Finding codes on infinite grids automatically

Ville Salo Ilkka Törmä

February 19, 2024

Abstract

We apply automata theory and Karp's minimum mean weight cycle algorithm to minimum density problems in coding theory. Using this method, we find the new upper bound $53/126 \approx 0.4206$ for the minimum density of an identifying code on the infinite hexagonal grid, down from the previous record of $3/7 \approx 0.4286$.

1 Introduction

It is well-known in symbolic dynamics that for a subshift of finite type which has sufficient gluing properties (such as strong irreducibility) the minimal (or maximal) frequency of any letter in a configuration is uniformly computable given the description of the subshift. This observation in particular applies to many subshifts studied in the field of coding theory, such as variants of identifying codes and locating-dominating codes. Indeed, the minimal frequency is lower semicomputable for any subshift of finite type, and under gluing properties it is upper semicomputable simply because it is approximated arbitrarily well by periodic points (even though it may not actually be reached by any periodic point [16]).

This paper is dedicated to Iiro Honkala on his 60th birthday. As Iiro has worked on such density problems, and they are really problems about symbolic dynamics (and ergodic theory) in thin disguise, this felt like a natural common point of interest to look into when we agreed to contribute to this volume. In particular, Ville Junnila suggested to us the study of the minimal density of identifying codes on the hexagonal grid, which we will refer to as α_0 in this article. The lowest-density code known so far was found by Cohen, Honkala, Lobstein and Zemor in 2000 [3], with density $3/7 \approx 0.4286$ (showing $\alpha_0 \leq 3/7$).

The best known lower bound $\alpha_0 \geq 23/55 \approx 0.41818$ was found in 2014 by Derrick Stolee, who automated the so-called discharging method through linear programming [23]. Code construction has also previously been attacked with computer methods, in particular [2] used a heuristic method to find codes for many identifying code problems in regular gridlike graphs. However, this method was not able to improve on the upper bound 3/7 for the hexagonal grid.

We tried two new methods for lowering α_0. First, we naively input this problem into the SAT-solver Gluecard, accessed through the PySAT library [10], to directly compute the minimal densities of periodic configurations of small periods. This was already successful: we found the code of density $11/26 \approx 0.4231$ shown in Figure 3, reducing the gap between Stolee's lower bound and the known upper bound from about 0.0103 to about 0.0049.

Second, as the configurations with one fixed period $\vec{v}$ form in essense a $\mathbb{Z}$-subshift of finite type, one can compute the exact minimal density of a configuration with period $\vec{v}$ using simple graph/automata theory, as it is always reached by a cycle. Specifically, we constructed a finite-state automaton for the $\vec{v}$-periodic configurations for various $\vec{v}$, recognizing the "strip" between periods, and used Karp's minimum mean weight cycle algorithm [13] to compute the exact minimal density. We ran the program on the Dione cluster of the FGCI (Finnish Grid and Cloud Infrastructure) project, which has 80 CPUs and about 200 gigabytes of RAM per computation node. The Python source code is available at [20].

This method found a better code than the SAT solution, with density $53/126 \approx 0.4206$, reducing the gap between Stolee's lower bound and the known upper bound to about 0.0025. The evolution of the bounds is illustrated in Figure 1. The automata method turned out superior to the SAT solver (at least on this one problem), as the solver was not able to reproduce the optimal solution even given the periods.

Theorem 1. *The minimal density identifying code on the hexagonal grid is at most* $53/126$.

Proof. In Figure 4, we exhibit an identifying code with vertical period 126, and horizontal period 1 with shear 11. □

We also looked at four other problems from the literature with the automata-theoretic method, namely

- the optimal radius-2 locating-dominating codes on the king grid α_1, previously known bounds $\alpha_1 \in [1/10, 1/8]$ from [2, 17, 18].
- the optimal redundant radius-1 locating-dominating codes on the king grid α_2, previously known bounds $\alpha_2 \in [3/11, 5/16]$ from [11],
- the radius 2 identifying codes for the square grid α_3, previously known bounds $\alpha_3 \in [6/35, 5/29]$ from [12, 9], and
- the radius 2 identifying codes for the triangular grid α_4, previously known bounds $\alpha_4 \in [2/15, 1/6]$ from [1].

In each case, we ran the search for all height-shear combinations until the cluster ran out of memory. We were not able to improve on any of the densities, which could mean that our method does not apply well to these codes, or alternatively might suggest that these codes are (close to) optimal and more work is needed on the side of lower bounds.

The most interesting discovery we made from these searches is the new locating-dominating code of density $1/8$ for α_1, which has periods $(40, 0)$ and $(-6, 1)$, and took 87 hours of computation time on the Dione cluster to find. This code is pictured in Figure 2. The previously known code from [2] simply repeats the block

$$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{matrix}$$

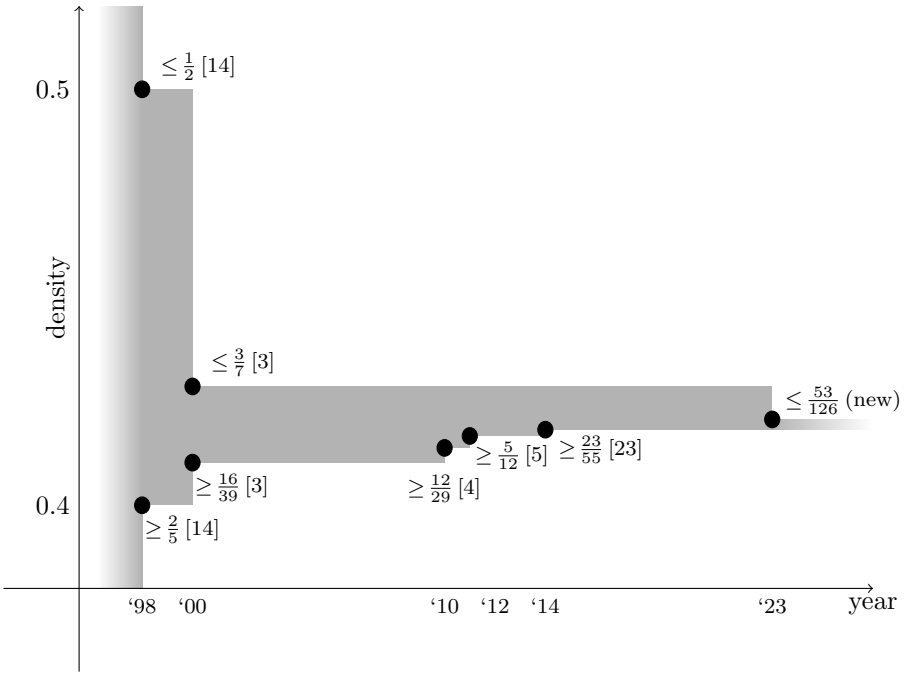

Figure 1: Evolution of the known lower and upper bounds on the minimal density of an identifying code on the hex grid. Years are based on publication years for [14] and [3], and arXiv years for others. The problem was introduced in [14].

with horizontal and vertical period 4. Interestingly, the old code is also identifying, but our new code in Figure 2 is not. The other periods we tried did not produce codes that reach or surpass the known upper bound.

2 Preliminaries

We recall some definitions from multidimensional symbolic dynamics. Let A be a finite alphabet. A d-dimensional *pattern* is a mapping $P : D \to A$ for a finite or infinite *domain* $D = D(P) \subset \mathbb{Z}^d$. The d-dimensional *full shift* over A is the set $A^{\mathbb{Z}^d}$ of full patterns equipped with the prodiscrete topology, which is generated by the clopen *cylinder sets* $[P] = \{x \in A^{\mathbb{Z}^d} \mid x|_{D(P)} = P\}$ for all finite patterns P. The group $\mathbb{Z}^d$ acts on $A^{\mathbb{Z}^d}$ by the *shifts* $\sigma_{\vec{v}} : A^{\mathbb{Z}^d} \to A^{\mathbb{Z}^d}$, defined by $\sigma_{\vec{v}}(x)_{\vec{n}} = x_{\vec{n}+\vec{v}}$, which are homeomorphisms. If $d = 1$, we denote $\sigma_1 = \sigma$ and call it the *left shift*.

A *subshift* is a topologically closed σ-invariant subset $X \subset A^{\mathbb{Z}^d}$. The elements $x \in X$ are called *configurations*. Every subshift is defined by some set of *forbidden finite patterns* F:

$$X = \bigcap_{P \in F} \bigcap_{\vec{v} \in \mathbb{Z}^d} \sigma_{\vec{v}}(A^{\mathbb{Z}^d} \setminus [P]).$$

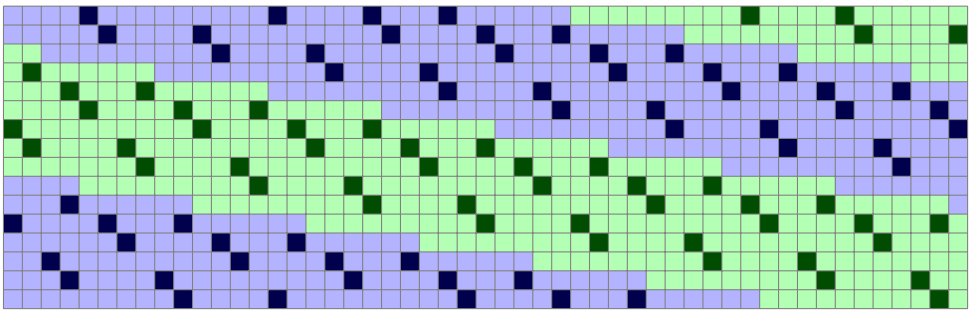

Figure 2: A new locating-dominating code of density 1/8 for the radius-2 king grid. The colors green and blue are included to guide the eye and show the repeating pattern (in the form discovered by the algorithm, i.e. a row of width 40).

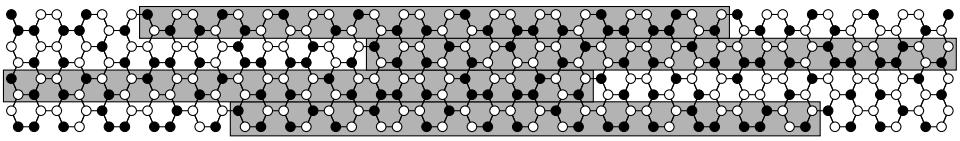

Figure 3: An identifying code with density 11/26 on the hexagonal grid, found using the SAT solver Gluecard.

If F can be chosen finite, then X is a *shift of finite type* (SFT). The patterns P such that $[P] \cap X \neq \emptyset$ form the *language* of the subshift X.

A subshift $X \subset A^{\mathbb{Z}^d}$ is *strongly irreducible* if there exists a *gluing radius* $r \geq 0$ such that for all configurations $x, y \in X$ and all sets $U, V \subset \mathbb{Z}^d$ with $\min\{\|\vec{v} - \vec{w}\| \mid \vec{v} \in U, \vec{w} \in V\} \geq r$, there exists a third configuration $z \in X$ with $z|_U = x|_U$ and $z|_V = y|_V$. It has the *uniform filling property* if this holds whenever U is a hypercube, and it is *block gluing* if this holds whenever both U and V are hypercubes.

We use several types of graphs. If V is a (countable, but possibly infinite) set and $E \subset \{(u, v) \mid u, v \in V, u \neq v\}$, then $G = (V, E)$ is a *directed graph*. If $(u, v) \in E$ implies $(v, u) \in E$, we say G is an *undirected graph*. When an undirected graph is understood from context, $v \in V$ and $r \geq 1$, we write $B_r(v)$ for the closed ball of radius r in (V, E) with the path metric, or explicitly

$$B_r(v) = \{u \in V \mid \exists n \leq r : \exists v = v_0, v_1, \ldots, v_n = u : \forall i : \{v_i, v_{i+1}\} \in E\}.$$

A *weighted graph* is a triple $G = (V, E, \lambda)$ where (V, E) is a graph and $\lambda : E \to \mathbb{R}$ associates a real weight to each edge. In our applications, weighted graphs are always finite and directed, and they should be thought of as finite state automata.

A *radius-r identifying code* [14] on an undirected graph $G = (V, E)$ is a subset $C \subset V$, whose elements are called *codewords*, satisfying the following conditions:

- $B_r(u) \cap C \neq \emptyset$ for all $u \in V$,
- $B_r(u) \cap C \neq B_r(v) \cap C$ for all distinct $u, v \in V$.

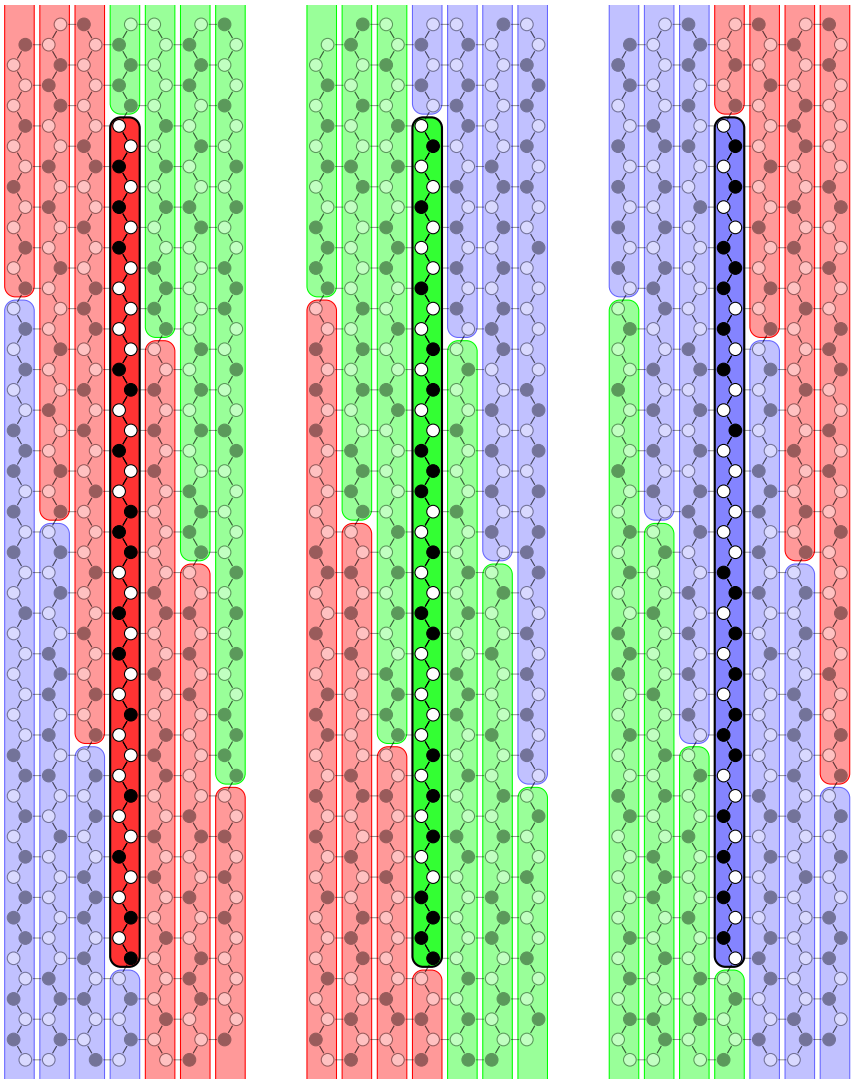

Figure 4: An identifying code with density 53/126 on the hexagonal grid found using automata theory and Karp's algorithm.

The set $B_r(u) \cap C$ is called the *identifying set* of u in C. If we only require the second condition for distinct $u, v \in V \setminus C$, then the code is called a *radius-r locating-dominating code* [21]. By definition every identifying code is locating-dominating, but the converse is typically not true. When the radius is not mentioned, it is taken to be 1. A *redundant radius-r locating-dominating code* is a radius-r locating-dominating code that stays radius-r locating-dominating even if any single codeword is removed. These were first studied in [22].

Like in the case of subshifts, the set 2^V of all functions $x : V \to \{0, 1\}$ is considered with the topology with clopen basis given by cylinders $[P] = \{x \in 2^V \mid x|_D = P\}$ for patterns $P : D \to \{0, 1\}$ with finite domain $D = D(P) \subset V$, and is homeomorphic to Cantor space when V is countably infinite. It is easy to see that radius-r identifying codes are a closed subset of 2^V, and the same is true for (redundant) locating-dominating codes.

Suppose that $\mathbb{Z}^d$ acts freely on the graph $G = (V, E)$ by graph automorphisms in such a way that the vertices have a finite number of distinct orbits. Let $D \subset V$ be a set of representatives for these orbits. Then V is the disjoint union $\bigcup_{\vec{v} \in \mathbb{Z}^d} (\vec{v} \cdot D)$, and the set 2^V of all functions $x : V \to \{0, 1\}$ is isomorphic to the full shift $(2^D)^{\mathbb{Z}^d}$ in the obvious way. Furthermore, the set of radius-r identifying codes corresponds to a strongly irreducible subshift of finite type $X \subset (2^D)^{\mathbb{Z}^d}$, and the same is true for the other types of codes as well.

We recall a few basic facts about abelian groups: every finitely-generated group G which is torsion-free (meaning the identity element is not a proper power of another element) is isomorphic to $\mathbb{Z}^d$ under addition, for some d. In particular, all subgroups $L \leq \mathbb{Z}^d$ are isomorphic to $\mathbb{Z}^e$ for some $e \leq d$ (e cannot be larger than d by basic algebra, or even by considering growth rates). Since $\mathbb{Z}^d$ and $\mathbb{Z}^{d'}$ are not isomorphic when $d \neq d'$, the value of e is uniquely determined by the subgroup L, and this is called the *rank* of L. If $L \leq \mathbb{Z}^d$ is of rank $e < d$, then there exists $\vec{v} \in \mathbb{Z}^d$ such that $\langle L, \vec{v} \rangle$ has rank $e + 1$. Indeed, one can pick any $\vec{v}$ outside the convex hull of L (which is an e-dimensional subspace of $\mathbb{R}^d$). A rank-d subgroup of $\mathbb{Z}^d$ is sometimes called a *lattice*. If X is a subshift, we say $x \in X$ is *totally periodic* if the set of shifts that fix it form a lattice.

We also need basic definitions from ergodic theory. For general references on ergodic theory see e.g. [6, 24, 8]. Here, we only consider ergodic theory on subshifts on groups $\mathbb{Z}^d$. For the purpose of measure theory, a subshift $X \subset A^{\mathbb{Z}^d}$ is always considered with its Borel σ-algebra (generated by the clopen sets). By a *measure* we always mean a Borel probability measure μ, i.e. $\mu(X) = 1$ and μ gives a measure precisely to the Borel sets. We write $\mathcal{M}$ for the set of all measures on X, and $\mathcal{M}_\sigma$ for the shift-invariant ones, i.e. those μ satisfying $\mu(\sigma_{\vec{v}}(B)) = \mu(B)$ for all Borel sets B and all vectors $\vec{v} \in \mathbb{Z}^d$. A measure is *ergodic* if $\mu(B) \in \{0, 1\}$ for all Borel sets $B \subset X$ satisfying $\sigma_{\vec{v}}(B) = B$.

If X, Y are topological spaces, $f : X \to Y$ is a *simple continuous function* if it is continuous and for some finite partition $C_1, \ldots, C_n$ of X and values $y_1, \ldots, y_n \in Y$, we have $f|_{C_i} \equiv y_i$. Note that if Y is Hausdorff, the C_i must form a clopen partition. If X is a d-dimensional subshift, a simple continuous function $f : X \to Y$ has a *radius* R such that $f(x)$ only depends on $x|_{B_R}$, where $B_R \subset \mathbb{Z}^d$ denotes the ball of radius R with respect to Manhattan distance.

If X is a subshift and μ is a measure on X, then a function $f : X \to \mathbb{R}_{\geq 0}$ is $L^1(\mu)$ (or just L^1) if f is measurable (preimages of intervals are Borel sets) and the Lebesque integral $\int f d\mu$ is finite. It is $L^\infty(\mu)$ (or just L^∞) if it is measurable

and for some $X_0 \subset X$ with $\mu(X_0) = 1$ and $r < \infty$ we have $f(x) < r$ for all $x \in X_0$. All simple continuous functions to $\mathbb{R}_{\geq 0}$ are of course L^1 and L^∞.

If $f : X \to \mathbb{R}_{\geq 0}$ is an L^1 function and $\mathcal{I}$ is a σ-algebra contained in the Borel sets of X, then $\mathbb{E}_\mu(f|\mathcal{I})$ denotes the expectation of f on $\mathcal{I}$, i.e. the unique $\mathcal{I}$-measurable (meaning preimages of intervals are in $\mathcal{I}$) function $g : X \to \mathbb{R}_{\geq 0}$ whose integral w.r.t. μ agrees with that of f on every $B \in \mathcal{I}$. Such an expectation always exists, and is unique in the sense that any two expectations of f on $\mathcal{I}$ agree in values on a set of measure 1.

We also need a few notions from computability; for a general reference see e.g. [19]. A subshift is *effectively closed*, if it can be defined by a computably enumerable set of forbidden patterns, i.e. a Turing machine can output some defining set of forbidden patterns (in any order). Equivalently, the entire complement of the language of the subshift can be enumerated. Such a machine is called an *effective description (by forbidden patterns)* of the subshift. If a subshift is explicitly assumed to be an SFT, its description is a finite list of forbidden patterns. A real number $\alpha \in \mathbb{R}$ is *lower (upper) semicomputable* if there exists an algorithm that outputs (in any order) all rational numbers strictly smaller (greater) than α. If a real number is both lower and upper computable, we say it is computable.

Consider a set L of finite words over some finite alphabet and a function $F : L \to \mathbb{R}$. In our applications, L will typically be a set of effective descriptions of some subclass of effectively closed subshifts, and $F(w)$ will be the minimum density of the subshift described by the Turing machine with encoding $w \in L$. We say $F(w)$ is *lower (upper) semicomputable uniformly in* w if there exists a single algorithm that, given w, outputs all rational numbers strictly smaller (greater) than $F(w)$. If $F(w)$ is uniformly both lower and upper semicomputable in w, we say it is *computable uniformly in* w. If $F(w)$ is always rational and there exists an algorithm that, given w, computes it in finite time, we say $F(w)$ is *exactly computable from* w. Note that being exactly computable from w is a strictly stronger property than being rational and computable uniformly in w. In all of the above cases, the algorithm is allowed to do anything on input words that are not in L, including producing no output.

3 Densities of symbols in subshifts

Definition 1. *Let $X \subset A^{\mathbb{Z}^d}$ be a subshift. The* minimum density *of a symbol $a \in A$ is defined as*

$$\inf_{x \in X} \lim_{n} \frac{|\{\vec{v} \in [-n, n]^d \mid x_{\vec{v}} = a\}|}{(2n+1)^d},$$

where for those $x \in X$ for which the limit does not exist, we evaluate it to ∞.

We recall some basic results in ergodic theory (which hold in much greater generality). The use of these results is to replace various spatial averages by measures, which simplifies arguments where the geometric areas where density is computed can have different shapes.

The first result is a version of the famous pointwise ergodic theorem, and is a direct specialization of [8, Theorem 14.A8]. For this theorem, consider any sequence of hypercubes $H_n = \vec{v}_n + [1, m_n]^d$ with $m_n \longrightarrow \infty$, such that $H_n \subset H_{n+1}$ for all n, and for $f : X \to \mathbb{R}_{\geq 0}$ define $(R_n f)(x) = \frac{1}{m_n^d} \sum_{\vec{v} \in H_n} f(\sigma_{\vec{v}}(x))$.

Theorem 2. *Let $X \subset A^{\mathbb{Z}^d}$ be a subshift, and $\mathcal{I}$ the family of shift-invariant Borel sets. Then for any simple continuous function $f : X \to \mathbb{R}_{\geq 0}$ we have $\lim_n (R_n f)(x) = \mathbb{E}_\mu(f|\mathcal{I})(x)$ for all $x \in X_0$, for some X_0 with $\mu(X_0) = 1$.*

Corollary 1. *Suppose that X is a $\mathbb{Z}^d$-subshift with invariant ergodic measure μ, and $f : X \to \mathbb{R}_{\geq 0}$ is a simple continuous function. Letting $\alpha = \int f d\mu$, for some $X_0 \subset X$ with $\mu(X_0) = 1$ we have $(R_n f)(x) \longrightarrow \alpha$ for all $x \in X_0$.*

Proof. By the previous theorem, $(R_n f)(x) \longrightarrow \mathbb{E}_\mu(f|\mathcal{I})(x)$ for all $x \in X_0$, where $\mu(X_0) = 1$. The function $\mathbb{E}_\mu(f|\mathcal{I})$ is shift-invariant almost everywhere, so since μ is ergodic, it must be a constant function almost everywhere, i.e. $f(x) = \alpha$ for x in a set of measure 1. We recall the argument: consider $A = f^{-1}(r)$. We have $\mu(\sigma_{\vec{v}}(A)\Delta A) = 0$ since f is shift-invariant almost everywhere, where Δ denotes symmetric difference, so also $\mu((\bigcup_{\vec{v}\in\mathbb{Z}^d} \sigma_{\vec{v}}(A))\Delta A)| = 0$. Since $\bigcup_{\vec{v}\in\mathbb{Z}^d}(\sigma_{\vec{v}}(A)) \in \mathcal{I}$, its measure is in $\{0, 1\}$, so the same is true for A.

This constant α must be $\int f d\mu$, since by definition the expectation has the same integral as f over the whole space, or in a formula

$$\alpha = \int \alpha d\mu \int \mathbb{E}_\mu(f|\mathcal{I}) d\mu = \int f d\mu. \qquad \square$$

From now on, we concentrate on the hypercubes $H_n = [-n, n]^d$.

Lemma 1. *Let X be a $\mathbb{Z}^d$-subshift and $f : X \to \mathbb{R}_+$ be a simple continuous function. The infimum*

$$\inf_{x\in X} \lim_n \frac{\sum_{\vec{v}\in[-n,n]^d} f(\sigma_{\vec{v}}(x))}{(2n+1)^d}$$

is always reached by some configuration, and is equal to $\min_{\mu\in\mathcal{M}} \int f(x) d\mu(x)$.

Note that, setting $f(x) = \begin{cases} 1 & \text{if } x_{\vec{0}} = a \\ 0 & \text{otherwise,} \end{cases}$ this applies to the minimum density.

Proof. Let α be the value of the infimum. Let the radius of f be R. If $x \in X$, write $\imath(x) = \lim_n \frac{\sum_{\vec{v}\in[-n,n]^d} f(\sigma_{\vec{v}}(x))}{(2n+1)^d}$ when this limit exists. For any $x \in X$ where it exists, we can construct a shift-invariant measure μ such that $\int f(x) d\mu(x) \leq \imath(x)$. Namely, let $\mu_n = \frac{1}{(2n+1)^d} \sum_{\vec{v}\in[-n,n]^d} \delta_{\sigma_{\vec{v}}(x)}$ where δ_y denotes the Dirac measure on a configuration y (giving probability 1 to y). The measure μ_n is just the convex combination of the Dirac measures, i.e. picks a random shift of x by some $\vec{v} \in [-n, n]^d$. Thus the measure $\int f(x) d\mu_n(x)$ is the sum of f-values of shifts of x in the hypercube $[-n, n]^d$. Note that this sum only depends on the contents of x in $[-n - R, n + R]^d$.

Taking a weak-$*$ limit of these measures along the sequence of n reaching the $\liminf$ defining α (using weak-$*$ compactness of the set of measures), we have $\int f(x) d\mu(x) = \alpha$. Furthermore, it is easy to see that $\mu(C) = \mu(\sigma_{\vec{u}}(C))$ for any clopen set C and any $\vec{u} \in \mathbb{Z}^d$. Namely, $|\mu_n(C) - \mu_n(\sigma_{\vec{u}}(C))| \to 0$ because $\mu_n(C)$ counts the sum of f-values of shifts of x in the hypercube $[-n, n]^d$, and $\mu_n(\sigma_{\vec{u}}(C))$ the same over $[-n, n]^d + \vec{u}$, and these counts differ by at most $O(n^{d-1})$, so the difference disappears in the normalization by $\frac{1}{(2n+1)^d}$ as $n \to \infty$.

The above shows that $\inf_{\mu\in\mathcal{M}} \int f(x)d\mu(x) \leq \alpha$. We now observe that $\int f(x)d\mu(x)$ is continuous in μ (by the definition of the topology), so because $\mathcal{M}$ is weak-$*$ compact, the minimum is reached by some measure μ, showing $\min_{\mu\in\mathcal{M}} \int f(x)d\mu(x) \leq \alpha$.

Next, let μ be any shift-invariant measure. Recall that $\mathcal{M}$ is spanned by its ergodic measures, meaning that μ can be written as a (continuous) convex combination of ergodic measures ν, i.e. measures where every Borel set has measure 0 or 1. In a formula, $\mu(A) = \int \mu_y(A)d\nu(y)$ for all Borel sets A, for some probability measure ν on the set of ergodic measures μ_y. See [8, 14.10]. In particular, some ergodic measure μ_y must satisfy $\int f(x)d\mu_y(x) \leq \alpha$ as well.

The pointwise ergodic theorem for $\mathbb{Z}^d$-actions says that for an ergodic measure μ_y, for μ_y-almost all configurations $x \in X$ we have

$$\lim_n \frac{\sum_{\vec{v}\in[-n,n]^d} f(\sigma_{\vec{v}}(x))}{(2n+1)^d} = \int f(x)d\mu_y(x),$$

showing $\alpha \leq \int f(x)d\mu_y(x) = \min_{\mu\in\mathcal{M}} \mu([a])$. □

When X is a subshift of finite type with the uniform filling property, the infimum is reached by some configuration by a simple argument that avoids the use of ergodic theory (or even measures). Namely, we can reach it by gluing together squares with small integral as in [7].

Before getting to our actual method in the next section, we prove in the remainder of this section that just from general principles, the minimum density of any symbol can be computed for most subshifts in coding theory. Of course, these methods require being able to efficiently calculate minimal densities in very large finite graphs.

Lemma 2. *Let $X \subset A^{\mathbb{Z}^d}$ be an effectively closed subshift. Then the minimum density of any $a \in A$ in X is lower semicomputable uniformly in the description of X by forbidden patterns.*

Proof. Let α be the minimum density. By the definition of X, for each n, we can calculate better and better upper approximations to the set of $[-n,n]^d$-patterns in X. For each n, define

$$d_n = \frac{\min\{|\{\vec{v} \in [-n,n]^d \mid x_{\vec{v}} = a\}| \mid x \in X\}}{(2n+1)^d}.$$

We can calculate arbitrarily good lower approximations to these d_n, since the minimum can only increase as the set becomes smaller.

Partitioning a configuration $x \in X$ into disjoint translates of $[-n,n]^d$, we see $\alpha \geq d_n$ for all n. On the other hand, if x reaches the infimum defining α, then writing f_n for the density of as in $[-n,n]^d$, we have $f_n \to \alpha$. Since $f_n \geq d_n$, we have $\liminf_n d_n \leq \alpha$. On the other hand, we can easily construct a shift-invariant measure giving measure $\liminf_n d_n$ to $[a]$ as in the proof of Lemma 1 so

$$\alpha = \min_{\mu\in\mathcal{M}} \mu([a]) \leq \liminf_n d_n \leq \alpha.$$

Now $\alpha \geq d_n$ and $\liminf_n d_n = \alpha$ together imply $\lim d_n = \alpha$. Thus, lower approximations of d_n over all n provide converging lower approximations to α. □

Lemma 3. *Let $X \subset A^{\mathbb{Z}^d}$ be a subshift of finite type. Suppose that one of the following conditions holds:*

- *the language of X is computable, X is block-gluing, and a gluing radius r is known, or*
- *X has a totally periodic point, and X has the uniform filling property.*

Then the minimum density of any $a \in A$ is computable uniformly in the description of X.

More precisely, we show that there exists a Turing machine such that given a Turing machine that describes the language of a subshift which is block-gluing and of gluing radius r, it computes arbitrarily good upper and lower approximations to the minimum density (assuming this data is correct). Under the second condition, we show a bit more: given the description of any SFT X, if X happens to have a totally periodic point and happens to have the uniform filling property, then the algorithm correctly computes the density in the sense that it eventually enumerates all lower and upper approximations of the density, and even if it is given an SFT *without* these properties, it never outputs an incorrect approximation.

For example, any subshift of finite type with a *safe symbol* (one contained in no forbidden pattern) satisfies both conditions. When $d \leq 2$, block gluing implies[1] the existence of a totally periodic point; in particular in dimension two we can compute the minimum density of any subshift of finite type with the uniform filling property.

Proof. Let the minimum density be α. First suppose the first condition holds. Since the language is computable, we can compute the minimal density d_n of a pattern on $[-n, n]^d$. As seen in the previous proof, $\alpha \geq d_n$. On the other hand, if P has density d_n, then we can glue together a grid of copies of P with density at most $d_n + O(r/n)$ showing $\alpha \leq d_n + O(r/n)$. If r is known, this shows that the minimum density is indeed computable.

Suppose next that the second condition holds. We conclude lower semicomputability from the previous lemma and show upper semicomputability only. First, we can find a totally periodic point $x \in X$ in finite time. Then, for any pattern P, we can successively glue P into a periodic set of positions in x using UFP so that the occurences are disjoint, and by the total periodicity of x and the fact X is of finite type, we can always perform this gluing the same way as long as the gluing positions are on the period lattice of x. The density of the resulting configuration is at most $\alpha + O(r/n)$.

This means that totally periodic points can be used to give converging upper approximations to α, which shows its upper semicomputability since the densities of totally periodic points can be easily enumerated in a subshift of finite type. □

[1]For a proof that the uniform filling property implies this, see [15]. For our stronger claim, glue together a vertical stack of copies of a horizontal strip. Since we are in an SFT, we can use the same gluing between any two of them, and thus we obtain a vertically periodic point. Then we recall that a two-dimensional SFT with a vertically periodic point has a totally periodic point.

4 Density of periodic configurations

Recall that a *simple cycle* in a graph G is a cycle that does not repeat vertices.

Lemma 4. *Let $G = (V, E, \lambda)$ be a finite directed weighted graph. Then the minimum density of a bi-infinite walk on G is reached by a simple cycle.*

Proof. Let α be the minimum density of a bi-infinite walk. We first show that α is also the infimum of the densities of all cycles in G. Let x be a bi-infinite walk of density α. For all $n \in \mathbb{N}$, the walk $w = x_{[-n,n]}$ can be cut into cycles plus at most $|V|$ additional edges. Namely, consider the leftmost node $q \in V$ in w. If it occurs in w at least twice, then we can remove from w a prefix that also ends in q, which is a cycle. If q occurs only once, we remove the leftmost edge of w. Then we continue with the remaining part of w. The second case can occur only once per each node of V, so at most $|V|$ times in total. If the density of w is close to α, then at least one of these cycles must have density close to α as well.

We now claim that α is also the infimum (and hence minimum) density of all simple cycles on G. Let c be a cycle whose density is close to α. If c is not simple, it repeats a vertex and we can dissect it into two strictly shorter cycles, c_1 and c_2. The density of c is an affine combination of the densities of c_1 and c_2, so one of them is less than or equal to the density of c. We repeat this procedure until a simple cycle remains. □

Lemma 5. *If (Z, σ) is a subshift of finite type, and $f : Z \to \mathbb{Q}_+$ is a simple continuous function, then the minimal f-density of a configuration of Z is a rational number, exactly computable from the descriptions of Z and f.*

Proof. We simply perform a recoding argument to reduce to the previous lemma. Specifically, any subshift of finite type is conjugate to the set of paths in a graph: take the edges to represent words of length $k + 1$ in the SFT, and nodes words of length k, where the maximal length of a forbidden word is at most $k + 1$; if k is further larger than the radius of f, then we can represent f-images by edge weights.

By the previous lemma, the minimal average weight of a configuration is reached by a cycle. The minimal average weight of a cycle, in turn, can be computed by Karp's algorithm [13]. □

Remark 1. *Recall that a d-dimensional* sofic shift *is a subshift of the form $\pi(X)$, where $X \subset A^{\mathbb{Z}^d}$ is an SFT and $\pi : X \to B^{\mathbb{Z}^d}$ is a relabeling map, defined by $\pi(x)_{\vec{n}} = \alpha(x_{\vec{n}})$ for all $\vec{n} \in \mathbb{Z}^d$ for some symbol-to-symbol function $\alpha : A \to B$. The previous lemma is also true for one-dimensional sofic shifts, as we can simply replace a sofic shift $Z = \pi(X)$ by the SFT X, and the function f by $f \circ \pi$.*

Lemma 6. *Let $Y \subset A^{\mathbb{Z}^d}$ be a subshift of finite type, and let $f : Y \to \mathbb{N}$ be a simple continuous function. Let $L \subset \mathbb{Z}^d$ be a rank-$(d-1)$ subgroup. Then the minimal f-density of an L-periodic configuration in Y is a rational number, exactly computable from the descriptions of Y, f and L.*

Proof. Let $\vec{v}$ be such that $\hat{L} = \langle L, \vec{v} \rangle$ has rank d. Let F be a set of coset representatives for $\hat{L}$, meaning $\hat{L} + F = \mathbb{Z}^d$, and each $\vec{v} \in \mathbb{Z}^d$ has a unique

representation as such a sum. Now consider the action of the shift by $\vec{v}$ on the L-periodic configurations Y_L in Y. It is easy that $\sigma_{\vec{v}}(Y_L) = Y_L$, and the system $(Y_L, \sigma_{\vec{v}})$ is topologically conjugate to a subshift of finite type, by the map $\phi : (Y_L, \sigma_{\vec{v}}) \to (Z, \sigma) \subset ((A^F)^{\mathbb{Z}}, \sigma)$ defined by $(\phi(x)_i)_{\vec{u}} = x_{i\vec{v}+\vec{u}}$.

The density of an L-periodic configuration is of course just $\inf_{\mu \in \mathcal{M}(Y_L)} f(x) d\mu(x)$ by Lemma 1, where $\mathcal{M}(Y_L)$ runs over $\mathbb{Z}^d$-invariant measures of Y_L. Letting $g(x) = \sum_{\vec{u} \in F} \sigma_{\vec{u}}(x)/|F|$, we see that

$$\inf_{\mu \in \mathcal{M}(Y_L)} f(x) d\mu(x) = \inf_{\mu \in \mathcal{M}(Y_L, \hat{L})} g(x) d\mu(x)$$

where $\mathcal{M}(Y_L, \hat{L})$ denotes the $\hat{L}$-invariant measures on Y_L (meaning measures invariant under translations $\sigma_{\vec{u}}$ with $\vec{u} \in \hat{L}$). Namely, any measure on the left gives the same integral on the right (since $\mathbb{Z}^d$-invariance implies $\hat{L}$-invariance), and any measure on the right gives the same measure on the left after taking the convex combination of its shifts over F.

Now since Y_L is L-invariant by definition, we have

$$\inf_{\mu \in \mathcal{M}(Y_L, \hat{L})} g(x) d\mu(x) = \inf_{\mu \in \mathcal{M}(Y_L, \sigma_{\vec{v}})} g(x) d\mu(x).$$

But as discussed $(Y_L, \vec{v})$ is topologically conjugate to some SFT (Z, σ) and g corresponds to some simple continuous function h. By the previous lemma, this value can be computed exactly. □

Definition 2. *The* hexagonal grid *is the undirected graph* $(\mathbb{Z}^2, E)$ *where* $E = \{\{(x, y), (x, y+1)\}, \{(x, y), (x+p, y)\} \mid (x, y) \in \mathbb{Z}^2, p = (-1)^{x+y}\}$.

In words, the formula says that we include all vertical edges, and we include an edge between (x, y) and $(x+1, y)$ if and only if $x+y$ is even. It is easy to see that this grid is isomorphic (as a graph) with the hexagonal grid as depicted in Figure 3 and Figure 4.

Suppose (V, E) is a graph of subexponential growth and $X \subset A^V$. Then we define the *minimal density* of a symbol $a \in A$ as

$$\inf_{x \in X} \lim_n \frac{|\{\vec{v} \in B_n \mid x_{\vec{v}} = a\}|}{|B_n|},$$

where B_n is the ball of radius n in the graph, around some fixed origin $o \in V$.

Remark 2. *This definition of density does not make much sense on graphs with exponential growth, where the boundary of a ball usually contains a positive proportion of its elements. In particular for subshifts on groups of exponential growth better definitions are obtained from Følner sequences (for amenable groups) and sofic approximations (for sofic groups).*

In the case of $(\mathbb{Z}^2, E_{\text{hex}})$, it is easy to see that translations by the group $T = \langle (0, 2), (1, 1) \rangle$ are graph automorphisms. Define $X_{\text{hex}} \subset \{0, 1\}^{\mathbb{Z}^2}$ as the identifying codes on the graph $(\mathbb{Z}^2, E_{\text{hex}})$. The set X_{hex} is in bijection with a $\mathbb{Z}^2$-subshift of finite type Y_{hex} over alphabet $\{0, 1\}^2$: Set $v_1 = (0, 2), v_2 = (1, 1)$, and use the action τ defined by $\tau_{(m,n)}(x) = \sigma_{mv_1+nv_2}(x)$ on X. Record in each vertex the value in that vertex and the vertex on top of it. The forbidden patterns of Y_{hex} are induced by subsets of $\mathbb{Z}^2$ that cannot all contain a 0 in a

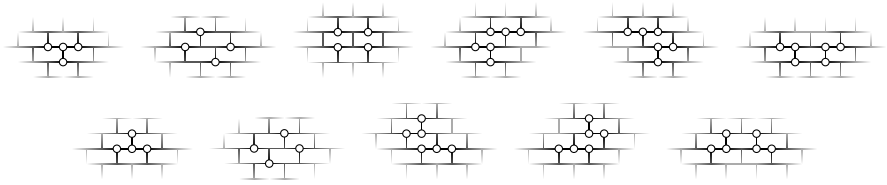

Figure 5: The minimal set F_{hex} of forbidden patterns for identifying codes on E_{hex}. The circles denote non-codewords.

valid identifying code, either because some node would have no codewords in its neighborhood, or because two nodes would have identical identifying sets. A minimal collection F_{hex} of such sets up to translation is shown in Figure 5.

From Lemma 6 we now obtain the following:

Lemma 7. *Given $\vec{v} \in T$, we can effectively compute the exact minimal density of an identifying code on E_{hex}, under the constraint that the code be $\vec{v}$-periodic.*

Most of the codes in the literature are found by looking at two periods, i.e. a period lattice of full rank. Both methods lead to the same upper bounds on the minimal density, since as shown in Lemma 6 and Lemma 4, the minimal density of a configuration with a fixed period lattice of rank-$(d-1)$ is actually reached by a configuration with period lattice of rank-d.

However, heuristically the new method should lead to a more efficient (or at least essentially different) algorithm, as we do not need to guess in advance how large the $\vec{v}$-directional period needs to be. And indeed, as reported in in Theorem 1, we were able to improve on the smallest density of an identifying code this way.

5 Implementation

By Lemma 7, we can effectively compute the minimal density of a $\vec{v}$-periodic identifying code on E_{hex}. Specifically, this reduces to computing the minimum mean weight cycle in a weighted directed graph $G = (V, E, \lambda)$, where $\lambda : E \to \mathbb{R}$ is the weight function. Let $(x, y) = \vec{v}$ and assume $y > 0$, $x + y$ even and $x \geq 0$. For $j \in \mathbb{Z}$, let $s_j = \lceil jx/y \rceil$, and define the *border* as $B = \{(s_j, j), (s_j + 1, j) \mid j \in \mathbb{Z}\}$. The nodes of our graph are sets C of translates of the sets F_{hex} that are invariant under translation by $\vec{v}$, meaning that $\{S + \vec{v} \mid S \in C\} = C$, and such that each $S \in C$ contains both a node (i, j) with $i \leq s_j + 1$ and a node (i, j) with $i \geq s_j$; we call this the *border condition*.

In our graph, we include an edge $(C, D) \in E$ whenever there exists a $\vec{v}$-periodic pattern $P \in \{0, 1\}^B$ with

$$D = \{S - (2, 0) \mid S \in C \cup C_B, \nexists \vec{n} \in S : P_{\vec{n}} = 1\}$$

where C_B contains all translates of sets in F_{hex} that intersect B and are contained within the half-plane $\{(i, j) \in \mathbb{Z}^2 \mid s_j \leq i\}$. The weight of such an edge is $\sum P$, or the number of 1-symbols in P. The idea is that a bi-infinite walk in G corresponds to a $\vec{v}$-periodic configuration specified by one (slanted) vertical strip – a translated version of B – at a time. An edge $(C, D) \in E$ corresponds

to an adjacent pair of such strips. The nodes maintain a collection of forbidden patterns that can be "dealt with" as soon as a 1-symbol, representing a codeword, is known to occur in them. If we have not been able to get rid of a forbidden pattern, it is shifted to the left. Patterns are forbidden from drifting to the left of the border B, as they would then become impossible to remove. Note that since the states are $\vec{v}$-periodic and contain only translates of sets in F_{hex} that satisfy the border condition, the number of possible states is finite. We construct our graphs by starting from the empty node $C_0 = \emptyset$ and including every node reachable from it. The resulting subgraph is strongly connected (as C_0 is reachable from every node by all-1 edges) and contains exactly the same labels of bi-infinite walks as the full graph.

Karp's algorithm [13] can be used to calculate the minimum mean weight of a cycle in the graph (G, E, λ). We chose this algorithm because it is easy to implement and readily parallelizable. We did not use any special tricks in the construction of the graph, apart from basic memory optimization and parallelization. Karp's algorithm was the bottleneck in all computations, and we used some tricks to be able to go further in the calculations. We explain these tricks in this section.

First, let us recall the basic Karp's algorithm for a weighted graph $G = (V, E, \lambda)$. The algorithm only works when the graph has an *initial node* s from which every other node is reachable. This can be guaranteed by passing to strongly connected components, but in our application G is always strongly connected. Now, for each node v and length $0 \leq k \leq n$, compute the minimal weight $F_k(v)$ of path from s to v. This can be done using the recurrence $F_k(v) = \min_{(w,v)\in E}(F_{k-1}(w) + \lambda(w, v))$. The minimum mean weight of a cycle is then

$$\alpha = \min_{v\in V} \max_{0\leq k<n} \frac{F_n(v) - F_k(v)}{n - k}. \tag{1}$$

This can be computed in space $O(n^2)$ where $n = |V|$ by storing the weights $F_k(v)$ in an $n \times (n + 1)$ array, or in space $O(n)$ by using a handful of length-n arrays: first, compute the weights $F_k(v)$ for $k = 0, 1, \ldots, n$ using two arrays, then store the weights $F_n(v)$ into a third array A, and finally recompute the weights $F_k(v)$ using two arrays while simultaneously computing (1) using the values of the array A and a fourth array that stores the running maximum of the fractions $(F_n(v)-F_k(v))/(n-k)$ for each $v \in V$. Both variants run in $O(n^2)$ time, but the latter takes approximately twice as long.

In order to actually compute a cycle with minimum mean weight, one takes a node $v^* \in V$ that attains the minimum in (1) and computes a length-n path P from s to v^* of minimum weight. It necessarily contains a minimum mean weight cycle as a contiguous sub-path, and it is easy to locate one knowing the value of α. The path P can be computed as part of the $O(n^2)$ space algorithm by storing, in addition to each value $F_k(v)$, a node w with $(w, v) \in E$ that minimizes $F_{k-1}(w) + \lambda(w, v)$, and following these nodes backward from v^* to s.

However, $O(n^2)$ space proved to be too high for the code of density 53/126 in Figure 4; we could prove its existence using the $O(n)$ variant, but not extract a concrete code, as the cluster ran out of memory. For this purpose we implemented a third variant that runs in space $O(n^{3/2})$ and takes about three times as long as the $O(n^2)$ variant:

1. Fix a starting node $s \in V$.

2. Pick $m = \lceil \sqrt{n} \rceil$ numbers $0 = n(1) < n(2) < \cdots < n(m) = n$ with $n(i+1) \leq n(i) + m$ for each i.

3. For each $k = 0, \ldots, n$, compute the weights $F_k(v)$ for $v \in V$ using two length-n arrays. If $k = n(i)$ for some i, then store these weights into one row of an $m \times n$ array A.

4. For each $k = 0, \ldots, n$, recompute the weights $F_k(v)$ and simultaneously compute α from (1) and the minimizing vertex v^*.

5. Set P as the path containing only v^*.

6. For each $i = m, m-1, \ldots, 2$, compute an $m \times n$ array B containing, for each $n(i-1) < k \leq n(i)$ and $v \in V$, the vertex w with $(w, v) \in E$ that minimizes $F_{k-1}(w) + \lambda(w, v)$. The values $F_{n(i-1)}(v)$ stored in A are used as a starting point. From this data, extract a length-m path Q from some $v \in V$ to the beginning of P that minimizes $F_{n(i-1)}(v) + \lambda(Q)$. Prepend it to P.

At the end of the algorithm, P is a length-n path from s to v^* of weight $F_n(v)$.

Acknowledgments

We thank Ville Junnila and Tuomo Lehtilä for suggesting problems to apply our method on. We thank Tim Austin for suggesting a reference for the pointwise ergodic theorems specific to the abelian setting. The computer resources of the Finnish IT Center for Science (CSC) and the FGCI project (Finland) are acknowledged.

References

[1] Irene Charon, Iiro Honkala, Olivier Hudry, and Antoine Lobstein. General bounds for identifying codes in some infinite regular graphs. *The electronic journal of combinatorics*, 8(1):R39, 2001.

[2] Irene Charon, Olivier Hudry, and Antoine Lobstein. Identifying codes with small radius in some infinite regular graphs. *The electronic journal of combinatorics*, 9(1):R11, 2002.

[3] Gérard D. Cohen, Iiro Honkala, Antoine Lobstein, and Gilles Zémor. Bounds for codes identifying vertices in the hexagonal grid. *SIAM Journal on Discrete Mathematics*, 13(4):492–504, 2000.

[4] Daniel W Cranston and Gexin Yu. A new lower bound on the density of vertex identifying codes for the infinite hexagonal grid. *arXiv preprint arXiv:1006.3779*, 2010.

[5] Ari Cukierman and Gexin Yu. New bounds on the minimum density of an identifying code for the infinite hexagonal grid. *Discrete Applied Mathematics*, 161(18):2910–2924, 2013.

[6] Manfred Denker, Christian Grillenberger, and Karl Sigmund. *Ergodic theory on compact spaces*. Lecture Notes in Mathematics, Vol. 527. Springer-Verlag, Berlin, 1976.

[7] Noam David Elkies. The still-life density problem and its generalizations. In *Voronoi's Impact on Modern Science, Book I*, volume 21 of *Proceedings of the Institute of Mathematics of the National Academy of Sciences of Ukraine*, pages 228–253. Institute of Mathematics of the National Academy of Sciences of Ukraine, 1998.

[8] Hans-Otto Georgii. *Gibbs measures and phase transitions*, volume 9 of *De Gruyter Studies in Mathematics*. Walter de Gruyter, Berlin, New York, 2011.

[9] Iiro Honkala and Antoine Lobstein. On the density of identifying codes in the square lattice. *Journal of Combinatorial Theory, Series B*, 85(2):297–306, 2002.

[10] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. PySAT: A Python toolkit for prototyping with SAT oracles. In O. Beyersdorff and C. Wintersteiger, editors, *Theory and Applications of Satisfiability Testing – SAT 2018*, volume 10929 of *Lecture Notes in Computer Science*, pages 428–437. Springer, Cham, 2018.

[11] Devin Jean and Suk Seo. Fault-tolerant locating-dominating sets on the infinite king grid. *arXiv preprint arXiv:2201.09399*, 2022.

[12] Ville Junnila. New lower bound for 2-identifying code in the square grid. *Discrete Applied Mathematics*, 161(13):2042–2051, 2013.

[13] Richard M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23(3):309–311, 1978.

[14] M.G. Karpovsky, K. Chakrabarty, and L.B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE Transactions on Information Theory*, 44(2):599–611, 1998.

[15] Samuel J. Lightwood. Morphisms from non-periodic $\mathbb{Z}^2$ subshifts I: constructing embeddings from homomorphisms. *Ergodic Theory and Dynamical Systems*, 23(2):587–609, 2003.

[16] Kenji Obata. On identifying codes in lattices. *Preprint, December*, 2003.

[17] Mikko Pelto. Optimal $(r, \leq 3)$-locating–dominating codes in the infinite king grid. *Discrete Applied Mathematics*, 161(16):2597–2603, 2013.

[18] Mikko Pelto. On locating-dominating codes in the infinite king grid. *Ars Combinatoria*, 124:353–363, 2016.

[19] Hartley Rogers Jr. *Theory of recursive functions and effective computability*. MIT press, 1987.

[20] Ville Salo and Ilkka Törmä. Hex automaton, 2023. GitHub repository.

[21] Peter J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci*, 22(4):445–455, 1988.

[22] Peter J. Slater. Fault-tolerant locating-dominating sets. *Discrete Mathematics*, 249(1):179–189, 2002. Combinatorics, Graph Theory, and Computing.

[23] Derrick Stolee. Automated discharging arguments for density problems in grids. *arXiv preprint arXiv:1409.5922*, 2014.

[24] Peter Walters. *An introduction to ergodic theory*, volume 79 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1982.