sigma^{\mathbb{Z}^d} \times Q \to \Sigma^{\mathbb{Z}^d} \times Q$ for which there is an f as above such that $T = T_f$ is called a **moving tape** $(\mathbb{Z}^d, n, k)$-**Turing machine** and f is its **local rule**. If there exists a $(\mathbb{Z}^d, n, k)$-Turing machine T^{-1} such that $T \circ T^{-1} = T^{-1} \circ T = \mathrm{id}$ we say T is reversible.*

These machines also have the following characterization with a slightly more dynamical feel to it. Say that two configurations x and y in $\Sigma^{\mathbb{Z}^d}$ are **asymptotic** (also called **homoclinic** in the literature), and write $x \sim y$, if there exists a finite $F \subset \mathbb{Z}^d$ such that $x|_{\mathbb{Z}^d \setminus F} = y|_{\mathbb{Z}^d \setminus F}$. In order to be more specific, we write $x \sim_F y$ to claim that a particular choice of F satisfies the property.

Lemma 2.10. *Let $T\colon \Sigma^{\mathbb{Z}^d} \times Q \to \Sigma^{\mathbb{Z}^d} \times Q$ be a function. Then T is a moving tape Turing machine if and only if it is continuous, and there exist a continuous function $s\colon \Sigma^{\mathbb{Z}^d} \times Q \to \mathbb{Z}^d$ and a finite $F \subset \mathbb{Z}^d$ such that $T_1(x, q) \sim_F \sigma^{s(x,q)}(x)$ for all $(x, q) \in \Sigma^{\mathbb{Z}^d} \times Q$.*

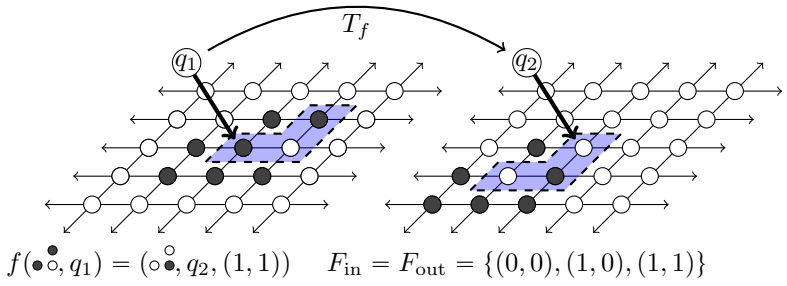


FIGURE 2. The action of a moving tape machine T_f.

Proof. It is easy to see that T_f for any local rule $f\colon \Sigma^{F_{\text{in}}} \times Q \to \Sigma^{F_{\text{out}}} \times Q \times \mathbb{Z}^d$ is continuous. The projection to the third component of f gives the function $-s$, and one can take F as the minimal neighborhood of T_f.

For the converse, since s is a continuous function from a compact space to a discrete one we conclude that the image of s is bounded. Furthermore it only depends on a finite set F_0 of coordinates of x. Since T is continuous, $T(x,q)|_{F+s(\Sigma^{\mathbb{Z}^d}\times Q)}$ depends only on a finite set of coordinates F_1 of x. It is then easy to extract a local rule

$$f\colon \Sigma^{F_0\cup F_1} \times Q \to \Sigma^{F-s(\Sigma^{\mathbb{Z}^d}\times Q)} \times Q \times \mathbb{Z}^d,$$

for T. □

We call the function s in the definition of these machines the **shift indicator** of T, as it indicates how much the tape is shifted depending on the local configuration around $\vec{0}$. In the theory of orbit equivalence and topological full groups, the analogs of s are usually called **cocycles**. Note that in the definition using local functions, the third coordinate of the image indicates how much the head moves, while the shift indicator shows how the configuration shifts, hence the minus sign next to s in the above proof.

Remark 2.11. In the previous lemma it is not enough that $T_1(x,q) \sim \sigma^{s(x,q)}(x)$ for all $(x,q) \in \Sigma^{\mathbb{Z}^d} \times Q$, we need the configurations to be uniformly asymptotic to each other (with a fixed $F \subset \mathbb{Z}^d$). Indeed, let $Q = \{1\}$ and consider the function $T\colon \Sigma^{\mathbb{Z}} \times Q \to \Sigma^{\mathbb{Z}} \times Q$ defined by $(T_1(x,1))_i = x_{-i}$ if $x_{[-|i|+1,|i|-1]} = 0^{2i-1}$ and $\{x_i, x_{-i}\} \neq \{0\}$, and $(T_1(x,1))_i = x_i$ otherwise. Clearly this map is continuous, the constant map $s(x,q) = \vec{0}$ gives a shift-indicator for it and $T_1(x,q) \sim x$ for every $x \in \Sigma^{\mathbb{Z}^d}$. However, T is not defined by any local rule since it can modify the tape arbitrarily far from the origin.

As for moving head machines, it is easy to see (either by constructing local rules or by applying the dynamical definition) that the composition of two moving tape Turing machines is again a moving tape Turing machine. This allows us to proceed as before and define their monoid and group.

Definition 2.12. *We denote by* $\mathrm{TM}_{\text{fix}}(\mathbb{Z}^d, n, k)$ *and* $\mathrm{RTM}_{\text{fix}}(\mathbb{Z}^d, n, k)$ *the monoid of moving tape* $(\mathbb{Z}^d, n, k)$*-Turing machines and the group of reversible moving tape* $(\mathbb{Z}^d, n, k)$*-Turing machines respectively.*

Now, let us show that both models are equivalent in group-theoretical terms. First, we define the natural monoid epimorphism $\Psi\colon \mathrm{TM}(\mathbb{Z}^d, n, k) \to \mathrm{TM}_{\text{fix}}(\mathbb{Z}^d, n, k)$ that shifts the configurations so that the head remains at the origin:

Definition 2.13. *Let $T \in \mathrm{TM}(\mathbb{Z}^d, n, k)$. We define $\Psi(T) \in \mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ as the moving tape Turing machine such that*

$$\Psi(T)(x, q) = (\sigma^{-\vec{v}}(y), r) \text{ when } T(x_q^{\vec{0}}) = y_r^{\vec{v}}.$$

In other terms, define the shift equivalence relation $\equiv$ on the moving head space $\Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d$ by $x_q^{\vec{u}} \equiv \sigma^{\vec{v}}(x)_q^{\vec{u}+\vec{v}}$ for all $x \in \Sigma^{\mathbb{Z}^d}$, $q \in Q$ and $\vec{u}, \vec{v} \in \mathbb{Z}^d$. Then for $T \in \mathrm{TM}(\mathbb{Z}^d, n, k)$, $\Psi(T)$ is the action induced by T on the quotient $\Sigma^{\mathbb{Z}^d} \times Q \times \mathbb{Z}^d/\equiv$. Note that for any two $\equiv$-classes U and V, if $T(U) \subset V$ then $T|_U : U \longrightarrow V$ is a bijection. As we shall see, this shows that Ψ preserves injectivity and surjectivity of Turing machines.

Proposition 2.14. *T is injective (surjective) if and only if $\Psi(T)$ is injective (surjective, respectively).*

Proof. If T is not injective then $T(c_1) = T(c_2)$ for some $c_1 \neq c_2$. Because T is injective in each $\equiv$-class, $c_1 \not\equiv c_2$. Hence there are two distinct $\equiv$-classes with the same image so that $\Psi(T)$ is not injective. Conversely, if $\Psi(T)$ maps two distinct $\equiv$-classes into the same class, there are $c_1 \not\equiv c_2$ such that $T(c_1) \equiv T(c_2)$. Because T is surjective in $\equiv$-classes, there is $c_1' \equiv c_1$ such that $T(c_1') = T(c_2)$, so T is not injective.

If T is surjective then clearly every $\equiv$-class has a pre-image so that $\Psi(T)$ is surjective. And if $\Psi(T)$ is surjective then every $\equiv$-class has a pre-image, and because T is surjective in $\equiv$-classes, every element of every $\equiv$-class has a pre-image, that is, T is surjective. □

The function Ψ is clearly always a monoid epimorphism. It is not injective in the trivial case $n = 1$: Indeed, we have that $\mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d, 1, k)$ is isomorphic to the symmetric group on k symbols, and $\mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d, 1, k)$ is isomorphic to the monoid of all functions from $\{1, \dots, k\}$ to itself, i.e. the symmetric monoid on $\{1, \dots, k\}$. Therefore both of these groups are finite when $n = 1$. On the other hand, clearly $\mathbb{Z}^d$ embeds into $\mathrm{RTM}(\mathbb{Z}^d, 1, k)$ and $\mathrm{TM}(\mathbb{Z}^d, 1, k)$ as the shifts are non-trivial elements of these monoids. Next, we show that in most other cases Ψ is injective.

Intuitively, in order to also have $\mathbb{Z}^d$ embed into the monoid of moving tape Turing machines, we need the configuration space to admit configurations with a certain degree of aperiodicity. We shall see that this is indeed the only obstruction and obtain as a corollary that for every $n \geqslant 2$ the map Ψ is an isomorphism.

Definition 2.15. *A subshift $X \subset \Sigma^{\mathbb{Z}^d}$ is said to be **locally aperiodic** if for every $x \in X$, every finite $F \subset \mathbb{Z}^d$ and every non-zero $\vec{v} \in \mathbb{Z}^d$ there exists $y \in [x|_F]$ such that $y_{\vec{u}} \neq y_{\vec{v}+\vec{u}}$ for some $\vec{u} \in \mathbb{Z}^d$.*

For example, the sunny-side-up subshift $X_1 \subset \{0, 1\}^{\mathbb{Z}}$ is locally aperiodic because every non-empty cylinder contains a configuration with exactly one occurrence of 1, and such a configuration is non-periodic. But this means that the morphism Ψ of Definition 2.13 is not necessarily injective even on reversible Turing machines over locally aperiodic subshifts. Consider, for example, the single state machine $T \in \mathrm{TM}(X_1, 1)$ that maps, for some fixed $\vec{v} \in \mathbb{Z}$,

$$x_1^{\vec{0}} \mapsto \begin{cases} x_1^{\vec{0}}, & \text{if } x_0 = 0, \\ \sigma^{\vec{v}}(x)_1^{\vec{v}}, & \text{if } x_0 = 1. \end{cases}$$

This machine shifts the configuration and the position by $\vec{v}$ if the cell under the head contains the unique 1 of the configuration, and does not do anything if the cell contains symbol 0. The Turing machine is clearly reversible, and $\Psi(T)$ is the identity regardless of the choice of vector $\vec{v}$.

To guarantee injectivity of Ψ we define an even more restrictive variant of local aperiodicity:

Definition 2.16. *A subshift $X \subset \Sigma^{\mathbb{Z}^d}$ is **strongly locally aperiodic** if for every $x \in X$, every finite $F \subset \mathbb{Z}^d$ and every non-zero $\vec{v} \in \mathbb{Z}^d$ there exists $y \in [x|_F]$ such that $y_{\vec{u}} \neq y_{\vec{u}+\vec{v}}$ for some $\vec{u}$ such that $\vec{u}, \vec{u} + \vec{v} \notin F$.*

Obviously $X = \{0, \dots, n-1\}^{\mathbb{Z}^d}$ is strongly locally aperiodic whenever $n \geqslant 2$. With this definition we have the following.

Lemma 2.17. *Let $X \subset \Sigma^{\mathbb{Z}^d}$ be strongly locally aperiodic. We have that*

$$\mathrm{TM}_{\mathrm{fix}}(X,k) \cong \mathrm{TM}(X,k)$$
$$\mathrm{RTM}_{\mathrm{fix}}(X,k) \cong \mathrm{RTM}(X,k).$$

Proof. Consider again the epimorphism $\Psi\colon \mathrm{TM}(\mathbb{Z}^d,n,k) \to \mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d,n,k)$ and suppose there exists a pair $T \neq T'$ in $\mathrm{TM}(X,k)$ such that $\Psi(T) = \Psi(T')$. Let $x_q^{\vec{0}}$ be such that $T(x_q^{\vec{0}}) \neq T'(x_q^{\vec{0}})$. Denoting by $W = F \cup F'$ the union of the neighborhoods F of T and F' of T' we get that T and T' can be described by rules of the form $f_T(x|_W, q) = (p, r, \vec{v})$ and $f_{T'}(x|_W, q) = (p', r', \vec{v}')$. Denote by $x_{[p]}$ and $x_{[p']}$ the configuration where the symbols of x in the support W have been replaced by p and p' respectively. Clearly $\Psi(T) = \Psi(T')$ implies that $r = r'$ and $\sigma^{-\vec{v}}(x_{[p]}) = \sigma^{-\vec{v}'}(x_{[p']})$.

If $\vec{v} = \vec{v}'$ then also $x_{[p]} = x_{[p']}$, which contradicts $T(x_q^{\vec{0}}) \neq T'(x_q^{\vec{0}})$. So we must have $\vec{v} \neq \vec{v}'$. As X is strongly locally aperiodic, there exists $y \in [x|_W]$ such that $y_{\vec{u}} \neq y_{\vec{u}+\vec{v}'-\vec{v}}$ for some position $\vec{u}$ that satisfies $\vec{u}, \vec{u} + \vec{v}' - \vec{v} \notin W$. As $y|_W = x|_W$, we have that $\Psi(T)(y,q) = (\sigma^{-\vec{v}}(y_{[p]}), r)$ and $\Psi(T')(y,q) = (\sigma^{-\vec{v}'}(y_{[p']}), r')$. Then $\Psi(T) = \Psi(T')$ implies that $y_{[p]} = \sigma^{\vec{v}-\vec{v}'}(y_{[p']})$, which is not true in position $\vec{u}$. □

A non-trivial full shift $\Sigma^{\mathbb{Z}^d}$ is strongly locally aperiodic, and thus Lemma 2.17 gives the following corollary.

Corollary 2.18. *If $n \geqslant 2$ then:*

$$\mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d,n,k) \cong \mathrm{TM}(\mathbb{Z}^d,n,k)$$
$$\mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d,n,k) \cong \mathrm{RTM}(\mathbb{Z}^d,n,k).$$

The previous result means that apart from the trivial case $n = 1$ where the tape plays no role, we can study the properties of these groups using any model.

2.3. **The uniform measure and reversibility.** Consider the space $\Sigma^{\mathbb{Z}^d} \times Q$. Let μ be the product of the uniform Bernoulli measure on $\Sigma^{\mathbb{Z}^d}$ and the uniform discrete measure on Q. That is, μ is the measure such that for every finite $F \subset \mathbb{Z}^d$ and $p \in \Sigma^F$, we have

$$\mu([p] \times \{q\}) = \frac{1}{kn^{|F|}}.$$

Theorem 2.19. *Let $T \in \mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d,n,k)$. Then the following are equivalent:*

(1) T is injective.
(2) T is surjective.
(3) $T \in \mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d,n,k)$.
(4) T preserves the uniform measure ($\mu(T^{-1}(A)) = \mu(A)$ for all Borel sets $A \subset \Sigma^{\mathbb{Z}^d} \times Q$).
(5) $\mu(T(A)) = \mu(A)$ for all Borel sets $A \subset \Sigma^{\mathbb{Z}^d} \times Q$.

Proof. Let T be arbitrary, and let F be its minimal neighborhood. Consider the cylinders $C_i = [p_i] \times \{q\}$ where $p_i \in \Sigma^F, q \in Q$. These cylinders form a clopen partition of $\Sigma^{\mathbb{Z}^d} \times Q$ into $kn^{|F|}$ cylinders of measure $\frac{1}{kn^{|F|}}$.

Now, because F is the minimal neighborhood of T, T is a homeomorphism from C_i onto $D_i = T(C_i)$, and D_i is a cylinder set of the form $[p'] \times \{q'\}$ for some $p' \in \Sigma^{\vec{v}+F}$, $q' \in Q$, which must be of the same measure as C_i as the domain $\vec{v} + F$ of p' has as many coordinates as the domain F of p. Note that D_i is not necessarily a cylinder centered at the origin, and the offset $\vec{v}$ is given by the shift-indicator.

Now, observe that injectivity is equivalent to the cylinders $D_i = T(C_i)$ being disjoint. Namely, they must be disjoint if T is injective, and if they are disjoint then T is injective because $T|_{C_i} : C_i \to D_i$ is a homeomorphism. Surjectivity, on the other hand, corresponds to the equality $\Sigma^{\mathbb{Z}^d} \times Q = \bigcup_i D_i$, since $\bigcup_i D_i = \bigcup_i T(C_i) = T(\Sigma^{\mathbb{Z}^d} \times Q)$.

Now, it is easy to show that injectivity and surjectivity are equivalent: If T is injective, then the D_i are disjoint, and $\mu(\bigcup_i D_i) = \sum_i \mu(D_i) = 1$, so we must have $\bigcup_i D_i = \Sigma^{\mathbb{Z}^d} \times Q$ because $\Sigma^{\mathbb{Z}^d} \times Q$ is the only clopen set of full measure. If T is not injective, then for some $i \neq j$ we have $D_i \cap D_j \neq \varnothing$. Then $D = D_i \cap D_j$ is a nonempty clopen set, and thus has positive measure. It follows that $\mu(\bigcup_i D_i) \leqslant \sum_i \mu(D_i) - \mu(D) < 1$, so $\bigcup_i D_i \subsetneq \Sigma^{\mathbb{Z}^d} \times Q$. Of course, since injectivity and surjectivity are equivalent, they are both equivalent to bijectivity, and thus to reversibility of T by Corollary 2.5.

The argument given above in fact shows that reversibility is equivalent to preserving the uniform Bernoulli measure in the forward sense – if T is reversible, then $\mu(T(A)) = \mu(A)$ for all clopen sets A, and thus for all Borel sets, while if T is not reversible, then there is a disjoint union of cylinders $C \cup D$ such that $\mu(T(C \cup D)) < \mu(C \cup D)$.

For measure-preservation in the usual (backward) sense, observe that the reverse of a reversible Turing machine is reversible and thus measure-preserving in the forward sense, so a reversible Turing machine must itself be measure-preserving in the traditional sense. If T is not reversible, then $\mu(T(C \cup D)) < \mu(C \cup D)$ for some disjoint cylinders C and D large enough that $T|_C$ and $T|_D$ are measure-preserving homeomorphisms. Then for $E = T(C) \cap T(D)$ we have $\mu(T^{-1}(E)) \geqslant \mu((T^{-1}(E) \cap C) \cup (T^{-1}(E) \cap D)) = 2\mu(E)$. □

Remark 2.20. The proof is based on showing that every Turing machine is a **local homeomorphism** and preserves the measure of all large-radius cylinders C in the forward sense $\mu(T(C)) = \mu(C)$. Note that preserving the measure of large-radius cylinders in the forward sense does not imply preserving the measure of all Borel sets (or even all cylinders), in general. For example, the machine with $n \geqslant 2, k = 1$ which turns the symbol in $F = \{\vec{0}\}$ to 0 without moving the head satisfies $\mu([p]) = \mu(T[p])$ for any $p \in \Sigma^S$ with $S \supset F$. But $\mu(\Sigma^{\mathbb{Z}^d} \times Q) = 1$ and $\mu(T(\Sigma^{\mathbb{Z}^d} \times Q)) = \mu([0]) = 1/2$.

Remark 2.21. Using Proposition 2.14 and Theorem 2.19 we see that also under the moving head model injectivity and surjectivity are equivalent.

We can also use the uniform measure to define the average movement of a Turing machine.

Definition 2.22. *Let $T \in \mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ with shift indicator function $s\colon \Sigma^{\mathbb{Z}^d} \times Q \to \mathbb{Z}^d$. We define the **average movement** $\alpha(T) \in \mathbb{R}^d$ as*

$$\alpha(T) = \mathbb{E}_\mu(s) = \int_{\Sigma^{\mathbb{Z}^d} \times Q} s(x, q) d\mu,$$

where μ is the uniform measure on $\Sigma^{\mathbb{Z}^d} \times Q$. For T in $\mathrm{TM}(\mathbb{Z}^d, n, k)$ we define α as the application to its image under the canonical epimorphism Ψ from Definition 2.13, that is, $\alpha(T) = \alpha(\Psi(T))$.

Of course, as $s(x, q)$ depends only on finitely many coordinates of x, this integral is actually a finite sum over the cylinders $p \in \Sigma^F$ for some finite $F \subset \mathbb{Z}^d$, and thus we have $\alpha(T) \in \mathbb{Q}^d$ for all T. The following lemma shows that in fact α is an homomorphism.

Lemma 2.23. *The map $\alpha\colon \mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k) \to \mathbb{Q}^d$ is a group homomorphism.*

Proof. If $T_1, T_2 \in \mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ then since reversibility implies measure-preservation, we have

$$\begin{aligned}\alpha(T_1 \circ T_2) &= \mathbb{E}_\mu(s_{T_1 \circ T_2}) \\ &= \mathbb{E}_\mu(s_{T_1} \circ T_2 + s_{T_2}) \\ &= \mathbb{E}_\mu(s_{T_1} \circ T_2) + \mathbb{E}_\mu(s_{T_2}) \\ &= \mathbb{E}_\mu(s_{T_1}) + \mathbb{E}_\mu(s_{T_2}) \\ &= \alpha(T_1) + \alpha(T_2),\end{aligned}$$

where $\mathbb{E}_\mu(s_{T_1 \circ T_2}) = \mathbb{E}_\mu(s_{T_1} \circ T_2 + s_{T_2})$ holds because $s_{T_1 \circ T_2}(x, q) = s_{T_1}(T_2(x, q)) + s_{T_2}(x, q)$ for all $(x, q) \in \Sigma^{\mathbb{Z}^d} \times Q$. □

3. Subgroups and generators

In this section we study several subgroups of $\mathrm{RTM}(\mathbb{Z}^d, n, k)$. The main result of this section is that there is a finitely-generated subgroup of "elementary Turing machines". In the following sections, we show that in pure computability terms, these are able to simulate general Turing machines.

The group of elementary Turing machines $\mathrm{EL}(\mathbb{Z}^d, n, k)$ is the subgroup of reversible Turing machines which is generated by the union of two natural subgroups: $\mathrm{LP}(\mathbb{Z}^d, n, k)$, the group of local permutations and $\mathrm{RFA}(\mathbb{Z}^d, n, k)$, the group of reversible finite-state automata. These two groups separately capture the dynamics of changing the tape and moving the head. We also define the group of oblivious Turing machines $\mathrm{OB}(\mathbb{Z}^d, n, k)$ as an extension of $\mathrm{LP}(\mathbb{Z}^d, n, k)$ where arbitrary tape-independent moves are allowed.

We recall that for the convenience of the reader, a dictionary of these groups can be found in Appendix A.

The main results we prove about these subgroups are the following, when $n \geqslant 2$:

- $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ is not finitely generated (Theorem 3.5),
- $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ is not finitely generated (Theorem 3.11).
- $\mathrm{RFA}(X, 1)$ is generated by "orbitwise shifts" and "controlled position swaps" for any one-dimensional subshift X (Theorem 3.28),
- $\mathrm{OB}(\mathbb{Z}^d, n, k)$ is finitely generated (Theorem 3.18),
- $\mathrm{EL}(\mathbb{Z}, n, k)$ is finitely generated (Theorem 1.1).

For the definitions of "orbitwise shifts" and "controlled position swap', see Section 3.3. For any class $\mathrm{CL}(\mathbb{Z}^d, n, k)$ of Turing machines with moving head we denote by $\mathrm{CL}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ the corresponding class of moving tape machines, that is, the image of the class under the morphism Ψ from Definition 2.13.

3.1. Definitions of subgroups.

3.1.1. *Oblivious Turing machines.* For $\vec{v} \in \mathbb{Z}^d$, define the machine $T_{\vec{v}}$ which does not modify the state or the tape, and moves the head by the vector $\vec{v}$ on each step. Denote $\mathrm{SHIFT}(\mathbb{Z}^d, n, k) = \langle \{T_{\vec{v}}\}_{\vec{v} \in \mathbb{Z}^d} \rangle$. Clearly $\mathrm{SHIFT}(\mathbb{Z}^d, n, k) \cong \mathbb{Z}^d$. Define also $\mathrm{SP}(\mathbb{Z}^d, n, k)$ as the **state-permutations**: Turing machines that never move and only permute their state as a function of the tape.

Definition 3.1. *We define the group* $\mathrm{LP}(\mathbb{Z}^d, n, k)$ *of* ***local permutations*** *as the subgroup of Turing machines in* $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ *whose shift-indicator is trivial, that is, the constant function* $x \mapsto \vec{0}$.

We define also $\mathrm{OB}(\mathbb{Z}^d, n, k) = \langle \mathrm{SHIFT}(\mathbb{Z}^d, n, k), \mathrm{LP}(\mathbb{Z}^d, n, k) \rangle$, *the group of* ***oblivious Turing machines****.*

In other words, $\mathrm{LP}(\mathbb{Z}^d, n, k)$ is the group of reversible machines that do not move the head, and $\mathrm{OB}(\mathbb{Z}^d, n, k)$ is the group of reversible Turing machines whose head movement is independent of the

state and the tape contents. Note that in the definition of both groups, we allow changing the state as a function of the tape, and vice versa, thus $\mathrm{SP}(\mathbb{Z}^d, n, k) \leqslant \mathrm{LP}(\mathbb{Z}^d, n, k)$.

Remark 3.2. The restricted wreath products $H \wr \mathbb{Z}^d$, where H is a finite group, are sometimes called **generalized lamplighter groups**, the original lamplighter group being $\mathbb{Z}/2\mathbb{Z} \wr \mathbb{Z}$. Thus, $\mathrm{OB}(\mathbb{Z}^d, n, k)$ can be seen as a doubly generalized lamplighter group, since the subgroup of $\mathrm{OB}(\mathbb{Z}^d, n, k)$ generated by the local permutations $\mathrm{LP}(\mathbb{Z}^d, n, 1)$ with radius 0 and $\mathrm{SHIFT}(\mathbb{Z}^d, n, 1)$ is isomorphic to the wreath product $S_n \wr \mathbb{Z}^d$ of the symmetric group S_n. Further on we show that, similar to lamplighter groups, $\mathrm{OB}(\mathbb{Z}^d, n, k)$ is also finitely generated.

3.1.2. *Finite-state automata.*

Definition 3.3. *We define the group* $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ *of* ***reversible finite-state automata*** *as the group of reversible* $(\mathbb{Z}^d, n, k)$*-Turing machines that do not change the tape. That is, the local rules are of the form* $f(p, q) = (p, q', \vec{v})$ *for all entries* $p \in \Sigma^F, q \in Q$.

Similarly, for a subshift $X \subset \Sigma^{\mathbb{Z}^d}$ we let $\mathrm{RFA}(X, k)$ be the subgroup of elements in $\mathrm{RTM}(X, k)$ that act without modifying the tape.

This group is "orthogonal" to $\mathrm{OB}(\mathbb{Z}^d, n, k)$ in the following sense,

$$\mathrm{RFA}(\mathbb{Z}^d, n, k) \cap \mathrm{LP}(\mathbb{Z}^d, n, k) = \mathrm{SP}(\mathbb{Z}^d, n, k),$$
$$\mathrm{RFA}(\mathbb{Z}^d, n, k) \cap \mathrm{OB}(\mathbb{Z}^d, n, k) = \langle \mathrm{SP}(\mathbb{Z}^d, n, k), \mathrm{SHIFT}(\mathbb{Z}^d, n, k) \rangle.$$

Remark 3.4. It follows directly from the definitions that the group $\mathrm{RFA}(\mathbb{Z}^d, n, 1)$ is isomorphic to the topological full group of the full $\mathbb{Z}^d$-shift on n symbols as defined in [18]. Similarly, if we fix a subshift X, then $\mathrm{RFA}_{\mathrm{fix}}(X, 1)$ is isomorphic to the topological full group of the shift action on X. The subscript "fix" is only needed when X is not strongly locally aperiodic, see Lemma 2.17.

As usual, the case $n = 1$ is not particularly interesting, and we have that $\mathrm{RFA}(\mathbb{Z}^d, 1, k) = \mathrm{RTM}(\mathbb{Z}^d, 1, k)$. In the general case the group is more interesting.

Theorem 3.5. *Let* $n \geqslant 2$. *Then* $\mathrm{RFA}(\Sigma^{\mathbb{Z}^d}, n, k)$ *is not finitely generated.*

Proof. We prove this in the moving-tape model. For $\vec{v} \in \mathbb{Z}^d$, let $\mathrm{Per}_{\vec{v}}(\Sigma^{\mathbb{Z}^d})$ be the set of configurations of $\Sigma^{\mathbb{Z}^d}$ whose stabilizer under the shift action contains $\vec{v}\mathbb{Z}^d$. Let $\mathbb{Z}_{\geqslant 2}$ be the set of integers $t \geqslant 2$. Let $\phi\colon \mathrm{RFA}_{\mathrm{fix}}(\Sigma^{\mathbb{Z}^d}, n, k) \to (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}_{\geqslant 2}}$ be the parity homomorphism where $\phi(T)_t$ is the sign of the permutation that T performs on the finite set $\mathrm{Per}_{(t,t,\dots,t)}(\Sigma^{\mathbb{Z}^d}) \times Q$.

As the image of a finitely generated group under a homomorphism is also finitely generated, it suffices to show that $\phi(\mathrm{RFA}_{\mathrm{fix}}(\Sigma^{\mathbb{Z}^d}, n, k))$ is not finitely generated. It suffices thus to prove that for any finite $m \geqslant 2$ the restriction of $\phi(\mathrm{RFA}_{\mathrm{fix}}(\Sigma^{\mathbb{Z}^d}, n, k))$ to $(\mathbb{Z}/2\mathbb{Z})^{\{2,\dots,m\}}$ is surjective. From here it clearly follows that $\phi(\mathrm{RFA}_{\mathrm{fix}}(\Sigma^{\mathbb{Z}^d}, n, k))$ is not finitely generated.

Let $t \geqslant 2$ and $\vec{v} = (t, t, \dots, t) \in \mathbb{Z}^d$. Let T_t be the machine which in state $q \neq 1$ acts trivially, and if $q = 1$ does the following: Let $\vec{e} = \vec{e}_1 = (1, 0, \dots, 0, 0)$ be the first canonical basis vector. For $\vec{u} \in \mathbb{Z}^d$ let $A_{\vec{u}} = \vec{u} + \{0, \dots, t-1\} \times \cdots \times \{0, \dots, t-1\}$. Then, on configuration $x \in \Sigma^{\mathbb{Z}^d}$,

- if the restriction of x to $A_{\vec{0}}$ contains a unique 1 which is at $\vec{0}$, and is otherwise identically zero, shift the configuration by $-\vec{e}$,
- if the restriction of x to $A_{\vec{e}}$ contains a unique 1 which is at $\vec{e}$, and it otherwise identically zero, shift the configuration by $\vec{e}$,
- otherwise do nothing.

The machine T_t is an involution and thus is reversible. From this construction it follows that $\phi(T_t)_t = 1$ and $\phi(T_t)_{t'} = 0$ for all $2 \leqslant t' < t$.

Now, let $y \in (\mathbb{Z}/2\mathbb{Z})^{\{2,\dots,m\}}$. Let $M_1 = \mathrm{id}$. Iteratively for $2 \leqslant j \leqslant m$ construct

$$M_j = \begin{cases} M_{j-1} & \text{if } y_j = \phi(M_{j-1})_j \\ T_j \circ M_{j-1} & \text{if } y_j \neq \phi(M_{j-1})_j \end{cases}$$

As ϕ is a homomorphism it follows that $\phi(M_m) = y$ and therefore the restriction of $\phi(\mathrm{RFA}_{\mathrm{fix}}(\Sigma^{\mathbb{Z}^d}, n, k))$ to $(\mathbb{Z}/2\mathbb{Z})^{\{2,\dots,m\}}$ is surjective. □

We shall introduce a new point of view on finite-state machines, which we call the **permutation model**. This model be helpful in the upcoming proof that elementary Turing machines are finitely generated. In this model, we associate to every Turing machine in $\mathrm{RFA}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ an automorphism of $(\Sigma \times (\mathbb{Z}/2\mathbb{Z})^Q)^{\mathbb{Z}^d}$. The main idea behind this correspondence is that given $T \in \mathrm{RFA}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$, every configuration $x \in \Sigma^{\mathbb{Z}^d}$ induces an action over $\mathbb{Z}^d \times Q$, namely, a head pointing at some position in $\mathbb{Z}^d$ in state q would move to a new pair (position,state) in $\mathbb{Z}^d \times Q$ after applying T. As T is reversible, this induces a permutation over $\mathbb{Z}^d \times Q$ which can be applied simultaneously to an arbitrary number of heads, which can in turn be represented as a configuration in $((\mathbb{Z}/2\mathbb{Z})^Q)^{\mathbb{Z}^d}$. We make this embedding precise in the proof of the following proposition.

Proposition 3.6. *The group* $\mathrm{RFA}_{fix}(\mathbb{Z}^d, n, k)$ *embeds into* $\mathrm{Aut}(A^{\mathbb{Z}^d})$ *for* $|A| = n2^k$.

Proof. Note that any pair $(T, x) \in \mathrm{RFA}_{\mathrm{fix}}(\mathbb{Z}^d, n, k) \times \Sigma^{\mathbb{Z}^d}$ induces a permutation $\sigma_{T,x}$ over $\mathbb{Z}^d \times Q$. Namely, let s be the shift-indicator of T, and let

$$\sigma_{T,x}(\vec{v}, q) = (\vec{v} - s(\sigma^{-\vec{v}}(x), q), T_2(\sigma^{-\vec{v}}(x), q))$$

We identify A as the alphabet $\Sigma \times (\mathbb{Z}/2\mathbb{Z})^Q$ and thus the configurations of $A^{\mathbb{Z}^d}$ can be seen as pairs $(x, y) \in \Sigma^{\mathbb{Z}^d} \times ((\mathbb{Z}/2\mathbb{Z})^Q)^{\mathbb{Z}^d}$.

Finally, let $\varphi \colon \mathrm{RFA}_{\mathrm{fix}}(\mathbb{Z}^d, n, k) \to \mathrm{Aut}(\mathcal{A}^{\mathbb{Z}^d})$ be the map defined by $\varphi(T) = \phi_T$ where:

$$(\phi_T)_1(x, y)_{\vec{v}} = x_{\vec{v}},$$

$$(\phi_T)_2(x, y)_{\sigma_{x,T}(\vec{v},q)} = 1 \text{ if and only if } y_{(\vec{v},q)} = 1.$$

That is, ϕ_T does not modify the $\Sigma^{\mathbb{Z}^d}$ tape, and for every position $(\vec{v}, q) \in \mathbb{Z}^d \times Q$ marked with a one, we interpret it as a Turing machine head in state q in position $\vec{v}$ and mark in the image the state and position it would end up after applying T. This is clearly a cellular automaton on $\Sigma^{\mathbb{Z}^d} \times ((\mathbb{Z}/2\mathbb{Z})^Q)^{\mathbb{Z}^d}$ as the shift indicator of T has a finite radius.

We claim φ is an embedding. A direct computation shows that the permutation induced by $T_1 \circ T_2$ is just $\sigma_{T_1,x} \circ \sigma_{T_2,x}$ thus showing that φ is a homomorphism. Now, if $T_1 \neq T_2$ there is some pair (x, q) where they act differently. If we consider the configuration (x, y) where $y_{0^d,q} = 1$ and 0 elsewhere. Clearly $\phi_{T_1}(x, y) \neq \phi_{T_2}(x, y)$. Therefore φ is injective. □

Definition 3.7. *For any* $T \in \mathrm{RFA}_{fix}(\mathbb{Z}^d, n, k)$ *the automorphism* $\phi_T \in \Sigma^{\mathbb{Z}^d} \times (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}^d \times Q}$ *obtained by appling the embedding of the previous proof is called the* ***permutation model*** *of* T.

The permutation model has the remarkable property of being linear in the second component. Namely, given $T \in \mathrm{RFA}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$ and its permutation model ϕ_T, we have that for any $x \in \Sigma^{\mathbb{Z}^d}$ and $y, z \in (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}^d \times Q}$:

$$\phi_T(x, y + z) = (x, (\phi_T)_2(x, y) + (\phi_T)_2(x, z)),$$

where the sum is computed coordinate-wise.

It is known that the automorphism group of any nontrivial full shift embeds into the automorphism group of any uncountable sofic shift [35]. Thus we have the following corollary:

Corollary 3.8. *If X is an uncountable sofic $\mathbb{Z}$-subshift, then* $\mathrm{RFA}(\mathbb{Z}, n, k)$ *embeds into* $\mathrm{Aut}(X)$*. In particular* $\mathrm{RFA}(\mathbb{Z}, n, k)$ *embeds into* $\mathrm{Aut}(\mathcal{A}^{\mathbb{Z}})$ *for any* $|\mathcal{A}| \geqslant 2$*.*

3.1.3. *Elementary Turing machines.*

Definition 3.9. *The group of* ***elementary Turing machines*** $\mathrm{EL}(\mathbb{Z}^d, n, k)$ *is the group generated by finite-state machines and local permutation, that is*

$$\mathrm{EL}(\mathbb{Z}^d, n, k) = \langle \mathrm{RFA}(\mathbb{Z}^d, n, k), \mathrm{LP}(\mathbb{Z}^d, n, k) \rangle.$$

The group of elementary Turing machines is generated by machines which either do not change the tape or do not move the head, and its aim is to approximate the group $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ using simple building blocks. In other words, it can be understood as a sufficiently rich class of Turing machines which can be constructed from simple atoms. For instance, Langdon's ant [29] is an example of a machine in $\mathrm{EL}(\mathbb{Z}^2, 2, 4)$.

Clearly the group $\mathrm{LP}(\mathbb{Z}^d, n, k)$ is not finitely generated because it is locally finite and infinite. We have also shown that $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ is not finitely generated and will soon show that $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ is not finitely generated either. However, we are later going to show that both $\mathrm{OB}(\mathbb{Z}^d, n, k)$ and $\mathrm{EL}(\mathbb{Z}, n, k)$ are finitely generated.

Before studying these machines, we show that $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ is not finitely generated. This proof uses the average movement homomorphism α defined in Section 2.3.

Lemma 3.10.

$$\alpha(\mathrm{RTM}(\mathbb{Z}^d, n, k)) = \left\langle \frac{e_i}{kn^j} : j \in \mathbb{N}, i \in \{1, \dots, d\} \right\rangle \leqslant (\mathbb{Q}^d, +)$$

Proof. Consider the $(\mathbb{Z}, n, k)$-Turing machine $T_{\mathtt{SURF},m}$ given by the local function $f\colon \Sigma^{\{0,\dots,m\}} \times Q \to \Sigma^{\{0,\dots,m\}} \times Q \times \mathbb{Z}$ which is defined as follows: For $a \in \Sigma$ and $q < k$ let $f(0^m a, q) = (0^m a, q+1, 0)$ and $f(0^m a, k) = (a0^m, 1, 1)$. Otherwise $f(u, q) = (u, q, 0)$. This machine is reversible, and satisfies that $\alpha(T_{\mathtt{SURF},m}) = 1/kn^m$. This machine can easily be extended to a $(\mathbb{Z}^d, n, k)$-Turing machine with average movement $(1/kn^m, 0, \dots, 0)$, and an analogous construction yields a Turing machine with the exact same movement in other coordinates. Thus we obtain that

$$\left\langle \frac{e_i}{kn^j} : j \in \mathbb{N}, i \in \{1, \dots, d\} \right\rangle \leqslant \alpha(\mathrm{RTM}(\mathbb{Z}^d, n, k)).$$

To obtain the equality, observe that the integral $\int_{\Sigma^{\mathbb{Z}^d} \times Q} s(x, q) d\mu$ defining the average movement is a finite sum over cylinders, the contribution of each cylinder is an integer vector, and the measure of a cylinder is in the group generated by $\frac{1}{kn^j}$. Thus every element in $\alpha(\mathrm{RTM}(\mathbb{Z}^d, n, k))$ can be written as a finite sum of $\frac{e_i}{kn^j}$. □

The image under the homomorphism α of $(\mathrm{RTM}(\mathbb{Z}^d, n, k))$ computed above is not finitely generated whenever $n \geqslant 2$, therefore we conclude that $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ cannot be finitely generated for $n \geqslant 2$.

Theorem 3.11. *For $n \geqslant 2$, the group* $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ *is not finitely generated.*

Although α is not a homomorphism on $\mathrm{TM}(\mathbb{Z}^d, n, k)$, using Theorem 2.19 we obtain that $\mathrm{TM}(\mathbb{Z}^d, n, k)$ cannot be finitely generated either.

Theorem 3.12. *For $n \geqslant 2$ the monoid* $\mathrm{TM}(\mathbb{Z}^d, n, k)$ *is not finitely generated.*

Proof. Let $T \in \mathrm{RTM}(\mathbb{Z}^d, n, k)$. As T is injective, we conclude that if $T = T_n \circ \cdots \circ T_1$ then T_1 is also injective. By Theorem 2.19 this means that T_1 is reversible, which in turn implies T_2 is injective and so on. Therefore, if $\mathrm{TM}(\mathbb{Z}^d, n, k)$ is generated by S, then there exists $S' \subset S$ such that $\langle S' \rangle = \mathrm{RTM}(\mathbb{Z}^d, n, k)$. But every such S' is infinite by Theorem 3.11 and thus S must also be infinite. □

In the remainder of the section we study how close $\mathrm{EL}(\mathbb{Z}^d, n, k)$ is to $\mathrm{RTM}(\mathbb{Z}^d, n, k)$. We also give a natural infinite generating set for $\mathrm{RFA}(X, k)$ for every $\mathbb{Z}$-subshift X. Later on we show that $\mathrm{EL}(\mathbb{Z}, n, k)$ is finitely generated which implies that $\mathrm{EL}(\mathbb{Z}, n, k)$ is strictly contained in $\mathrm{RTM}(\mathbb{Z}, n, k)$. However, we will first provide a simple proof of that fact which works in any dimension.

Proposition 3.13. *Let* $\frac{1}{k}\mathbb{Z} = \{\frac{n}{k} \mid n \in \mathbb{Z}\}$. *We have that*

$$\alpha(\mathrm{RFA}(\mathbb{Z}^d, n, k)) = \alpha(\mathrm{EL}(\mathbb{Z}^d, n, k)) = \left(\frac{1}{k}\mathbb{Z}\right)^d.$$

In particular, $\mathrm{EL}(\mathbb{Z}^d, n, k)$ *is strictly contained in* $\mathrm{RTM}(\mathbb{Z}^d, n, k)$.

Proof. Clearly $\alpha(T) = 0$ for every $T \in \mathrm{LP}(\mathbb{Z}, n, k)$, so the first equality holds. Let us then consider average movement values of finite-state automata. The machine T_j that increments the state by 1 on each step (modulo the number of states) and walks one step along the jth axis whenever it enters the state 1, has $\alpha(T_j) = (0, \ldots, 0, 1/k, 0, \ldots, 0)$. We obtain

$$\left(\frac{1}{k}\mathbb{Z}\right)^d = \langle \alpha(T_j) \mid 1 \leqslant j \leqslant d \rangle \leqslant \alpha(\mathrm{RFA}(\mathbb{Z}^d, n, k)).$$

Next, let us show that for every finite-state machine T, we have $\alpha(T) \in (\frac{1}{k}\mathbb{Z})^d$. For this, consider the behavior of T on the all-zero configuration. Given a fixed state q, T moves by an integer vector $\vec{v}_q$, thus contributing $\frac{1}{k}\vec{v}_q$ to the average movement. Let $\vec{v} = \sum_{q \in Q} \frac{1}{k}\vec{v}_q$ be the average movement of T on the all-zero configuration.

We claim that $\alpha(T) = \vec{v}$. Note that by composing T with a suitable combination of the machines T_j and their inverses, it is enough to prove this in the case $\vec{v} = \vec{0}$. Now, for a large m, Let $p \in \Sigma^{\{-m,\ldots,m\}^d}$ be a pattern, $\vec{u} \in \{-m, \ldots, m\}^d$ a position and $q \in Q$ a state. Complete p to a configuration $x_p \in \Sigma^{\mathbb{Z}^d}$ by writing 0 in every cell outside $\{-m, \ldots, m\}^d$. Write $\alpha_m(T)$ for the average movement of T for the finitely many choices of p, u, q. Formally, if s_T is the shift indicator of T:

$$\alpha_m(T) = \frac{1}{k(2m+1)^d n^{(2m+1)^d}} \sum_{p,u,q} s_T(\sigma^{-\vec{u}}(x_p), q).$$

As $m \to \infty$, it is easy to show that $\alpha_m(T) \to \alpha(T)$, as the movement vector of T is distributed correctly in all positions except at the boundary of $\{-m, \ldots, m\}^d$ which grows as $o(m^d)$.

On the other hand, for any fixed $p \in \Sigma^{\{-m,\ldots,m\}^d}$ the average movement of T on x_p started from a random state and a random position is $\vec{0}$, that is,

$$\sum_{u,q} s_T(\sigma^{-\vec{u}}(x_p), q) = \vec{0}.$$

This follows from the fact that $T \in \mathrm{RFA}(\mathbb{Z}^d, n, k)$ and thus the action is simply a permutation of the set of position-state pairs and the fact that $\vec{v} = \vec{0}$. From here we conclude that the sum restricted to $u \in \{-m, \ldots, m\}^d$ is $o(m^d)$. It follows that $\alpha(T) = \lim \alpha_m(T) = \vec{0}$. □

3.2. **The oblivious Turing machines are finitely generated.** In this section, we show that $\mathrm{OB}(\mathbb{Z}^d, n, k)$ is finitely generated. Our proof is based on the existence of strongly universal reversible gates for permutations of A^m, which can be found in [1, 40] for the binary alphabet case, and generalized to other alphabets in [8]. We need a finite generating set for permutations of $Q \times \Sigma^m$, and hence the

proof in [8] has to be adjusted to account for non-homogeneous alphabet sizes (that is, due to possibly having $n \neq k$).

Let us remark that the case $n = 1$ is trivial: The group $\mathrm{LP}(\mathbb{Z}^d, 1, k)$ is finite and $\mathrm{SHIFT}(\mathbb{Z}^d, 1, k)$ is generated by the single step moves. We hence assume that $n \geqslant 2$.

The following result was proved in [8] (Lemmas 3 and 5):

Lemma 3.14. *Let $H = (V, E)$ be a connected undirected graph.*

(a) The transpositions $(s\ t)$ for $\{s, t\} \in E$ generate $\mathrm{Sym}(V)$, the set of permutations of the vertex set.

(b) Let $\Delta \subset \mathrm{Sym}(V)$ be a set of permutations of V that contains for each edge $\{s, t\} \in E$ a 3-cycle $(x\ y\ z)$ where $\{s, t\} \subset \{x, y, z\}$. Then Δ generates $\mathrm{Alt}(V)$, the set of even permutations of the vertex set.

Let $m \geqslant 1$, and consider permutations of $Q \times \Sigma^m$. **Controlled swaps** are transpositions $(s\ t)$ where $s, t \in Q \times \Sigma^m$ have Hamming distance one. **Controlled 3-cycles** are permutations $(s\ t\ u)$ where the Hamming distances between the three pairs are 1, 1 and 2.

Let us denote by $C_m^{(2)}$ and $C_m^{(3)}$ the sets of controlled swaps and 3-cycles in $\mathrm{Sym}(Q \times \Sigma^m)$ respectively. Let $H = (V, E)$ be the graph with vertices $V = Q \times \Sigma^m$ and edges $\{s, t\}$ that connect elements s and t having Hamming distance one. This is a connected undirected graph, so we get from Lemma 3.14(b) that the controlled 3-cycles generate all its even permutations:

Lemma 3.15. *Let $n \geqslant 2$ and $m \geqslant 1$. The group $\mathrm{Alt}(Q \times \Sigma^m)$ is generated by $C_m^{(3)}$.*

Let $\ell \leqslant m$, and let f be a permutation of $Q \times \Sigma^\ell$. We can apply f on $1 + \ell$ coordinates of $Q \times \Sigma^m$ (including the first), while leaving the other $m - \ell$ coordinates untouched. More precisely, the **prefix application** $\hat{f}$ of f on $Q \times \Sigma^m$, defined by

$$\hat{f}(q, s_1, \ldots, s_\ell, \ldots, s_m) = (f_1(q, s_1, \ldots, s_\ell), \ldots, f_{\ell+1}(q, s_1, \ldots, s_\ell), s_{\ell+1}, \ldots, s_m),$$

applies f on the first $1 + \ell$ coordinates. To apply it on other choices of coordinates we conjugate $\hat{f}$ using rewirings of symbols. For any permutation $\pi \in \mathrm{Sym}(\{1, \ldots, m\})$ we define the **rewiring** permutation of $Q \times \Sigma^m$ by

$$r_\pi : (q, s_1, \ldots, s_m) \mapsto (q, s_{\pi(1)}, \ldots, s_{\pi(m)}).$$

It permutes the positions of the m tape symbols according to π. Now we can conjugate the prefix application $\hat{f}$ using a rewiring to get $\hat{f}_\pi = r_\pi^{-1} \circ \hat{f} \circ r_\pi$, we call $\hat{f}_\pi$ an **application** of f in the coordinates $\pi(1), \ldots, \pi(\ell)$. Let us denote by

$$[f]_m = \{\hat{f}_\pi \mid \pi \in \mathrm{Sym}(m)\},$$

the set of permutations of $Q \times \Sigma^m$ that are applications of f. For a set P of permutations we denote by $[P]_m$ the union of $[f]_m$ over all $f \in P$.

Note that if n is even and $f \in \mathrm{Sym}(Q \times \Sigma^\ell)$ for $\ell < m$ then $[f]_m$ only contains even permutations. The reason is that the coordinates not participating in the application of f carry a symbol of the even alphabet Σ. The application $[f]_m$ then consists of an even number of disjoint permutations of equal parity – hence the result is even. In contrast, for the analogous reason, if n is odd then $[f]_m$ only contains odd permutations whenever f is itself is an odd permutation.

Lemma 3.16. *Let $m \geqslant 6$, and let $G_m = \langle [C_4^{(2)}]_m \rangle$ be the group generated by the applications of controlled swaps of $Q \times \Sigma^4$ on $Q \times \Sigma^m$. If $n = |\Sigma|$ is odd then $G_m = \mathrm{Sym}(Q \times \Sigma^m)$. If n is even then $G_m = \mathrm{Alt}(Q \times \Sigma^m)$.*

Proof. For even n, by the note above, $[C_4^{(2)}]_m \subset \mathrm{Alt}(Q \times \Sigma^m)$, and for odd n there are odd permutations in $[C_4^{(2)}]_m$. So in both cases it is enough to show $\mathrm{Alt}(Q \times \Sigma^m) \subset G_m$. We also note that, obviously, $[G_{m-1}]_m \subset G_m$.

Based on the decomposition in Figure 3, we first conclude that any controlled 3-cycle f of $Q \times \Sigma^m$ is a composition of four applications of controlled swaps of $Q \times \Sigma^{m-2}$. In the figure, the components of $Q \times \Sigma^m$ have been ordered in parallel horizontal wires, the Q-component being among the topmost three wires. Referring to the symbols in the illustration, the gate on the left is a generic 3-cycle $(pszabcdw\ ptzabcdw\ qszabcdw)$ where one of the first three wires is the Q-component, $a, b, c, d \in \Sigma$ and $w \in \Sigma^{m-6}$. The proposed decomposition consists of two different controlled swaps p_1 and p_2 applied twice in the order $f = p_1p_2p_1p_2$. Because p_1 and p_2 are involutions, the decomposition amounts to identity unless the input is of the form $xyzabcdw$ where $x \in \{p, q\}$ and $y \in \{s, t\}$. When the input is of this form, it is easy to very that the circuit on the right indeed amounts to the required 3-cycle. We conclude that $C_m^{(3)} \subset \langle [C_{m-2}^{(2)}]_m \rangle$, for all $m \geqslant 6$. By Lemma 3.15,

$$\mathrm{Alt}(Q \times \Sigma^m) = \langle C_m^{(3)} \rangle \subset \langle [C_{m-2}^{(2)}]_m \rangle. \tag{1}$$

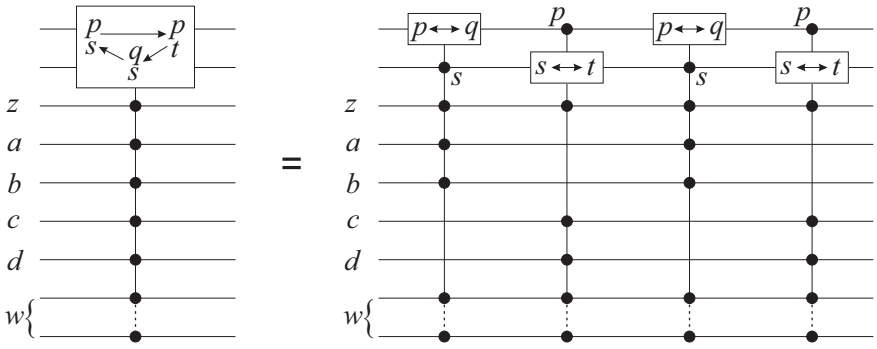


FIGURE 3. A decomposition of a controlled 3-cycle of $Q \times \Sigma^m$ on the left into a sequence of four applications of controlled swaps of $Q \times \Sigma^{m-2}$ on the right. The ordering of the wires is such that topmost three wires contain the Q-component and the two wires changed by the 3-cycle (one of which may or may not be the Q-component). Black circles are control points: the gate computes the identity unless the wire carries the symbol indicated at the left of the wire or next to the control point.

We proceed by induction on m. The base case $m = 6$ is clear: By (1),

$$\mathrm{Alt}(Q \times \Sigma^6) \subset \langle [C_4^{(2)}]_6 \rangle = G_6.$$

Consider then $m > 6$ and suppose that G_{m-1} is as claimed. If n is odd then, by the inductive hypothesis,

$$[C_{m-2}^{(2)}]_m \subset [\mathrm{Sym}(Q \times \Sigma^{m-1})]_m \subset [G_{m-1}]_m \subset G_m.$$

By (1) then $\mathrm{Alt}(Q \times \Sigma^m) \subset \langle [C_{m-2}^{(2)}]_m \rangle \subset G_m$. As pointed out above, G_m contains odd permutations (all elements of $[C_4^{(2)}]_m$ are odd), so $G_m = \mathrm{Sym}(Q \times \Sigma^m)$ as claimed.

If n is even then an application of a permutation of $Q \times \Sigma^{m-2}$ on $Q \times \Sigma^m$ is also an application of an even permutation of $Q \times \Sigma^{m-1}$ on $Q \times \Sigma^m$. (For this reason we left two non-controlling wires for the gates on the right side of Figure 3.) By this and the inductive hypotheses,

$$[C_{m-2}^{(2)}]_m \subset [\mathrm{Alt}(Q \times \Sigma^{m-1})]_m \subset [G_{m-1}]_m \subset G_m,$$

so, by (1), we have the required $\mathrm{Alt}(Q \times \Sigma^m) \subset G_m$. □

Corollary 3.17. $[\mathrm{Sym}(Q \times \Sigma^m)]_{m+1} \subset \langle [\mathrm{Sym}(Q \times \Sigma^4)]_{m+1} \rangle$ *for all* $m \geqslant 5$.

Proof. If n is even then $[\mathrm{Sym}(Q\times\Sigma^m)]_{m+1}\subset\mathrm{Alt}(Q\times\Sigma^{m+1})$ and if n is odd then $[\mathrm{Sym}(Q\times\Sigma^m)]_{m+1}\subset\mathrm{Sym}(Q\times\Sigma^{m+1})$. In either case, the claim follows from Lemma 3.16 and $C_4^{(2)}\subset\mathrm{Sym}(Q\times\Sigma^4)$. □

In Corollary 3.17, arbitrary permutations of $Q\times\Sigma^m$ are obtained as projections of permutations of $Q\times\Sigma^{m+1}$. The extra symbol is an ancilla that can have an arbitrary initial value and is returned back to this value in the end. Such an ancilla is called a "borrowed bit" in [40]. It is needed in the case of even n to facilitate implementing odd permutations of $Q\times\Sigma^m$.

Now we are ready to prove the following theorem.

Theorem 3.18. $\mathrm{OB}(\mathbb{Z}^d,n,k)$ *is finitely generated.*

Proof. We construct a finite generating set $A_1\cup A_2\cup A_3$. Let A_1 contain the one step moves T_{e_i} for $i=1,\dots,d$. These clearly generate $\mathrm{SHIFT}(\mathbb{Z}^d,n,k)$.

Each $T\in\mathrm{LP}(\mathbb{Z}^d,n,k)$ is defined by a local rule $f\colon\Sigma^F\times Q\to\Sigma^F\times Q\times\{\vec{0}\}$ with a finite $F\subset\mathbb{Z}^d$. To have injectivity, we clearly need that $\pi\colon(p,q)\mapsto(f_1(p,q),f_2(p,q))$ is a permutation of $\Sigma^F\times Q$. We denote $T=P_\pi$. Let us fix an arbitrary $E\subset\mathbb{Z}^d$ of size 4, and let A_2 be the set of all $P_\pi\in\mathrm{LP}(\mathbb{Z}^d,n,k)$ determined by $\pi\in\mathrm{Sym}(\Sigma^E\times Q)$.

For any permutation α of $\mathbb{Z}^d$ with finite support, we define the cell permutation machine $C_\alpha\colon(p,q)\mapsto(p',q)$, where $p'_{\vec{v}}=p_{\alpha(\vec{v})}$ for all $\vec{v}\in\mathbb{Z}^d$. These are clearly in $\mathrm{LP}(\mathbb{Z}^d,n,k)$. We take A_3 to consist of the cell permutation machines $C_i=C_{(0\ e_i)}$ that, for each $i=1,\dots,d$, swaps the contents of the currently scanned cell and its neighbor with offset e_i.

Observe that A_1 and A_3 generate all cell permutation machines C_α. First, conjugating C_i with $T_{\vec{v}}\in\mathrm{SHIFT}(\mathbb{Z}^d,n,k)$ gives the cell permutation machine $C_\alpha=T_{\vec{v}}^{-1}C_iT_{\vec{v}}$ for the transposition $\alpha=(\vec{v}\ \vec{v}+e_i)$. Such transpositions generate all permutations of $\mathbb{Z}^d$ with finite support: This follows from Lemma 3.14(a) by considering a finite connected grid graph containing the support of the permutation.

Consider then an arbitrary $P_\pi\in\mathrm{LP}(\mathbb{Z}^d,n,k)$, where $\pi\in\mathrm{Sym}(\Sigma^F\times Q)$. We can safely assume $|F|\geqslant 5$. Let us pick one ancilla $v\in\mathbb{Z}^d\backslash F$ and denote $F'=F\cup\{v\}$. By Corollary 3.17, P_π is a composition of machines of type P_ρ for $\rho\in\mathrm{Sym}(\Sigma^H\times Q)$ where $H\subset F'$ has size $|H|=4$. It is enough to be able to generate these. Let α be a permutation of $\mathbb{Z}^d$ that exchanges E and H, two sets of cardinality four. Then $C_\alpha^{-1}P_\rho C_\alpha\in A_2$, which implies that P_ρ is generated by $A_1\cup A_2\cup A_3$. □

3.3. **Generators for finite-state automata on any one-dimensional subshift.** In this section, we show that while finite-state automata are not finitely generated on the one-dimensional full shift, the group of finite-state automata has a natural generating set on every one-dimensional subshift.

We shall show that a generating set for this group is composed of two types of objects: there is a (possibly infinite rank) abelian group that translates orbits and is an abstracted notion of average movement, and a collection of elements with zero average-movement which is generated by "controlled position swaps" which are similar in spirit to the controlled swaps of the previous section. We show this result for the topological full group (one state) and will extend it to the case of multiple states using Lemma 2.8.

Similar results are known for topological full groups of minimal systems (see for example [31, 19]). The main additional issue is with average movement, which does not actually apply in our main application of the full shift. Usually aperiodicity is assumed when studying topological full groups, but periodic points do not pose any problems, except for the small issue that without strong local aperiodicity we might not have $\mathrm{RFA}_{\mathrm{fix}}(X,k)\cong\mathrm{RFA}(X,k)$. We study $\mathrm{RFA}(X,k)$, understanding that the groups are the same in all cases we are actually interested in.

As suggested, we begin by showing that on any $\mathbb{Z}$-subshift, there is a natural generalization of the average-movement homomorphism which measures the average movement separately on every orbit.

This homomorphism coincides with the average-movement α when $X = \Sigma^{\mathbb{Z}}$, and on minimal subshifts it corresponds to the index map defined for topological full groups [18].

The results of this section will rely strongly on our subshifts being one-dimensional. Our analogue of the index map will be based on counting heads to the left and right of some interval, which is not straightforward to generalize to the multidimensional case.

Suppose for the rest of this section that X denotes a one-dimensional subshift. In this section, we write $\mathrm{RFA}(X) = \mathrm{RFA}(X, 1)$. We can see the shift σ as an element of $\mathrm{RFA}(X)$ defined by $\sigma(x_1^i) = x_1^{i-1}$.

Let $\sim_n'$ be the relation on the language $L_n(X)$ of words of length n appearing on X defined by $u \sim_n' v$ if there exists $x \in X$ such that u and v occur in x. Let $\sim_n$ be the transitive closure of $\sim_n'$. We write $\sim$ for $\sim_n$ the equivalence relation induced by the collection of $\sim_n$ for every $n \in \mathbb{N}$.

Definition 3.19. *A subshift $X \subset \Sigma^{\mathbb{Z}}$ is **weakly chain-transitive** if for every $n \in \mathbb{N}$ and $u, v \in L_n(X)$ we have $u \sim v$.*

We use the prefix "weak" to distinguish this notion from other (stronger) notions of chain-transitivity in the literature such as the one found in [26].

For any subshift X, let $\widehat{X} \subset \prod_n L_n(X)/\sim_n$ be the image of X under the factor map $\pi(x)_n = x_{[0,n-1]}/\sim_n$. Considering $\widehat{X}$ as a dynamical system with a trivial $\mathbb{Z}$-action, it is easy to see that $\pi\colon X \to \widehat{X}$ is a factor map: $x_{[0,n-1]} \sim x_{[-1,n-2]}$ for all $x \in X$ and $n \in \mathbb{N}$, so $\pi(\sigma(x)) = \pi(x) = \mathrm{id}(\pi(x))$. Notice that X is weakly chain-transitive if and only if $\widehat{X}$ is a singleton.

Recall from Definition 2.7 that given a subshift $X \subset \Sigma^{\mathbb{Z}}$, we define a subshift $\sqrt[k]{X} \subset (\Sigma \cup \{\#\})^{\mathbb{Z}}$ where configurations of X occur in a coset of $k\mathbb{Z}$ and all other positions are filled with the symbol $\#$.

Lemma 3.20. *Let X be a weakly chain-transitive subshift. Then $\sqrt[k]{X}$ is weakly chain-transitive.*

Proof. Let u', v' be two words of the same length in $\sqrt[k]{X}$. Then possibly by extending u' and v' we may write them as subwords of words u'' and v'' of the form

$$u' \sqsubset u'' = \#^{n-1}u_1\#^{k-1}u_2\#^{k-1}u_3\cdots u_n\#^{k-1},$$

$$v' \sqsubset v'' = \#^{k-1}v_1\#^{k-1}v_2\#^{k-1}v_3\cdots v_n\#^{k-1},$$

for some $n \in \mathbb{N}$ and words u, v in $L_n(X)$. Since X is weakly chain-transitive, there is a finite chain $u = w_0 \sim w_1 \sim \cdots \sim w_k = v$ such that w_i and w_{i+1} occur in the same point of X for all i. Then the corresponding interspersed versions of the w_i give a chain between u'' and v'', and thus between u' and v'. □

Definition 3.21. *We say that $T \in \mathrm{RFA}(X)$ is an **orbitwise shift** if for every $x \in X$ there exists $k \in \mathbb{Z}$ such that*

$$T(x_1^i) = x_1^{i-k} \text{ for every } i \in \mathbb{Z}.$$

*An **abstract orbitwise shift** is a continuous function $f\colon X \to \mathbb{Z}$ such that $f(x) = f(\sigma(x))$ for all $x \in X$. Write $\mathrm{OS}(X)$ and $\mathrm{AOS}(X)$ for the group of orbitwise shifts and the group of abstract orbitwise shifts respectively.*

Note that orbitwise shifts form a subgroup of $\mathrm{RFA}(X)$, and abstract orbitwise shifts form a group under pointwise addition. Orbitwise shifts are a much smaller group than $\mathrm{RFA}(X)$ in general. For example $\mathrm{OS}(X)$ is always abelian, while one can show that $\mathrm{RFA}(X)$ may contain a free group on two generators, see for example [14] or [4].

Lemma 3.22. *The abstract orbitwise shifts are precisely the continuous functions $f\colon X \to \mathbb{Z}$ that factor through $\widehat{X}$ in the sense that $f = g \circ \pi$ for some continuous map $g\colon \widehat{X} \to \mathbb{Z}$.*

Proof. Suppose that $f = g \circ \pi$ for some $g\colon \widehat{X} \to \mathbb{Z}$. Then f is continuous as the composition of two continuous functions, and

$$f(x) = g(\pi(x)) = g(\pi(\sigma(x))) = f(\sigma(x)),$$

so f is an abstract orbitwise shift.

On the other hand, if $f\colon X \to \mathbb{Z}$ is continuous and $f(x) = f(\sigma(x))$, then the map $g\colon \widehat{X} \to \mathbb{Z}$ given by the formula $g(\pi(x)) = f(x)$ for all $x \in X$ is well-defined: suppose not, and that $\pi(x) = \pi(y)$ and $f(x) \neq f(y)$. Since f is continuous and X is compact, $f(z)$ depends only on finitely many coordinates of z, that is, there exists n such that $f(z) = f(z')$ whenever $z_{\{-n,\dots,n\}} = z_{\{-n,\dots,n\}}$. Since $\pi(x) = \pi(y)$, $u = x_{\{-n,\dots,n\}}$ and $v = y_{\{-n,\dots,n\}}$ are connected by a finite chain $u = w_0 \sim w_1 \sim \cdots \sim w_k = v$ such that w_i and w_{i+1} occur in the same configuration of X for all $0 \leqslant i < k$.

Since the image of f is determined by the restriction to the central $2n+1$ coordinates, we can write $n_i = f(w_i)$ for this unique value. Now, since $f(z) = f(\sigma(z))$ for all $z \in X$, it is easy to see that in fact $n_i = n_{i+1}$ for all i, by moving along orbits of points connecting the words w_i and w_{i+1}. This shows that g is well-defined. Furthermore, as the image of f is determined by the restriction to the central $2n+1$ coordinates, it follows easily that g is continuous. □

We shall now define our analogue of the index map. Let $T \in \mathrm{RFA}(X)$ and let $\phi = \phi_T$ be its permutation model, so that ϕ is an automorphism of $X \times (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}}$. Let $r \in \mathbb{N}$ such that the biradius of ϕ is bounded by r (i.e., the maximum of radii of ϕ and ϕ^{-1} is bounded by r as an automorphism). Let $u \in L(X)$ such that $|u| \geqslant 4r$ and let $(x, y) \in X \times (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}}$ be a configuration where

(1) $x|_{\{0,\dots,|u|-1\}} = u$
(2) $y_m = 1$ if and only if $r \leqslant m < |u| - r$.

Recall that the automorphism ϕ preserves the number $|y^{-1}(1)|$, therefore $\phi_2(x, y)$ also has $|u| - 2r$ heads. We define

$$L_\phi(u) = |\phi(x, y)^{-1}(1) \cap \{0, \dots, 2r-1\}| - r \text{ and } R_\phi(u) = |\phi_2(x, y)^{-1}(1) \cap \{|u| - 2r, \dots, |u| - 1\}| - r.$$

This definition is best explained informally. Count the number of heads on the left side of the coordinate $2r$ (exclusive) of u after applying ϕ and call this $L^*_\phi(u)$. Let $R^*_\phi(u)$ be the number of heads on the right side of coordinate $|u| - 2r$ (inclusive). We clearly have $L^*_\phi(u) + R^*_\phi(u) = 2r$, as ϕ permutes the heads on any configuration and since its biradius is r, coordinates in $[2r, |u| - 2r - 1]$ all contain heads. Then $L_\phi(u) = L^*_\phi(u) - r$ and $R_\phi(u) = R^*_\phi(u) - r$ satisfy that $L_\phi(u) + R_\phi(u) = 0$. For an illustration in the case where ϕ is the permutation model associated to the square of the shift, see Figure 4.

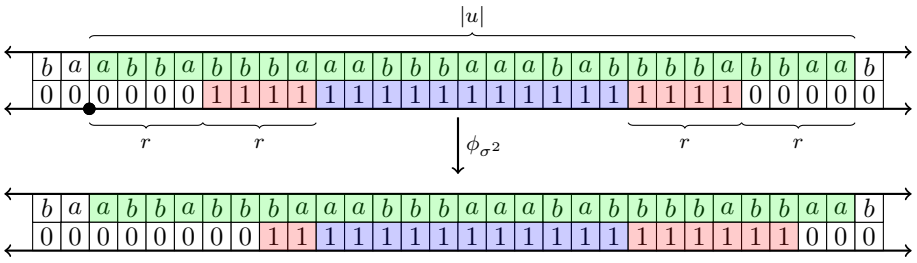


FIGURE 4. For $T = \sigma^2$ its permutation model ϕ_{σ^2} moves the heads to the right twice. If we choose $r = 4$ we observe that the sum of the red zones is $2r$, the blue zone remains unchanged, $L_\phi(u) = -2$ and $R_\phi(u) = 2$.

Definition 3.23. *Let X be a $\mathbb{Z}$-subshift, $T \in \mathrm{RFA}(X)$ and let ϕ be its permutation model. The **head index map** of X and T is the map $H_T \colon X \to \mathbb{Z}$ given by*

$$H_T(x) = \lim_{n\to\infty} L_\phi(x|_{\{0,\dots,n-1\}}).$$

Notice that the function H_T is well defined, as the values of L_ϕ stabilize after $n \geqslant 4r$, where r is the biradius of ϕ. It is also noteworthy that choosing any value r' greater than the biradius r does not change the value. Indeed, as ϕ only permutes the heads in a finite radius, any head which is at the left of $2r'$ at a distance greater than the biradius will necessarily stay left of $2r'$, and thus will not change the value $L_\phi(u) = L^*_\phi(u) - r'$ as the difference $r' - r$ is accounted for on $L^*_\phi(u)$.

Lemma 3.24. *Let X be a $\mathbb{Z}$-subshift. For every $T \in \mathrm{RFA}(X)$ we have that $H_T \in \mathrm{AOS}(X)$ and the map $\gamma \colon \mathrm{RFA}(X) \to \mathrm{AOS}(X)$ defined by $\gamma(\phi_T) = H_T$ is a homomorphism. This homomorphism splits with section $\beta \colon \mathrm{AOS}(X) \to \mathrm{RFA}(X)$ defined by $\beta(f)(x) = \sigma^{f(x)}(x)$, and $\beta(\mathrm{AOS}(X)) = \mathrm{OS}(X)$, so β gives an isomorphism between $\mathrm{AOS}(X)$ and $\mathrm{OS}(X)$.*

Proof. Consider an arbitrary ϕ in the permutation model acting on $X \times (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}}$ with biradius r. We first claim that if $|u| \geqslant 4r$ and $au \sqsubset X$, then $L_\phi(au) = L_\phi(u)$. To see this, note that all heads except the leftmost new one map exactly as before, and thus $R^*_\phi(au) = R^*_\phi(u)$. As $R^*_\phi(v) + L^*_\phi(v) = 2r$ for every $v \in L(X)$ we conclude that $L^*_\phi(au) = L^*_\phi(u)$ and thus $L_\phi(au) = L_\phi(u)$. Symmetrically, one shows that $L_\phi(ua) = L_\phi(u)$.

Suppose now that $u, v \sqsubset x \in X$. Let $w \sqsubset x$ be any word containing both u and v, and apply the observation of the previous paragraph repeatedly to get $L_\phi(u) = L_\phi(w) = L_\phi(v)$. It follows that $H_T(x) = H_T(\sigma(x))$ for all x. The function H_T is continuous because the limit stabilizes after $n > 4r$. We conclude that $H_T \in \mathrm{AOS}(X)$.

To see that $\gamma(T) = H_T$ is a homomorphism, we need the following stronger fact about L_ϕ: fix $m \in \mathbb{N}$ and let $(x, y) \in X \times (\mathbb{Z}/2\mathbb{Z})^{\mathbb{Z}}$ be a configuration where $x_{[0,|u|-1]} = u$ where $|u| \geqslant m + 4r$, and let $y = ...000.0^r v 1^{|u|-|v|-r}000...$ where $v \in (\mathbb{Z}/2\mathbb{Z})^m$ is any word of length m. Then the number of heads in $\phi_2(x, y)|_{\{0,\dots,m+2r-1\}}$ is $|v^{-1}(1)| + r + H_T(x)$. To see this, simply fill in the gaps of v (turn its 0-symbols to 1-symbols). At each step, the number of heads in the image increases by one, and the head is always added among the coordinates in $\{0, \dots, m + 2r - 1\}$ since ϕ has biradius r. After filling the gaps in v, we have $m + r + H_T(x)$ heads in $\phi_2(x, y)|_{\{0,\dots,m+2r-1\}}$ by the definition of L_ϕ.

Now, computation of $\gamma(T \circ T')(x)$ can be done by first applying $\phi' = \phi_{T'}$ on the left side of a long word u in x, with heads positioned suitably on coordinates of u, and then applying $\phi = \phi_T$ to the resulting scattered set of heads. By the previous paragraph, we see that γ is a homomorphism, that is, $\gamma(T \circ T')(x) = \gamma(T)(x) + \gamma(T')(x)$ for all $x \in X$. The map β is well-defined essentially by the definition of $\mathrm{AOS}(X)$, and it is clearly a section for γ. □

As mentioned, the homomorphism γ generalizes the index map defined on elements of the topological full group. It is well-known that the index map is $\mathbb{Z}$-valued in the minimal case. This is true more generally for all weakly chain-transitive subshifts, because for a weakly chain-transitive subshift the maximal invariant symbolic factor is trivial.

Lemma 3.25. *If X is weakly chain-transitive, then its orbitwise shifts are precisely the shifts.*

Proof. Since $\widehat{X}$ is a singleton, Lemma 3.22 implies that $\mathrm{AOS}(X) \cong \mathbb{Z}$ and therefore $\beta \circ \gamma$ must map every element of $\mathrm{RFA}(X)$ to a power of σ by the definition of β. □

A clear example where the previous lemma holds is the full $\mathbb{Z}$-shift $\Sigma^{\mathbb{Z}}$. Given any $T \in \mathrm{RFA}_{\mathrm{fix}}(X)$, we can compose T with orbitwise shifts in order to force the average movement to be zero in every

orbit, that if, to obtain $T \in \mathrm{RFA}_{\mathrm{fix}}(X)$ such that $\gamma(T)$ is identically zero. Therefore, it only remains to find a way to generate all elements of $\mathrm{RFA}_{\mathrm{fix}}(X)$ with have no average movement on any orbit.

Definition 3.26. *Let $u, v \in \Sigma^*, a \in \Sigma$. A reversible finite-state automaton $T = T_{u,a,v} \in \mathrm{RFA}(X)$ is a **controlled position swap** if $T((xu.avy)_1^j) = (xu.avy)_1^{1-j}$ for $j \in \{0, 1\}$, and $T(z) = z$ for all $z \in X \times X_1$ whose image these rules do not determine.*

Lemma 3.27. *Let $u, v \in \Sigma^*, a \in \Sigma$. Then the controlled position swap $T_{u,a,v}$ is well-defined and in $\mathrm{RFA}(X)$ whenever uav is non-unary (i.e., when $uav \notin \{a\}^*$).*

Proof. The cylinders $[u.av]$ and $[ua.v]$ have empty intersection when uav is non-unary. Thus, the conditions under which we move the head do not overlap, and the finite-state machine $T_{u,a,v}$ is well-defined. It is clearly an involution, thus invertible. □

More generally, given a clopen set $C \subseteq X$, we define the map $T_C \in \mathrm{RFA}(X)$, which we call a **clopen-controlled position swap**.

$$T_C(x_1^i) = \begin{cases} x_1^{i+1} & \text{if } \sigma^{-i}(x) \in C \\ x_1^{i-1} & \text{if } \sigma^{-i+1}(x) \in C \\ x & \text{otherwise} \end{cases}$$

As above, these maps are well-defined for any clopen set C such that $C \cap \sigma(C) = \varnothing$. Any such T_C can be obtained as a composition of finitely many controlled position swaps, since clopen sets can be written as a finite union of cylinder sets.

Theorem 3.28. *Let X be a subshift. Then $\mathrm{RFA}(X)$ is generated by orbitwise shifts and controlled position swaps.*

Proof. For $T \in \mathrm{RFA}(X)$, let $P_T \subset \{(0,0)\} \cup \mathbb{Z}_+^2$ be the set of all pairs (m, k) such that there exists $x \in X$ and $i \in \mathbb{Z}$ such that $T(x_1^{i+j}) = x_1^{i+j+m}$ for all $0 \leqslant j < k$. Order these pairs lexicographically (m is more significant). The order type is a suborder of $1 + \omega^2 \cong \omega^2$, thus well-founded. It is now enough to prove that, whenever T has zero average movement,

(1) P_T has a maximal element M_T,
(2) if $M_T > (0,0)$, there exists controlled position swap T' such that $M_{T \circ T'} < M_T$, and
(3) T is the identity map if and only if $M_T = (0,0)$.

Namely, by well-foundedness we must reach $(0,0)$ in finitely many steps by iterating the second item, and by the third we have reached the identity map.

Fix now $T \in \mathrm{RFA}(X)$. To see that a maximum $M_T = (m, k)$ exists, first observe that $P_T \neq \varnothing$ since $(0,0) \in P_T$. Next, for $(m, k) \in P_T$ the local rule of T clearly gives a finite upper bound on m. Finally, if m is maximal and $(m, k) \in P_T$ for arbitrarily large k, then $m > 0$ (since $P_T \subset \{(0,0)\} \cup \mathbb{Z}_+^2$ by choice), and by compactness there exists a configuration $x \in X$ satisfying $T(x_1^i) = x_1^{i+m}$ for all $i \in \mathbb{Z}$, thus average movement on x is clearly nonzero.

Suppose now that $M_T = (m, k) > (0,0)$. Now, let $C \subset X$ be the set of configurations x such that $\forall i \in [-k+1, 0] : T(x_1^i) = x_1^{i+m}$. Observe that $x \in C \implies T(x_1^1) = x_1^{1+j}$ where $j \leqslant m-2$, namely $j \leqslant m-1$ by the maximality of k, and $j = m-1$ would contradict bijectivity of T. Clearly C is a clopen set, and $C \cap \sigma(C) = \varnothing$ because $k > 0$. We let $T' = T_C$.

Consider now an arbitrary point $x \in X$ and $i \in \mathbb{Z}$.

(1) If $\sigma^{-i}(x) \notin C$ and $\sigma^{-i+1}(x) \notin C$, then $(T \circ T')(x_1^i) = T(x_1^i)$.
(2) If $\sigma^{-i}(x) \in C$, then $(T \circ T')(x_1^i) = T(x_1^{i+1})$.
(3) If $\sigma^{-i+1}(x) \in C$, then $(T \circ T')(x_1^i) = T(x_1^{i-1})$.

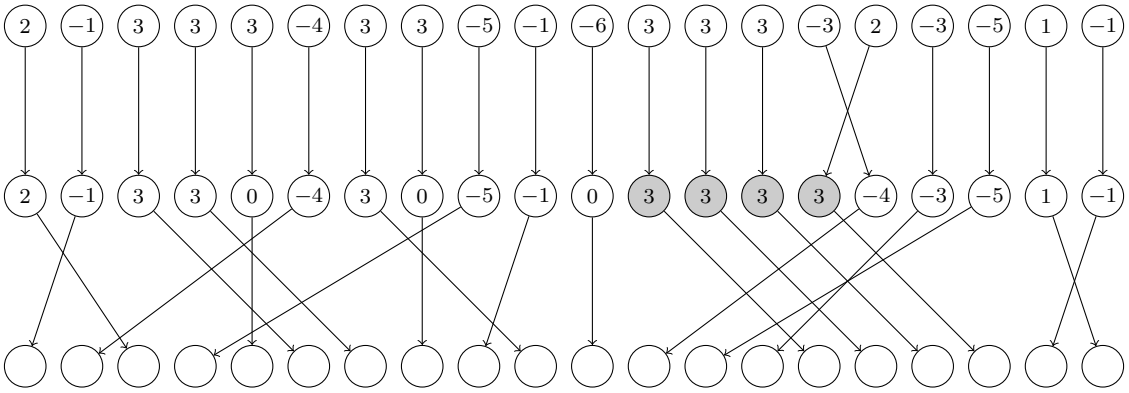


FIGURE 5. The composition $T \circ T'$ illustrated with $m = 3$, $k = 4$. The movement of heads in the permutation model on a configuration x is shown, and values at the nodes are those of y (central row) and z (top row). The four nodes forming m^k are highlighted. The configuration x determines these moves, but its contents are not shown.

Defining $y, z \in \mathbb{Z}^{\mathbb{Z}}$ by $y_i = j$ where $T(x_1^i) = x_1^{i+j}$ and $z_i = j$ where $(T \circ T')(x_1^i) = x_1^{i+j}$, the difference between y to z is precisely that all subsequences of the form $m^k j \in \mathbb{Z}^{k+1}$ are replaced by $m^{k-1}(j+1)(m-1)$. Since $j \leqslant m-2$, the run of ms become strictly shorter, and no symbols larger than m are introduced, thus $M_{T \circ T'} < M_T$ are required. See Figure 5 for an illustration. □

Remark 3.29. In the proof above, only the part about non-zero average movement and orbitwise shifts crucially depends on the properties of X. The other part does not: if $T \in \mathrm{RFA}(Y)$, T has average movement zero (on Y), and $X \subset Y$, then one can perform all choices in the proof canonically so that decomposing T into controlled position swaps on Y and then restricting to X is equivalent to first restricting to X and then applying our construction on X.

Let us use Lemma 2.8 to extract corollaries in the case where we have more than one state. If $k \geqslant 2$, then recall that the isomorphism between $\mathrm{RTM}(X, k)$ and $\mathrm{RTM}(\sqrt[k]{X}, 1)$ simply uses the different positions on #-segments to encode the state. Thus, moving a head one step to the right along a contiguous segment of #-symbols corresponds to increasing the state by one. Moving a head from the rightmost #-symbol to non-# symbol to the right of it means corresponds to changing the state to 1 and stepping to the right.

Thus, translating the previous result amounts to the following: Let $X \subset \Sigma^{\mathbb{Z}}$ be a locally aperiodic subshift and $k \geqslant 2$. For $u, v \in \Sigma^*$ and $q \in \{1, \dots, k-1\}$ define the **controlled state swap** $f_{u,q,v} \in \mathrm{RFA}_{\mathrm{fix}}(X, k)$ as

$$f_{u,q,v}(x, q') = (x, q'')$$

where $q'' = q'$ if $q' \notin \{q, q+1\}$ or if $x_{\{-|u|,\dots,|v|-1\}} \neq uv$, and otherwise let $q'' \in \{q, q+1\} \backslash \{q'\}$. For $u, v \in \Sigma^*, a \in \Sigma$, define a **stateful controlled position swap** as $f(xu.avy, k) = (xua.vy, 1)$ and $f(xua.vy, 1) = (xy.avy, k)$ for all tails x, y, and $f(z) = z$ for all points not of this form. This is a well-defined involution for all u, v (since $k \geqslant 2$).

Corollary 3.30. *Let X be a weakly chain-transitive locally aperiodic subshift and $k \geqslant 2$. Then $\mathrm{RFA}(X, k)$ is generated by the shift map, controlled state swaps, and stateful controlled position swaps.*

3.4. **Elementary Turing machines are finitely generated.** In this section, we prove Theorem 1.1. That is, we show that for any $n, k \geqslant 1$ the group $\mathrm{EL}(\mathbb{Z}, n, k)$ is finitely-generated.

Proof of Theorem 1.1. The case $n = 1$ is easy. Suppose thus that $n \geqslant 2$. Let $X = \Sigma^{\mathbb{Z}}$. Since $\mathrm{OB}(X)$ is finitely generated and $\mathrm{OS}(X) = \langle\sigma\rangle$, we only need to show that the controlled position swaps can be implemented when $k = 1$, and that the stateful controlled position swaps and controlled state swaps can be implemented when $k \geqslant 2$. We work in the fixed-head model.

Suppose first that $k = 1$. Let $s_\ell = T_{\epsilon,1,0^\ell}$. By conjugating with an element of $\mathrm{OB}(X)$, these swaps generate all controlled swaps where the left control is ϵ. Namely, to generate $T_{\epsilon,a,v}$ where av is not unary (see Lemma 3.27), conjugate s_ℓ where $\ell = |v|$ with any local permutation mapping

$$f(x.avy) = x.10^\ell y \text{ and } f(xa.vy) = x1.0^\ell y.$$

Such f exists because av is not unary and thus does not overlap with its 1-shift – simply map these patterns correctly, and fill the permutation arbitrarily. Swaps at other positions are obtained by conjugating with σ. Thus, we only need to show that s_ℓ can be decomposed into elements of $\mathrm{OB}(X)$, shifts, and finitely many controlled swaps. We do this by induction on ℓ.

We begin with a construction in the case when n is even. We begin with the intuitive description. Since whether the head is on the coordinate 0 or 1 in the word 10^ℓ can be checked locally, we can consider the position to be just a binary digit c (so $c = 0$ means we are on the zero-coordinate of a word 10, $c = 1$ means we are on the one-coordinate of such pattern, and otherwise we fix the point anyway). Note that using elements of $\mathrm{OB}(X)$, we can swap positions to the right of the pattern 10 independently of c. Thus, our controlled swaps may depend on positions arbitrarily far to the right from the central 10, as long as they do not depend on more than a constant number of coordinates.

We begin with state cwb where c is the head position (either 0 or 1, marking whether we are on the first or second coordinate of w), and $w = 10u$ where $u \in \Sigma^{\ell-1}$ is the control word that we are checking for value $0^{\ell-1}$. The permutation is the following: First, flip c if $b = 0$ (by moving b next to the initial 10 of w performing a small-radius controlled swap, and moving b back). Then add 1 (modulo n) to b (by an element of $\mathrm{OB}(X)$) if $u = 0^{\ell-1}$. Then repeat these two steps n times. The end result is that c is flipped once if $u = 0^{\ell-1}$, since $b = 0$ at exactly one of the c-flipping steps, and otherwise c is flipped either 0 or n times and thus is not changed.

More precisely, define first $g \in \mathrm{RFA}(X)$ as the controlled swap $T_{\epsilon,1,00}$, that is,

$$g(x1.00y) = x.100y, \quad g(x.100y) = x1.00y$$

and $g(z) = z$ for points not of this form. This is clearly in $\mathrm{RFA}(X)$, and we take it in our set of generators. Also define $h \in \mathrm{OB}(X)$ as any local permutation such that

$$h(x1.0uby) = x1.0buy, \quad h(x.10uby) = x.10buy$$

for all $u \in \Sigma^{\ell-1}, b \in \Sigma$ (and $h(z)$ may be arbitrary for points not of this form), which is again well-defined since no point is of both forms.

Now, defining $f' = h^{-1} \circ g \circ h$ we have

$$f'(x1.0u0y) = x.10u0y, \quad f'(x.10u0y) = x1.0u0y$$

for all $u \in \Sigma^{\ell-1}$ and $f'(z) = z$ for points not of this form.

Next, define $f'' \in \mathrm{OB}(X)$ by

$$f''(x.10^\ell by) = x.10^\ell(b+1)y, \quad f''(x1.0^\ell by) = x1.0^\ell(b+1)y$$

for all $b \in \{0, 1, \ldots, n-1\}$ (where increment is modulo n) and $f''(z) = z$ for all $z \in X$ not of this form.

Then $(f'' \circ f')^n$ is our desired map.

Suppose then that n is odd. We will use the proof structure of [39]. Again, consider the input to be $c10ub$ where $c \in \{0,1\}$ indicates the position of the head (either on the symbol 1 or the symbol 0 of the word 10 after c), and $u \in \Sigma^{\ell-1}$ and $b \in \Sigma$. We construct the machine by induction. Suppose we have already constructed a machine that flips c conditioned on $u = 0^{\ell-1}$, and let us show how to check $b = 0$. First flip b between 0 and 1 if $u = 0^{\ell-1}$ (and fix the value of b if $b \in \Sigma\backslash\{0,1\}$), using a machine in $\mathrm{OB}(X)$. Then flip the bit c if $b = 0$ using a machine in $\mathrm{RFA}(X)$ conjugated by a machine in $\mathrm{OB}(X)$, as above. Repeat these steps. The resulting machine g^2 is in $\mathrm{EL}(X)$ and will flip c if and only if $b \in \{0,1\}$ and $u \in 0^{\ell-1}$.

Now, conjugate the machine by affine translations: let $h \in \mathrm{OB}(X)$ add 2 to the value of b. Then $h^{-\frac{n-1}{2}} \circ (g^2 \circ h)^{\frac{n-1}{2}}$ will flip the value of c if and only if $u = 0^{\ell-1}$ and $b \neq 0$. Finally, flip c if $u = 0^{\ell-1}$, so that the resulting machine flips c if and only if $ub = 0^{\ell}$.

The details of translation to machines in $\mathrm{RFA}(X)$ and $\mathrm{OB}(X)$ are omitted, as they are similar as in the case of an even alphabet.

In the case $k \geqslant 2$, we need to perform controlled state swaps and stateful controlled position swaps using finitely many elements of $\mathrm{RFA}(X)$ and elements of $\mathrm{OB}(X)$. The proof of controlled state swaps is the same as the proof above, but now the bit c (which indicated the position above) indicates which state we are in, out of the two we are swapping (and nothing happens if we are in neither state). The proof of stateful controlled position swaps is also similar; and now the bit c is 0 if we are in state k, and 1 if we are in state k, and in state 1 the control word is seen through an offset of one. □

4. Computability aspects

In this section we study the computability aspects of $\mathrm{RTM}(\mathbb{Z}^d, n, k)$. We begin the section by briefly showing which properties of Turing machines are computable. In particular we prove that injectivity and thus the reversibility of a machine in $\mathrm{TM}(\mathbb{Z}^d, n, k)$ is decidable. This property, along with the possibility to compute the rule of a composition and an inverse gives a recursive presentation for $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ which has a decidable word problem.

In this context, we proceed to study the torsion problem of $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ and its subgroups, that is, whether there exists an algorithm which given a description of $T \in \mathrm{RTM}(\mathbb{Z}^d, n, k)$ always halts and accepts if and only if there exists $n \geqslant 1$ such that $T^n = \mathrm{id}$. In this context we show that $\mathrm{EL}(\mathbb{Z}, n, k)$ has an undecidable torsion problem (Theorem 1.2). Furthermore, we use this result to show that the automorphism group of any uncountable $\mathbb{Z}$-subshift contains a finitely generated subgroup with undecidable torsion problem (Corollary 1.3).

Finally, we study the torsion problem for $\mathrm{RFA}(\mathbb{Z}^d, n, k)$. We show by a simple argument that $\mathrm{RFA}(\mathbb{Z}, n, k)$ has a decidable torsion problem. Interestingly, the torsion problem in $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ for $d \geqslant 2$ is undecidable. We present a detailed proof of this result which draws upon the undecidability of the snake tiling problem [24]. These two results add up to the Theorem 1.4 we discussed in the introduction.

4.1. Basic decidability results.

First, we observe that basic management of local rules is decidable.

Lemma 4.1. *Given two local rules f, g in the moving head model,*

- *It is decidable whether $T_f = T_g$,*
- *We can effectively compute a local rule for $T_f \circ T_g$,*
- *It is decidable whether T_f is injective,*
- *It is decidable whether T_f is reversible, and*
- *We can effectively compute a local rule for T_f^{-1} when T_f is reversible.*

Proof. For the first claim, let $f\colon \Sigma^{F_1} \times Q \to \Sigma^{F_2} \times Q \times \mathbb{Z}^d$ and $g\colon \Sigma^{F_3} \times Q \to \Sigma^{F_4} \times Q \times \mathbb{Z}^d$ and define $F = F_1 \cup F_2 \cup F_3 \cup F_4$. Extend the rules f, g so that they are defined as $f', g'\colon \Sigma^F \times Q \to \Sigma^F \times Q \times \mathbb{Z}^d$ satisfying $T_f = T_{f'}$ and $T_g = T_{g'}$. If for some $(p, q) \in \Sigma^F \times Q$ $f'(p, q) \neq g'(p, q)$, then clearly $T_{f'} \neq T_{g'}$. Otherwise, we have $T_{f'} = T_{g'}$.

Finding a local rule for the composition of two Turing machines is a straightforward, if somewhat tedious, exercise.

For the decidability of reversibility, we give a semi-algorithm for both directions. First, if T_f is reversible, then it has a reverse T_g. We thus only need to enumerate local rules g, and check whether $T_f \circ T_g = \mathrm{id}$, which is decidable by using the two previously described procedures.

If T_f is not injective, then $T_f(x, y) = T_f(x', y')$ for some $(x, y), (x', y') \in \Sigma^{\mathbb{Z}^d} \times X_k$ with $y_{\vec{0}} \neq 0$. If r is the move-radius of f, then necessarily the nonzero position of y' is at distance at most r from the origin and $x_{\vec{v}} = x'_{\vec{v}}$ for $|\vec{v}|$ larger than the radius of T_f. Then we can assume $x_{\vec{v}} = x'_{\vec{v}} = 0$ for all such $\vec{v}$. It follows that if T_f is not injective, it is not injective on the finite set of configurations $(x, y) \in \Sigma^{\mathbb{Z}^d} \times X_k$ where $(x, y)_{\vec{v}} = 0$ for all $|\vec{v}|$ larger than the radius of T_f, which we can check algorithmically.

By Theorem 2.19, injectivity is equal to reversibility, so the decidability of the reversibility of a Turing machine is a direct consequence of the previous argument.

Finally, if T_f is reversible, we can effectively construct a reverse ${T_f}^{-1}$ for it by enumerating all Turing machines and outputting the first T' such that $T_f \circ T' = T' \circ T_f = \mathrm{id}$. □

From the results above, we obtain that the set of all possible local rules f which generate reversible actions T_f gives a recursive countable presentation of $\mathrm{RTM}(\mathbb{Z}^d, n, k)$. Furthermore, that presentation has a decidable word problem.

Proposition 4.2. $\mathrm{RTM}(\mathbb{Z}^d, n, k)$ *admits a recursive presentation with decidable word problem.*

Proof. The set of local rules giving moving head Turing machines can be recursively enumerated. Indeed, consider the sequence of sets $\Lambda_n = \{-n, \dots, n\}^d$ indexed by $n \in \mathbb{N}$ and list in some lexicographical order all local rules $f\colon \Sigma^{\Lambda_n} \times Q \to \Sigma^{\Lambda_n} \times Q \times \Lambda_n$. For each $n \in \mathbb{N}$ this is a finite set and thus all local rules can be recursively listed as $(f_i)_{i\in\mathbb{N}}$. By Lemma 4.1 it is decidable which local rules define reversible Turing machines, and thus one can run that algorithm on each f_i to obtain a recursive enumeration $(f_{\varphi(i)})_{i\in\mathbb{N}}$ of all reversible moving head Turing machines. Also, using Lemma 4.1 one can reduce every word $f_{\varphi(i_1)} f_{\varphi(i_2)} \cdots f_{\varphi(i_k)}$ to some equivalent rule f_{eq}. It suffices to test the equality of f_{eq} with the identity machine to decide the word problem of this presentation. □

4.2. **The torsion problem of elementary Turing machines.**

Definition 4.3. *Let G be a group which is generated by $S \subset G$. The* ***torsion problem*** *of G is the set of words $w \in S^*$ for which there is $n \in \mathbb{Z}_+$ such that the element represented by w^n is the identity of G.*

If a group G is recursively presented, then the torsion problem is recursively enumerable. However, the torsion problem may not be decidable even when G has decidable word problem. Many such examples are known, and the main result of this section provides a new such example.

As discussed in the introduction, we say a moving head Turing machine is **classical** if its in- and out-radii are 0, and its move-radius is 1. Here we characterize reversibility in classical Turing machines. If T_0 has in-, out- and move-radius 0, that is, T_0 only performs a permutation of the set of pairs $(s, q) \in \Sigma \times Q$ at the position of the head, then we say $T_0 \in \mathrm{LP}(\mathbb{Z}, n, k)$ is a **state-symbol permutation**. If T_1 has in-radius -1, never modifies the tape, and only makes movements by vectors in $\{-1, 0, 1\}$, then $T_1 \in \mathrm{RFA}(\mathbb{Z}, n, k)$ is called a **state-dependent shift**.

If we consider the class of all classical Turing machines on some finite alphabet and number of states then the torsion problem is undecidable. This result was shown by Kari and Ollinger in [25] –they

call it the periodicity problem in their setting– using a reduction from the mortality problem which in turn they also prove to be undecidable following a reduction from the mortality problem of reversible 2-counter machines. In this section, we show that the torsion problem is also undecidable for elementary Turing machines with a fixed alphabet and number of states. We do this by reducing to the classical machines. We begin by describing them in our setting.

Proposition 4.4. *A classical Turing machine $T \in \mathrm{RTM}(\mathbb{Z}, n, k)$ is reversible if and only if it can be expressed in the form $T_1 \circ T_0$ where T_0 is a state-symbol permutation and T_1 is a state-dependent shift. In particular, classical reversible Turing machines are in $\mathrm{EL}(\mathbb{Z}, n, k)$.*

Proof. We only need to show that if T is reversible then it is of the stated form. Let $f_T \colon \Sigma \times Q \to \Sigma \times Q \times \{-1, 0, 1\}$ be a local rule for T in the moving tape model. We claim that if $f_T(a, q) = (b, r, d)$ and $f_T(a', q') = (b', r, d')$ then $d = d'$. Namely, if it is not the case, one can easily find two configurations with the same image. There are multiple cases to consider which can all be treated similarly, for instance, let x, y be a left and right infinite sequence of symbols of Σ respectively and suppose $d = 0$ and $d' = 1$. In this case:

$$T((xb'.ay), q) = (xb'.by, r) = T((x.a'by), q').$$

Therefore contradicting the reversibility of T. Repeating this argument over all pairs $d \neq d'$ we obtain that the direction of movement is entirely determined by the output state. Of course, for T to be injective, also f_T must be injective, so the map $g \colon \Sigma \times Q \to \Sigma \times Q$ defined by $g(a, q) = (b, r)$ if $f_T(a, q) = (b, r, d)$ is injective, thus a bijection. From here, the only remaining possibility is that T_0 is the permutation g, and T_1 as the finite-state automaton with local rule $f_{T_1}(a, q) = (a, q, d)$ if $f_T(b, q') = (b', q, d)$ for some $(b, q) \in \Sigma \times Q$. □

It follows that the inverse of a reversible classical Turing machine is always of the form $T_0 \circ T_1$ where T_0 is a state-symbol permutation and T_1 is a state-dependent shift. In the terminology of Section 3, the theorem implies that all reversible classical Turing machines are elementary.

Now we are ready to prove Theorem 1.2, namely, that the torsion problem of $\mathrm{EL}(\mathbb{Z}^d, n, k)$ is undecidable for all $n \geqslant 2$, $k \geqslant 1$ and $d \geqslant 1$.

Proof of Theorem 1.2. As $\mathrm{EL}(\mathbb{Z}, n, 1)$ embeds into $\mathrm{EL}(\mathbb{Z}^d, n, k)$ for every $k, d \geqslant 1$, It is enough to prove this for $d = 1$ and $k = 1$.

The machines constructed in [25] are already elementary, but their alphabets are not bounded (note that these are classical Turing machines so there are only finitely many machines of any given state-alphabet pair). However, one can simulate them with elementary Turing machines with fixed alphabets. That is, given a classical reversible Turing machine T (on some finite alphabet and number of states), one can construct a machine in $\mathrm{EL}(\mathbb{Z}, n, 1)$ which is periodic if and only if T is. Our result then follows from the main result of [25].

Let T be a given classical Turing machine with state set Q and tape alphabet Σ, and let f_T be its local rule. Note that in this proof we use Σ and Q for the alphabet and state set of T, rather than those of the machines in $\mathrm{EL}(\mathbb{Z}, n, 1)$.

By Proposition 4.4 we have $T = T_1 \circ T_0$ where T_0 is a state-symbol permutation and T_1 is a state-dependent shift. We can further assume that T_1 does not use movement by 0: For each state q that would move by 0 we introduce a new state q', and modify shift T_1 to move by -1 in state q and by $+1$ in state q'. We also modify state-symbol permutation T_0 to enter always state q' instead, whenever it would enter q, and to map $(a, q') \mapsto (a, q)$ for all tape symbols a. In this way a single non-moving step gets replaced by two steps moving to the right and to the left, respectively. Clearly the modified

machine is periodic if and only if the original one is. Let us denote by $L \subset Q$ and $R \subset Q$ the sets of left and right moving states in T_1. Then Q is the disjoint union of L and R

Let $\perp$ be a symbol which is not in Q and m be such that $n^m \geqslant |\Sigma \times (Q \cup \{\perp\})|$ so that we can encode tape symbols (represented by pairs in $\Sigma \times \{\perp\}$) and state-symbol pairs as unique blocks of length m over the n letter alphabet. Pair (a, q) represents a tape cell containing $a \in \Sigma$ and being read by the machine in state $q \in Q$, while $(a, \perp)$ is a cell containing $a \in \Sigma$ and not currently read by the machine. Take an arbitrary one-to-one encoding function $\Sigma \times (Q \cup \{\perp\}) \to \{1, \ldots, n\}^m$. Let us denote by $[a, r]$ the encoding of $(a, r) \in (Q \cup \{\perp\})$, a word of length m in the alphabet $\{1, \ldots, n\}$.

The idea is that a configuration of T with tape content $x \in \Sigma^{\mathbb{Z}}$, state $q \in Q$ and position $p \in \mathbb{Z}$ is represented as the encoded configuration with tape content $x' \in \{1 \ldots, n\}^{\mathbb{Z}}$, where $x'_{[im,(i+1)m)} = [x_i, \perp]$ for $i \neq p$ and $x'_{[pm,(p+1)m)} = [x_p, q]$, and with the stateless head in position pm. Notice that there might be many configurations in $\{1, \ldots, n\}^{\mathbb{Z}}$ that are not valid encodings.

We construct an element of $\mathrm{EL}(\mathbb{Z}, n, 1)$ that is a composition $T'_1 \circ T'_0$ where $T'_0 \in \mathrm{LP}(\mathbb{Z}, n, 1)$ and $T'_1 \in \mathrm{RFA}(\mathbb{Z}, n, 1)$ are both involutions. Informally, the local permutation T'_0 implements the state-symbol permutation T_0 on the encoded tape, including writing the new state in the correct position, while the finite-state automaton T'_1 simply moves the machine head m positions to the right or to the left to scan the block containing the new state. We complete the local rules of T'_0 and T'_1 into involutions that also perform the inverse steps, and act as identities in all other situations. This automatically causes the machine to reverse its time direction and start retracing it steps backwards if it encounters a non-valid pattern.

More precisely, the local permutation T'_0 is defined with neighborhood $F_0 = \{-m, \ldots, 2m-1\}$ and so that the local rule f_0 is a permutation of $\{1, \ldots, n\}^{F_0}$. The machine sees three m-blocks: the current block in the center and neighboring blocks on both sides. For every $a, b, c, d \in \Sigma$ and $q, p \in Q$,

(1) if $f_T(a, q) = (b, p, -1)$ then f_0 swaps

$$[c, \perp][a, q][d, \perp] \longleftrightarrow [c, p][b, \perp][d, \perp],$$

(2) if $f_T(a, q) = (b, p, +1)$ then f_0 swaps

$$[c, \perp][a, q][d, \perp] \longleftrightarrow [c, \perp][b, \perp][d, p].$$

All other patterns in $\{1 \ldots, n\}^{F_0}$ are mapped to themselves. By the reversibility of T this f_0 is a well-defined involution.

The reversible single-state finite automaton T'_1 uses the neighborhood $F_1 = \{-2m, \ldots, 3m-1\}$, so the machine sees two blocks to the left and to the right from its current block. Its local rule is a function $f_1 \colon \{1, \ldots, n\}^{F_1} \longrightarrow \mathbb{Z}$ where $f_1(u)$ gives the head movement on tape pattern u.

(1) Let $p \in L$. Then f_1 maps, for all $b, c, d \in \Sigma$,

$$\begin{array}{ccccccc} * & [c,p] & [b,\perp] & [d,\perp] & * & \mapsto & -m, \\ * & * & [c,p] & [b,\perp] & [d,\perp] & \mapsto & +m, \end{array}$$

where $*$ represents any m-block in $\{1 \ldots, n\}^m$. In other words, if the machine sees a state $p \in L$ on the block to its left and non-states $\perp$ at its current block and the block on its right, the machine moves left. If it sees a state $p \in L$ at its current block and $\perp$ on the two blocks to its right, it moves right. These moves are inverses of each other: if one is applicable now, the other one is applicable on the next time step and it makes the machine return to its original position.

(2) Symmetrically, let $p \in R$. Then f_1 maps, for all $c, b, d \in \Sigma$,

$$\begin{array}{ccccccc} * & [c,\perp] & [b,\perp] & [d,p] & * & \mapsto & +m, \\ [c,\perp] & [b,\perp] & [d,p] & * & * & \mapsto & -m. \end{array}$$

Also these moves are inverses of each other.

On all other patterns in $\{1\dots,n\}^{F_1}$ the movement is by 0. Because L and R are disjoint, T_1' is well-defined. It is also an involution.

Machines T_0' and T_1' are so designed that $T_1' \circ T_0'$ simulates one step of T on valid encodings of its configurations. It follows that if T is not periodic neither is $T_1' \circ T_0'$.

Suppose then that $T_1' \circ T_0'$ is not periodic. When iterating T_0' and T_1' alternatingly, if at any moment either machine does not change the configuration then the iteration starts to retrace its steps back in time. This follows from the fact that T_0' and T_1' are involutions. Any orbit containing two such changes of time direction is periodic. By this and compactness, if $T_1' \circ T_0'$ is not periodic, there exists a bi-infinite orbit where each application of T_0' and T_1' changes the configuration. By the construction, the machine locally only sees valid encodings. The orbit – in either forward or backward time direction – is then a valid simulation of T on the portion of the tape that T sees. It follows that T has a non-periodic orbit as well. □

Combining Theorem 1.2 and Theorem 1.1, we immediately obtain the following corollary.

Corollary 4.5. *There is a finitely generated subgroup of* $\mathrm{RTM}(\mathbb{Z}, n, k)$ *whose torsion problem is undecidable.*

4.3. **The torsion problem of cellular automata.** In this section, we show that there is a finitely generated group of cellular automata whose torsion problem is undecidable. It is tempting to try to prove this by constructing an embedding from $\mathrm{EL}(\mathbb{Z}, n, k)$ to $\mathrm{Aut}(A^{\mathbb{Z}})$, but this is impossible: it is well-known that $\mathrm{Aut}(A^{\mathbb{Z}})$ is residually finite, however, the group of finitely supported permutations of $\mathbb{N}$ embeds into $\mathrm{LP}(\mathbb{Z}, n, k)$ and therefore $\mathrm{EL}(\mathbb{Z}, n, k)$ can not be residually finite. Nevertheless, there are standard ways to construct cellular automata from Turing machines, and we show that while one cannot preserve the complete group structure, one can preserve torsion.

One could do this similarly as in the previous section, by showing directly that an automorphism simulating a Turing machine will be periodic if and only if the Turing machine is. We will prove a slightly stronger abstract result, namely we construct a map from $\mathrm{EL}(\mathbb{Z}, n, k)$ to $\mathrm{Aut}(A^{\mathbb{Z}})$ that preserves some of the group structure of $\mathrm{EL}(\mathbb{Z}, n, k)$ while adding only some "local" identities. To achieve this, it seems we cannot use the same construction as in the previous section, as discussed below in Remark 4.9.

In the following, the free monoid generated by the elements of a group G is written G^*, and consists of formal words w where $w_i \in G$ for all $i = 1, \dots, |w|$. For $w \in G^*$, write $\underline{w} = w_1 \cdot w_2 \cdots w_{|w|} \in G$ for the corresponding element of G.

Definition 4.6. *Let G and H be groups and $\mathcal{P}$ be a group property. We say a function $\phi\colon G \to H$ is $\mathcal{P}$-preserving if the following holds: For every finite set $F \subset G^*$ the group $\langle \underline{w} \mid w \in F \rangle \leqslant G$ has property $\mathcal{P}$ if and only if the group $\langle \underline{\phi(w_1)\phi(w_2)\cdots\phi(w_{|w|})} \mid w \in F \rangle$ has property $\mathcal{P}$*

We remark that $\mathcal{P}$-preserving functions ϕ need not be morphisms as we do not ask that $\phi(w_1 w_2) = \phi(w_1)\phi(w_2)$. We only demand that property $\mathcal{P}$ is preserved when applying ϕ to the symbols appearing in the words in F.

In what follows we are going to use the property $\mathcal{P}$ of being finite. We use this property to extend computability invariants such as the torsion problem of a group onto another group even if no embedding from the first group to the second exists. This kind of extension obviously demands that the function ϕ is in some way computable. We will say a function $\phi\colon G \to H$ is **computable** if both G and H have decidable word problem for some fixed presentation and there is a Turing machine which turns any word w in the presentation of G such that $\underline{w} = g \in G$ into a word u in the presentation of H such that $\underline{u} = \phi(g) \in H$.

Lemma 4.7. *Let G be a finitely generated group with undecidable torsion problem and generating set S, and suppose there exists a computable finiteness-preserving function $\phi\colon G \to H$. Then the subgroup $H' = \langle \{\phi(s)\}_{s\in S}\rangle \leqslant H$ has undecidable torsion problem.*

Proof. Suppose the group H' generated by the $\phi(s)$ has decidable torsion problem and let $w \in S^*$. As ϕ is finiteness-preserving we have that $\langle \bar{w}\rangle$ is finite if and only if $\langle \phi(w_1)\phi(w_2)\cdots\phi(w_{|w|})\rangle$ is finite. This means $\underline{w}$ has finite order in G if and only if $\phi(w_1)\phi(w_2)\cdots\phi(w_{|w|})$ has finite order in H'. We can compute $\phi(w_1)\phi(w_2)\cdots\phi(w_{|w|})$ from w and from the finite set $\{\phi(s)|s \in S\} \subset H$. Thus, the algorithm to decide the torsion problem in H' can be used to decide the torsion problem in G, raising a contradiction. □

The previous lemma indicates that in order to prove that $\mathrm{Aut}(A^{\mathbb{Z}})$ contains a finitely generated subgroup with undecidable torsion problem, it now suffices to provide a computable finiteness-preserving map from $\mathrm{EL}(\mathbb{Z}, n, k)$ into $\mathrm{Aut}(A^{\mathbb{Z}})$.

Lemma 4.8. *Let $A = \Sigma^2 \times (\{\leftarrow, \rightarrow\} \cup Q \times \{\uparrow, \downarrow\})$. There is a computable finiteness-preserving map $\phi\colon \mathrm{RTM}(\mathbb{Z}, n, k) \to \mathrm{Aut}(A^{\mathbb{Z}})$.*

Proof. The alphabet A consists of triples and thus $A^{\mathbb{Z}}$ can be thought of as consisting of three tapes. The two first tapes carry a configuration in $\Sigma^{\mathbb{Z}}$ while the third tape has symbols in $(\{\leftarrow, \rightarrow\} \cup (Q \times \{\uparrow, \downarrow\}))$ and deals with the heads and calculation zones. A head is represented by a tuple in $Q \times \{\uparrow, \downarrow\}$, where Q is the set of states of the Turing machine and $\{\uparrow, \downarrow\}$ is the track the machine is in. If this value is $\uparrow$, the head is on the first ("topmost") tape and if it is $\downarrow$, on the second ("bottom") tape. $\leftarrow$ means the head is to the left of the current cell on the current zone (if the current zone contains a head), while $\rightarrow$ means the head is to the right.

A configuration in $A^{\mathbb{Z}}$ is split into zones by the contents of the third tape. Namely, every finite portion of the second track can be split uniquely into pieces of the forms $\rightarrow^*$ (q, a) $\leftarrow^*$ and $\rightarrow^*\leftarrow^*$ where $(q, a) \in Q \times \{\uparrow, \downarrow\}$. We call these pieces **zones**, see Figure 6. To define the finiteness-preserving map $\phi\colon \mathrm{RTM}(\mathbb{Z}, n, k) \to A^{\mathbb{Z}}$ it is enough to do so in every piece of this form.

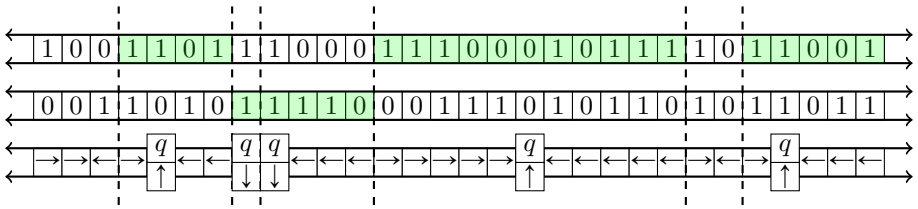


FIGURE 6. A finite word in A^* is divided into zones by the third layer. The dashed lines separate each zone and the colours indicate which tape is being pointed at by the arrow next to the state.

Let $T \in \mathrm{RTM}(\mathbb{Z}, n, k)$ be a moving head Turing machine of radius r. We define $\phi(T) \in \mathrm{Aut}(A^{\mathbb{Z}})$ by defining its action over each zone as follows: If the zone has no head or the zone is of size less than $2r + 1$, do nothing. Otherwise let $u_0, \ldots, u_{m-1}$ and $v_0, \ldots, v_{m-1}$ be the words in the first and second track respectively, $(q, a) \in Q \times \{\uparrow, \downarrow\}$ be the head and $\ell \in \{0, \ldots, m-1\}$ the position of the head in the third track. Using this information we can construct the configuration $x \in \Sigma^{\mathbb{Z}}$ given by:

$$x_i = \begin{cases} u_j & \text{if } j = (i \mod 2m) \in \{0, \ldots, m-1\} \\ v_{2m-j-1} & \text{if } j = (i \mod 2m) \in \{m, \ldots, 2m-1\} \end{cases}$$

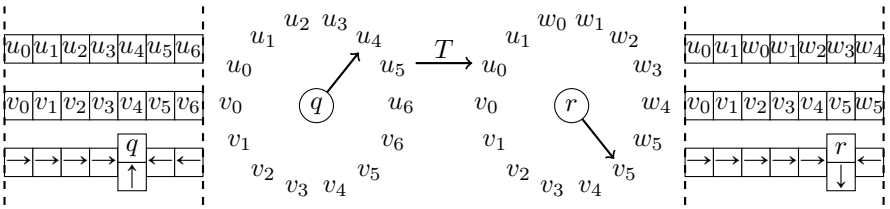


FIGURE 7. Every zone is wrapped around as a conveyor belt, where $\phi(T)$ acts as if it were T seeing a periodic word.

And apply T to x, where the position of the head is on ℓ if $a = \uparrow$ and on $2m - \ell - 1$ otherwise. After applying the Turing machine's local rule, recode the result again updating the left and right arrows so that the zone does not change its shape as shown in Figure 7. With this definition, in each bounded zone $\phi(T)$ induces a permutation of all possible heads and tape contents while on unbounded zones it acts as T on an infinite configuration.

The map $\phi(T)$ is clearly continuous and shift invariant and therefore $\phi(T) \in \mathrm{Aut}(A^{\mathbb{Z}})$. Also, ϕ is clearly a computable map. It suffices to show that ϕ preserves the property of being finite.

Consider $F \subset \mathrm{RTM}(\mathbb{Z}, n, k)^*$, and $H = \langle \phi(w_1)\phi(w_2)\cdots\phi(w_{|w|}) \mid w \in F \rangle$ generated in $\mathrm{Aut}(A^{\mathbb{Z}})$. If $\langle F \rangle$ is infinite, just note that the action of $\phi(T)$ over any configuration such that the third tape is a single unbounded zone with a head, (that is $\ldots \leftarrow\leftarrow\leftarrow (q, \uparrow) \rightarrow\rightarrow\rightarrow \ldots$) replicates exactly the behavior of T on the first tape. Therefore each element of $\langle F \rangle$ will act differently over at least one configuration of this form, implying that H is infinite. Conversely, if $\langle F \rangle$ is finite, consider the action over any zone which is unbounded or larger than the maximum movement from the origin attained by a machine in $\langle F \rangle$. Obviously such a machine acts as the original machine, so the action on such zones satisfies the same relations as the action of $\langle F \rangle$. Consider then the zones whose length is bounded by a fixed length h. The number of different machine actions on these zones is bounded as each action is a permutation over a finite set, thus again the movement of the Turing machine head is bounded. Therefore H is finite and a rough bound is $|H| \leqslant |\langle F \rangle| \cdot (\prod_{m \leqslant h} (2kmn^{2m})!)$ □

Remark 4.9. The above is not the only possible construction for making cellular automata out of Turing machines. For example in [25], one instead uses a direction bit, and flips the running direction of the Turing machine (from forward to backward) if it hits the boundary of a computation zone. This roughly corresponds to the construction in the previous section. This construction does not, at least without some modifications, give a finiteness-preserving map in the sense of the definition above: Suppose f and g are Turing machines satisfying no relations. Pick $F = \{(f^{-1}g^{-1}) \cdot g \cdot f\}$ (a formal product of length 3 in G^*) in Definition 4.6. Since $\mathrm{id} = (f^{-1}g^{-1}) \cdot g \cdot f$ is torsion as a Turing machine, we should have that the corresponding cellular automaton $\phi(f^{-1}g^{-1}) \circ \phi(g) \circ \phi(f)$ is. But on a configuration where the direction bit points backwards in time, this cellular automata in fact simulates the Turing machine $g \circ f \circ g^{-1} \circ f^{-1}$ which is of infinite order.

Using that $\mathrm{EL}(\mathbb{Z}^d, n, k)$ is finitely generated and considering the restriction of the map of Lemma 4.7 to $\mathrm{EL}(\mathbb{Z}^d, n, k)$ we obtain that:

Corollary 4.10. *For some alphabet A with at least two symbols, there is a finitely generated subgroup $G \leqslant \mathrm{Aut}(A^{\mathbb{Z}})$ with undecidable torsion problem.*

In Lemma 7 of [35], it is shown that for every finite set A, and every uncountable sofic subshift X, we have that $\mathrm{End}(A^{\mathbb{Z}})$ embeds into $\mathrm{End}(X)$. In particular $\mathrm{Aut}(A^{\mathbb{Z}})$ embeds into $\mathrm{Aut}(X)$. Letting A be the alphabet of Corollary 4.10 we obtain Corollary 1.3, that is, that the automorphism group of every uncountable sofic $\mathbb{Z}$-subshift contains a finitely generated subgroup G with undecidable torsion problem.

4.4. **The torsion problem of finite-state machines.** The torsion problem of $\mathrm{OB}(\mathbb{Z}^d, n, k)$ is not of much interest, as an element of this group is periodic if and only if its shift component is 0^d. The group $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ for $n \geqslant 2$, however, is quite interesting. Namely, the decidability of the torsion problem turns out to be dimension-sensitive as stated in Theorem 1.4. In this section we prove that theorem in two parts, namely, the one-dimensional part is a corollary of Theorem 4.11 whereas the multidimensional result is given in Theorem 4.12.

We first show that for $d = 1$ the torsion problem is decidable. In fact, we obtain this from a more general result. We say the finiteness problem of a group presentation is decidable if there exists a Turing machine which on input a finite set of words in the presentation accepts if and only if the elements of the group represented by those words generate a finite subgroup. This generalizes the torsion problem which can be regarded as the special case where the set is a singleton.

The proof we presented in the appendix of the conference version [4] contained errors, here we present a new proof which is based on a strong form of the pigeonhole principle, namely the Ramsey theorem. A dynamical proof of the same result is given in [36].

Theorem 4.11. *Let X be a sofic $\mathbb{Z}$-subshift. Then the finiteness problem of $\mathrm{RFA}(X, n, k)$ is decidable for every $n, k \geqslant 1$.*

Proof. Let F be a finite set of Turing machines represented by their local rules and $n \in \mathbb{N}$. By Lemma 4.1 given $u, v \in F^*$ we can compute the relation $\sim$ where $u \sim v$ if and only if $\underline{u} = \underline{v}$. A semialgorithm which accepts if and only if $\langle F \rangle$ is finite computes the set $U_n = (\bigcup_{m \leqslant n} F^n)/\sim$ and accepts if $U_n = U_{n+1}$ for some n. In other words, the finiteness problem is recursively enumerable, it suffices thus to show it is co-recursively enumerable.

For the other direction, we may assume that the movement of the machines in F only depends on the current symbol under them, and the machines move by at most one cell on each step. To see this, take a higher power presentation of X by a sufficiently large power p (see Section 1.4 of [30]), and record the position in a cell modulo p in the state, so that the size of Q changes to kp.

Now, to each word $u \sqsubset X$ associate

$$\phi(u)\colon \{\leftarrow, \rightarrow\} \times Q \to 2^{\{\leftarrow,\rightarrow\} \times Q},$$

where the interpretation of $(d', q') \in \phi(u)(q, d)$ is that, when entering the word u in state q by moving in direction d, it is possible to apply a finite sequence of finite-state machines in F so that eventually the machine exits u by moving in direction d' in state q'. For example by $(\rightarrow, q') \in \phi(u)(\rightarrow, q)$ we mean that there exist $f_1, f_2, \ldots, f_k \in F$ such that

$$(f_k \circ \cdots \circ f_1)(q, x.uy) = (q', xu.y)$$

where x and y are tails such that $xuy \in X$, and the head does not leave u during the "intermediate steps", i.e. for all $1 \leqslant j < k$ we have $(f_j \circ \cdots \circ f_1)(q, xu'.u''y)$ where $u = u'u''$ and $|u'|, |u''| > 0$.

Recall that sofic $\mathbb{Z}$-subshifts have a regular language [30] and thus have an associated syntactic monoid. Associate to each nonempty u its class $\psi(u)$ in the syntactic monoid of $L(X)$. Write $I(u) = (\phi(u), \psi(u))$. Then I is a semigroup homomorphism where defined in the sense that if $uv \sqsubset X$ then

$$I(uv) = (\phi(uv), \psi(uv)) = (\phi(u)\phi(v), \psi(u)\psi(v)) = I(u)I(v),$$

where maps $\phi(u) \in \{\leftarrow, \rightarrow\} \times Q \to 2^{\{\leftarrow,\rightarrow\}\times Q}$ are given a semigroup structure by observing that the possible ways one can exit uv after entering it from the left or right are entirely determined by the corresponding information for u and v.

Now, by the Ramsey theorem, there exists N such that in every word $w \sqsubset X$ of length at least N, there is a subword uv such that u and v are nonempty, and $I(u) = I(v) = I(uv)$. To see this, let w be such a word and color the 2-subset $\{i, j\} \subset N$ by $I(w_{[i,j]})$. If N is large enough, there is a monochromatic subset of size 3, which corresponds to the word uv. In particular, by the previous paragraph it follows that

$$I(u) = I(uv) = I(u)I(v) = I(u)I(u) = I(u^2) \cdots = I(u^k)$$

for all k.[3]

If F generates an infinite group, then there are arbitrarily long words w that the head can walk over, under a suitable application of elements of F. Otherwise, every long enough word blocks movement, so F must generate a finite group. More precisely, we have arbitrarily long words $w \in X$ which are *traversable*, meaning

$$(\rightarrow, q') \in \phi(w)(\rightarrow, q) \text{ or } (\leftarrow, q') \in \phi(w)(\leftarrow, q)$$

for some $q, q' \in Q$.

If w has this property, then all its subwords do. It follows that if F generates an infinite group then there is a traversable word u such that $I(u) = I(uu)$. Once we find such a word u, we have $u^k \sqsubset X$ for all k and that each u^k is traversable, thus F must be infinite. □

Before tackling the problem in the multidimensional case, we recall the **snake tiling problem** introduced and shown to be undecidable in [24]. In this problem, a set T of square tiles with colored edges which have an associated direction arrow is given, and the goal is to find a partial tiling of $\mathbb{Z}^2$ – that is, some positions can be left without tiles – such that if among two adjacent tiles the arrow of one points to the other then they share the same color on the adjacent edge. Furthermore we require that at least one infinite path appears in the partial tiling while following the direction associated to the tiles.

For the next proof we are going to use a slightly modified version of the snake tiling problem, which is also undecidable [24]. Instead of using Wang tiles with just an outgoing direction we are going to use tiles which have both a left and right direction. Formally, let T be a finite set of tiles with colored edges and functions $\texttt{left}, \texttt{right}\colon T \to D$ where $D = \{(1, 0), (-1, 0), (0, 1), (0, -1)\}$ which satisfy $\texttt{left}(t) \neq \texttt{right}(t)$ for each $t \in T$. We are going to ask for a partial tiling $\tau\colon \mathbb{Z}^2 \to T \cup \{\epsilon\}$ such that there exists a function $p\colon \mathbb{Z} \to \mathbb{Z}^2$ such that $\tau(p(n)) \in T$, $p(n + 1) - p(n) = \texttt{right}(\tau(p(n)))$ and $p(n) - p(n + 1) = \texttt{left}(\tau(p(n + 1)))$ for all $n \in \mathbb{Z}$ and all tiles match their non-ϵ neighbors along the arrows. If a partial tiling τ with such a path exists, we say the instance $(T, \texttt{left}, \texttt{right})$ of the problem **admits a snake**.

One way to think about this version is that right arrows give instructions on how to walk to $+\infty$ in this path, while left arrows point to $-\infty$.

Theorem 4.12. *For all $n \geqslant 2, k \geqslant 1, d \geqslant 2$, there is a finitely generated subgroup of $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ whose torsion problem in undecidable.*

Proof. For the rest of the proof we assume $d = 2$. In the general case, the result follows from the obvious fact that $\mathrm{RFA}(\mathbb{Z}^2, n, k)$ embeds into $\mathrm{RFA}(\mathbb{Z}^d, n, k)$.

[3]Alternatively, one can apply the well-known fact that every finite semigroup has an idempotent, but the argument of the present paragraph makes it more explicit why one can pick the idempotent u in the language of X and so that it has the traversability property.

First, let us explain why, if we let the alphabet Σ be arbitrary and take the local rule of an element $f \in \mathrm{RFA}(\mathbb{Z}^2, |\Sigma|, 2)$, it is undecidable whether T_f is a torsion element in that group. We then explain how to do this construction in the case of a fixed alphabet and number of states. Finally, we show that the torsion problem still remains undecidable in a finitely generated subgroup of $\mathrm{RFA}(\mathbb{Z}^2, n, k)$ for every $n \geqslant 2$ and $k \geqslant 1$.

Consider an instance of our modified snake tiling problem $(T, \texttt{left}, \texttt{right})$ where T is the set of tiles. We choose $|\Sigma| > |T|$ and associate the first $|T|$ symbols in Σ to the tiles in T and the rest to the empty tile. We construct a local rule f which gives a non-torsion element T_f if and only if $(T, \texttt{left}, \texttt{right})$ admits a snake.

In this instance we ask for two states, we will call them **direction bits** $Q = \{R, L\}$ standing for right and left. The machine T_f has radius $F = \{(i, j) \in \mathbb{Z}^2 \mid |i| + |j| \leqslant 1\}$ and acts as follows:

- Let t be the tile at $(0, 0)$. If $t = \epsilon$, do nothing.
- Otherwise:
  - If the state is L. Check the tile in the direction $\texttt{left}(t)$. If it matches correctly with t move the head to that position, otherwise switch the state to R.
  - If the state is R. Check the tile in the direction $\texttt{right}(t)$. If it matches correctly with t move the head to that position, otherwise switch the state to L

The machine T_f is reversible and its inverse is given by the machine which does the same but switches the roles of R and L. If $(T, \texttt{left}, \texttt{right})$ admits a snake, it suffices to consider the configuration in $\Sigma^{\mathbb{Z}^2} \times X_q$ that contains an infinite snake and such that the head of the machine is positioned somewhere in the snake. Clearly T_f walks to infinity in that configuration without repeating positions, thus showing that T_f is a non-torsion element (recall we are using the moving head model). Conversely, if $(T, \texttt{left}, \texttt{right})$ does not admit a snake, then there is an uniform bound on how far T_f can walk from its starting position before encountering an error or entering a cycle and henceforth T_f has finite order. This concludes the proof of undecidability of the torsion problem when the alphabet is arbitrary.

Now we explain how to pass to a fixed alphabet and how to avoid using states. For this we encode the tiles into squares of size $n \times n$. By having the bottom left corner of the coding contain $\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}$ and having no two adjacent 1s elsewhere, we ensure that there is a unique way to "parse" a given tiling into encodings of squares. Clearly, the movements of the machine T_f are now inflated by n and anything which is not a valid coding of a tile is interpreted as the empty tile. Also, as there is a unique way to parse correct squares, one can also use the position of the head in the bottom left corner to code the states. Say, the lower left corner of $\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}$ codes L and the lower right corner of $(1, 0)$ codes R. Thus the state change amounts to a shift by either $(1, 0)$ or $(-1, 0)$.

The inconvenient part of the previous construction is that the subgroup of machines defined by it is not necessarily finitely generated. Thus a priori it might be the case that every finitely generated subgroup of $\mathrm{RFA}(\mathbb{Z}^2, n, k)$ still has decidable torsion problem. In order to show this does not hold we construct a finite set of machines which simulates the previous construction. For this we are going to use a specific 7×7 square coding which is shown on Figure 8. This coding is composed of three zones. The **outer zone** consists of a ring of 1s of side length 7 which serves to code unambiguously the boundary of the structure. The four bottom left 1s of this zone are used to code the states, this is obtained by forcing the head of the machine to always stay in one of these positions modulo $\mathbb{Z}^2/7\mathbb{Z}^2$. The **middle zone** consists of a ring of 0s of side length 5 which serves to separate the three zones so no ambiguity is possible. Finally there is the **inner zone** consisting of a 3×3 square containing a configuration in $\{0, 1\}^9$. Four of these bits l_1, l_2, r_1, r_2 serve to code two directions in $D = \{(1, 0), (-1, 0), (0, 1), (0, -1)\}$. The rest of the bits are going to be specified later on.

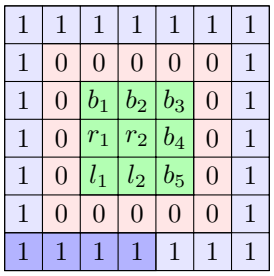


FIGURE 8. Basic coding of the construction. The outer ring of 1s (blue) codes the boundary of the cell and the state. The middle ring of 0s separates the zones The inner ring (green) codes the information.

For this construction we are going to use a two bit string $s \in \{0,1\}^2$ as the set of states (which is to be coded by the head position amongst the four fixed places in the outer ring of 1s). The first bit is the **direction bit**, that is, it takes the role of L and R for the first construction. The second bit is the **auxiliary bit**, whose role will become clear later on.

Let C be the set of all patterns of shape as in Figure 8 centered in one of four fixed positions in the ring of 1s –that is, such that the support is of the form $([-i, 7-i] \times [0,7]) \cap \mathbb{Z}^2$ for some $i \in \{0,1,2,3\}$–. We consider the following finite set of machines as our generating set S.

(1) $\{T_{\vec{v}}\}_{v \in D}$ that walks in the direction $\vec{v} \in D$ independently of the configuration.
(2) $T_{\texttt{walk}}$ that walks along the direction codified by l_1, l_2 or r_1, r_2 depending on the direction bit.
(3) $\{g_c\}_{c \in C}$ that flips the direction bit if the current pattern is $c \in C$,
(4) $\{h_c\}_{c \in C}$ that flips the auxiliary bit if the current pattern is $c \in C$,
(5) $\{g_{+,c}\}_{c \in C}$ that adds the auxiliary bit to the direction bit if the current pattern is $c \in C$, and
(6) $\{h_{+,c}\}_{c \in C}$ that adds the direction bit to the auxiliary bit if the current pattern is $c \in C$,

The machine $T_{\texttt{walk}}$ is the only one which needs to be carefully defined. It acts similarly to T_f defined in the beginning. Formally it does the following:

- If the pattern around the identity does not correspond to a $c \in C$, do nothing.
- Otherwise:
  - If the direction bit is 0 check the pattern centered in $\texttt{7left}(t)$ from the actual head position. If it is a valid $c' \in C$ in the same state and its two right bits code $-\texttt{left}(t)$ then move the head by $\texttt{7left}(t)$. Otherwise flip the direction bit to 1.
  - If the direction bit is 1. Check the pattern centered in $\texttt{7right}(t)$ from the actual head position. If it is a valid $c' \in C$ in the same state and its two left bits code $-\texttt{right}(t)$ then move the head by $\texttt{7right}(t)$. Otherwise flip the direction bit to 0.

The machines $T_{\vec{v}}$ simply act as the shift by $\vec{v} \in D$ which is clearly reversible. As $\langle D \rangle = \mathbb{Z}^2$ we have that for every vector $\vec{u} \in \mathbb{Z}^2$ the machine $T_{\vec{u}}$ which moves the head by $\vec{u}$ belongs to $\langle S \rangle$.

Let p^* be a pattern consisting of the concatenation of patterns from c which are well aligned along the columns and lines of 1s. More formally, for a finite $F \subset \mathbb{Z}^2$, p^* is a pattern with support $7F + (([-i, 7-i] \times [0,7]) \cap \mathbb{Z}^2)$ for some $i \in \{0,1,2,3\}$ and such that for every $\vec{v} \in F$ then $\sigma^{-7\vec{v}}(p^*)|_{([-i,7-i] \times [0,7]) \cap \mathbb{Z}^2} \in C$. We define g_{p^*} and h_{p^*} as the machines which flip the direction bit and the auxiliary bit respectively if they read p^*. We claim $g_{p^*}, h_{p^*} \in \langle S \rangle$. If p^* is defined by some singleton $F = \{\vec{v}\}$ it suffices to note that $g_{p^*} = T_{-7\vec{v}} \circ g_c \circ T_{7\vec{v}}$ and $h_{p^*} = T_{-7\vec{v}} \circ h_c \circ T_{7\vec{v}}$ for the appropriate $c \in C$. Inductively, we can choose $\vec{v} \in F$ and separate p^* as the disjoint union of the pattern $p^*_{F \setminus \{\vec{v}\}}$ defined by $F \setminus \{\vec{v}\}$, and the

pattern $p^*_{\vec{v}}$ defined by $\vec{v}$ and thus write:

$$g_{p^*} = (T_{-7\vec{v}} \circ g_{+,c} \circ T_{7\vec{v}} \circ h_{p^*_{F\setminus\{\vec{v}\}}})^2, \text{ and } h_{p^*} = (T_{-7\vec{v}} \circ h_{+,c} \circ T_{7\vec{v}} \circ g_{p^*_{F\setminus\{\vec{v}\}}})^2.$$

Consider an instance $(T, \texttt{left}, \texttt{right})$ of the snake tiling problem. The information associated to each tile $t \in T$ consists of a 4-tuple of colors (c_1, c_2, c_3, c_4) and the directions $\texttt{left}(t)$ and $\texttt{right}(t)$. Suppose the tiles of T are defined using N colors. Let $M \in \mathbb{N}$ such that $M^2 > \log_2(N)$. We define for each $t \in T$ a macrotile $\mathcal{M}(t)$ as a fixed square array of patterns of shape as in Figure 8 of side length M (see Figure 9). We fix an enumeration of these patterns from the bottom left to the upper right as $\{c_j\}_{1 \leqslant j \leqslant M^2}$ and denote the bit b_i of c_j as $b_{i,j}$. We demand $\mathcal{M}(t)$ to satisfy the following properties:

- For $i \in \{1, 2, 3, 4\}$ the sequence of bits $\{b_{i,j}\}_{1 \leqslant j \leqslant M^2}$ codifies the color c_i.
- $b_{5,1} = 1$ and for all $j > 1$ the bit $b_{5,j} = 0$.
- The bits l_1, l_2 and r_1, r_2 of c_1 code $\texttt{left}(t)$ and $\texttt{right}(t)$ respectively.
- If $\texttt{left}(t) = (1, 0)$ then for all $2 \leqslant j \leqslant M$ we have that l_1, l_2 and r_1, r_2 of c_j code $(1, 0)$ and $(-1, 0)$ respectively.
- If $\texttt{right}(t) = (1, 0)$ then for all $2 \leqslant j \leqslant M$ we have that l_1, l_2 and r_1, r_2 of c_j code $(-1, 0)$ and $(1, 0)$ respectively.
- If $\texttt{left}(t) = (0, 1)$ then for all $1 \leqslant j \leqslant M - 1$ we have that l_1, l_2 and r_1, r_2 of c_{1+jM} code $(0, 1)$ and $(0, -1)$ respectively.
- If $\texttt{right}(t) = (0, 1)$ then for all $1 \leqslant j \leqslant M - 1$ we have that l_1, l_2 and r_1, r_2 of c_{1+jM} code $(0, -1)$ and $(0, 1)$ respectively.

As $M^2 > \log_2(N)$ it is possible to satisfy the first requirement. The rest are possible to satisfy as $\texttt{left}(t) \neq \texttt{right}(t)$. An example of such a macrotile is represented in Figure 9.

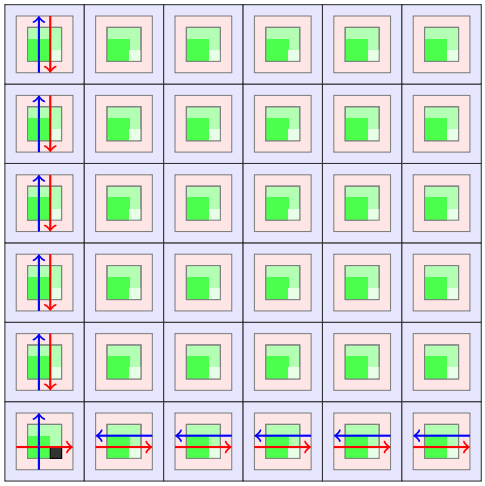
$\mathcal{M}(t) =$ 

FIGURE 9. An example of macrotile $\mathcal{M}(t)$ of side $M = 6$. The red arrows represent the function $\texttt{left}(t) = (1, 0)$ while the blue arrows represent $\texttt{right}(t) = (0, 1)$. The bottom left black square represents $b_{5,1} = 1$.

Associate all arrays of $M \times M$ codings which do not represent some $t \in T$ to the ϵ tile. Also, let $\mathcal{M}$ be the set of all patterns given as an array of 3×3 macrotiles which represent a valid local pattern of the snake problem and such that the middle tile is not an ϵ tile and are centered in the bottom left position of the middle macrotile.

Consider the machine $T^* \in \langle S \rangle$ given by:

$$T^* = (T_{\texttt{walk}})^M \circ \prod_{p^* \in \mathcal{M}} g_{p^*} \circ \prod_{c \in C} g_c$$

We claim that T^* is a torsion element if and only if $(T, \texttt{left}, \texttt{right})$ does not admit a snake.

If $(T, \texttt{left}, \texttt{right})$ admits a snake, it suffices to take a configuration with a snake, replace each tile and ϵ in it by a corresponding macrotile and put the head of the machine in the lower left corner of a macrotile belonging to the snake. The machine T^* will first detect some pattern $c \in C$, so exactly one g_c will flip the direction bit once. Then it will detect a valid pattern p^* of the snake problem and thus g_{p^*} will flip again the direction bit amounting to no action at all. Finally, $(T_{\texttt{walk}})^M$ will just walk towards the lower left corner of the next macrotile. As the initial configuration codes a snake, repeating this procedure will make T^* walk to infinity, and thus T^* is not a torsion element.

For the converse, we need to analyze more carefully the behavior of T^*. First of all, if in the initial configuration the head is not over a pattern $c \in C$, then T^* by definition acts trivially. Otherwise, suppose the head is over a $c \in C$. The application of $\prod_{p^* \in \mathcal{M}} g_{p^*} \circ \prod_{c \in C} g_c$ can at most do a state change, and thus the head still sees some $c' \in C$ afterwards. Also, by definition of $T_{\texttt{walk}}$, the head will always see an element of C after applying $(T_{\texttt{walk}})^M$. This means that the head will always be seeing a pattern in C after applying T^*.

There are two possible behaviors of T^* starting from a pattern in C. If the head is not over a valid array of macrotiles in $\mathcal{M}$ then the direction bit is flipped by g_c, the second part does nothing, and $T_{\texttt{walk}}$ is applied M times. Otherwise the direction bit is flipped two times, amounting to no flip at all and $T_{\texttt{walk}}$ is applied M times.

These two behaviors translate into the following: If the head is over a valid array of macrotiles in $\mathcal{M}$ then T^* can either move into another valid array (and correctly simulate the working of T_f defined at first in the proof), or it can fall outside a valid array of macrotiles. It it does this, then another application of T^* undoes the last M steps of $T_{\texttt{walk}}$ and changes the direction bit. Therefore the machine continues to live inside a valid array of $\mathcal{M}$ and simulate T_f. In this case we can use the uniform bound on the length of the snake to find a bound N such that $(T^*)^N$ acts trivially over all these configurations. The only case remaining is when initially the head is not over an array in $\mathcal{M}$ and after one application of T^* it stays that way. In this case, we just have that $(T^*)^2$ acts trivially over these configurations. Thus showing that $(T^*)^{2N} = id$ and thus T^* is a torsion element of $\langle S \rangle$. □

In the special case where $k = 1$ the previous result can be expressed in dynamical terms. Namely, $\mathrm{RFA}(\mathbb{Z}^d, n, 1)$ is exactly the topological full group of the full $\mathbb{Z}^2$-shift on n symbols. This yields Corollary 1.5, that is, for every $d \geqslant 2$ the topological full group of the full $\mathbb{Z}^2$-shift on n symbols contains a finitely generated subgroup with undecidable torsion problem.

5. Acknowledgments

S. Barbieri was partially supported by the ANR projects CODYS (ANR-18-CE40-0007), CoCoGro (ANR-16-CE40-0005) and the FONDECYT grants 11200037 and 1240085. V. Salo was partially supported by FONDECYT grant 3150552 and Academy of Finland grant 2608073211.

S. Barbieri, Departamento de Matemática y Ciencia de la Computación, Universidad de Santiago de Chile.

E-mail address: sebastian.barbieri@usach.cl

J. Kari, Department of Mathematics and Statistics, University of Turku.

E-mail address: jkari@utu.fi

V. Salo, Department of Mathematics and Statistics, University of Turku.
E-mail address: vosalo@utu.fi

References

[1] S. Aaronson, D. Grier, and L. Schaeffer. The classification of reversible bit operations. In *8th Innovations in Theoretical Computer Science Conference*, volume 67 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 23, 34. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2017.

[2] N. Aubrun, S. Barbieri, and M. Sablik. A notion of effectiveness for subshifts on finitely generated groups. *Theoret. Comput. Sci.*, 661:35–55, 2017.

[3] N. Aubrun and M. Sablik. Simulation of effective subshifts by two-dimensional subshifts of finite type. *Acta Appl. Math.*, 126:35–63, 2013.

[4] S. Barbieri, J. Kari, and V. Salo. The group of reversible Turing machines. In *Cellular automata and discrete complex systems*, volume 9664 of *Lecture Notes in Comput. Sci.*, pages 49–62. Springer, [Cham], 2016.

[5] J. Belk and C. Bleak. Some undecidability results for asynchronous transducers and the Brin-Thompson group $2V$. *Trans. Amer. Math. Soc.*, 369(5):3157–3172, 2017.

[6] J. Belk and F. Matucci. Conjugacy and dynamics in Thompson's groups. *Geom. Dedicata*, 169:239–261, 2014.

[7] V. D. Blondel, J. Cassaigne, and C. Nichitiu. On the presence of periodic configurations in Turing machines and in counter machines. *Theoret. Comput. Sci.*, 289(1):573–590, 2002.

[8] T. Boykett, J. Kari, and V. Salo. Strongly universal reversible gate sets. In *Reversible computation*, volume 9720 of *Lecture Notes in Comput. Sci.*, pages 239–254. Springer, [Cham], 2016.

[9] T. Ceccherini-Silberstein and M. Coornaert. *Cellular automata and groups*. Springer Monographs in Mathematics. Springer, Cham, 2023. Second edition.

[10] T. Ceccherini-Silberstein and M. Coornaert. *Exercises in cellular automata and groups*. Springer Monographs in Mathematics. Springer, Cham, 2023. With a foreword by Rostislav I. Grigorchuk.

[11] E. Czeizler and J. Kari. A tight linear bound on the synchronization delay of bijective automata. *Theoret. Comput. Sci.*, 380(1-2):23–36, 2007.

[12] J.-C. Delvenne, P. Kůrka, and V. Blondel. Decidability and universality in symbolic dynamical systems. *Fund. Inform.*, 74(4):463–490, 2006.

[13] B. Durand, A. Romashchenko, and A. Shen. Effective closed subshifts in 1D can be implemented in 2D. In *Fields of logic and computation*, volume 6300 of *Lecture Notes in Comput. Sci.*, pages 208–226. Springer, Berlin, 2010.

[14] G. Elek and N. Monod. On the topological full group of a minimal Cantor $\mathbf{Z}^2$-system. *Proc. Amer. Math. Soc.*, 141(10):3549–3552, 2013.

[15] A. Gajardo and P. Guillon. Zigzags in Turing machines. In *Computer science—theory and applications*, volume 6072 of *Lecture Notes in Comput. Sci.*, pages 109–119. Springer, Berlin, 2010.

[16] A. Gajardo and J. Mazoyer. One head machines from a symbolic approach. *Theoret. Comput. Sci.*, 370(1-3):34–47, 2007.

[17] A. Gajardo, N. Ollinger, and R. Torres-Avilés. The transitivity problem of Turing machines. In *Mathematical foundations of computer science 2015. Part I*, volume 9234 of *Lecture Notes in Comput. Sci.*, pages 231–242. Springer, Heidelberg, 2015.

[18] T. Giordano, I. F. Putnam, and C. F. Skau. Full groups of Cantor minimal systems. *Israel J. Math.*, 111:285–320, 1999.

[19] R. I. Grigorchuk and K. S. Medinets. On the algebraic properties of topological full groups. *Mat. Sb.*, 205(6):87–108, 2014.

[20] M. Hochman. On the dynamics and recursive properties of multidimensional symbolic systems. *Invent. Math.*, 176(1):131–167, 2009.

[21] E. Jeandel. Computability of the entropy of one-tape Turing machines. In *31st International Symposium on Theoretical Aspects of Computer Science*, volume 25 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages 421–432. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2014.

[22] K. Juschenko and N. Monod. Cantor systems, piecewise translations and simple amenable groups. *Ann. of Math. (2)*, 178(2):775–787, 2013.

[23] J. Kari. Representation of reversible cellular automata with block permutations. *Math. Systems Theory*, 29(1):47–61, 1996.

[24] J. Kari. Infinite snake tiling problems. In *Developments in language theory*, volume 2450 of *Lecture Notes in Comput. Sci.*, pages 67–77. Springer, Berlin, 2003.

[25] J. Kari and N. Ollinger. Periodicity and immortality in reversible computing. In *Mathematical foundations of computer science 2008*, volume 5162 of *Lecture Notes in Comput. Sci.*, pages 419–430. Springer, Berlin, 2008.

[26] A. Kazda. The chain relation in sofic subshifts. *Fund. Inform.*, 84(3-4):375–390, 2008.

[27] P. Kůrka. On topological dynamics of Turing machines. *Theoret. Comput. Sci.*, 174(1-2):203–216, 1997.

[28] P. Kůrka. Erratum to: Entropy of Turing machines with moving head. *Theor. Comput. Sci.*, 411(31-33):2999–3000, June 2010.

[29] C. G. Langton. Studying artificial life with cellular automata. *Phys. D*, 22(1-3):120–149, 1986. Evolution, games and learning (Los Alamos, N.M., 1985).

[30] D. Lind and B. Marcus. *An introduction to symbolic dynamics and coding*. Cambridge University Press, Cambridge, 1995.

[31] H. Matui. Some remarks on topological full groups of Cantor minimal systems. *Internat. J. Math.*, 17(2):231–251, 2006.

[32] H. Matui. Topological full groups of one-sided shifts of finite type. *J. Reine Angew. Math.*, 705:35–84, 2015.

[33] C. Moore. Generalized shifts: unpredictability and undecidability in dynamical systems. *Nonlinearity*, 4(2):199–230, 1991.

[34] R. Pavlov and M. Schraudner. Classification of sofic projective subdynamics of multidimensional shifts of finite type. *Trans. Amer. Math. Soc.*, 367(5):3371–3421, 2015.

[35] V. Salo. A note on subgroups of automorphism groups of full shifts. *Ergodic Theory Dynam. Systems*, 38(4):1588–1600, 2018.

[36] V. Salo. Subshifts with sparse traces. *Studia Math.*, 255(2):159–207, 2020.

[37] V. Salo and I. Törmä. Group-walking automata. In *Cellular automata and discrete complex systems*, volume 9099 of *Lecture Notes in Comput. Sci.*, pages 224–237. Springer, Heidelberg, 2015.

[38] V. Salo and I. Törmä. Plane-walking automata. In *Cellular automata and discrete complex systems*, volume 8996 of *Lecture Notes in Comput. Sci.*, pages 135–148. Springer, Cham, 2015.

[39] P. Selinger. Reversible k-valued logic circuits are finitely generated for odd k. *ArXiv e-prints*, Apr. 2016.

[40] S. Xu. Reversible logic synthesis with minimal usage of ancilla bits. Master's thesis, MIT, June 2015.

Appendix A. Dictionary of groups and monoids

$\mathrm{TM}(\mathbb{Z}^d, n, k)$: the monoid of moving-head Turing machines on $\mathbb{Z}^d$ with n symbols and k states (Definition 2.2).

$\mathrm{RTM}(\mathbb{Z}^d, n, k)$: the group of reversible moving-head Turing machines on $\mathbb{Z}^d$ with n symbols and k states (Definition 2.2).

$\mathrm{TM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$: the monoid of moving-tape Turing machines on $\mathbb{Z}^d$ with n symbols and k states (Definition 2.12).

$\mathrm{RTM}_{\mathrm{fix}}(\mathbb{Z}^d, n, k)$: the group of reversible moving-tape Turing machines on $\mathbb{Z}^d$ with n symbols and k states (Definition 2.12).

$\mathrm{SP}(\mathbb{Z}^d, n, k)$: the group of state permutations, that is, reversible moving-head Turing machines which do not move the head nor change the tape, and may only permute the state.

$\mathrm{LP}(\mathbb{Z}^d, n, k)$: the group of local permutations, that is, reversible moving-head Turing machines which do not move the head (Definition 3.1).

$\mathrm{RFA}(\mathbb{Z}^d, n, k)$: the group of reversible finite-state automata, that is, reversible moving-head Turing machines which do not modify the tape (Definition 3.3).

$\mathrm{SHIFT}(\mathbb{Z}^d, n, k)$: the group of reversible moving-head Turing machines that shift by some vector independent of the tape (this group is naturally isomorphic to $\mathbb{Z}^d$).

$\mathrm{OB}(\mathbb{Z}^d, n, k)$: the group of oblivious Turing machines, that is, the group generated by $\mathrm{LP}(\mathbb{Z}^d, n, k)$ and $\mathrm{SHIFT}(\mathbb{Z}^d, n, k)$ (Definition 3.1).

$\mathrm{EL}(\mathbb{Z}^d, n, k)$: the group of elementary Turing machines, that is, the group generated by $\mathrm{LP}(\mathbb{Z}^d, n, k)$ and $\mathrm{RFA}(\mathbb{Z}^d, n, k)$ (Definition 3.9).