3-coloring of grids Let $d > 1$ and write $\mathbb{Z}^d$ for the $d$-dimensional oriented infinite grid. It was shown, independently, that the complexity of the 4-coloring problem is $O(\log^* n)$ in the LOCAL model [BHK+17], and, in our language, $O(\log^* 1/\varepsilon)$ in the uniform local model [HSW17]. That is, [HSW17] showed that there is a fflip that produces a 4-coloring and has such a tail decay of the coding radius.

Concerning the 3-coloring problem $\Pi$, [BHK+17] showed that in the local model the complexity is $\Omega(d\sqrt{n})$ and [HSW17] that $0 < M(\Pi) \leq 2$. As an application of the approach that combines both techniques we show the following.

**Theorem 0.1.** Let $\Pi$ be the proper vertex 3-coloring problem. Then we have

$$\Pi \in \text{ULOCAL}((1/\varepsilon)^{1+o(1)})$$

on the $d$-dimensional infinite grid (for any dimension $d > 1$). In particular, $M(\Pi) = 1$.

There are three steps in the proof, here (UB) and (LB) stands for upper and lower bound respectively:

- (UB) We borrow an approach from descriptive combinatorics called TOAST construction. Let $T$ be a hierarchy with boundaries far apart (called toast) and consider the algorithm that colors recursively as follows. It attempts to do 2-coloring but if there is a parity collision with previous decision, then it uses the third color to change the parity.

- (UB) Standard way to produce toast via Voronoi cells can be done in a finitary fashion with coding radius exactly $(1/\varepsilon)^{1+o(1)}$. Note that in the construction we need to use MIS with larger and larger parameter.

- (LB) To get the lower bound we show that a uniform algorithm of faster tail decay can be modified to show $\Pi \in \text{ULOCAL}(O(\log^* 1/\varepsilon))$. This would contradict the result of [HSW17] mentioned above.

As the proof of (UB) is somewhat standard, we focus only on (LB). It is enough to work on 2-dimensional grids. We show that any uniform algorithm $A$ solving 3-coloring that satisfies

$$P(R_A > C/\varepsilon) \leq \varepsilon,$$

for an absolute constant $C \approx 1000^{1000}$, can be turned into a deterministic local algorithm that solves 3-coloring on finite tori in $O(\log^* n)$ many rounds.

First step is to pick $r_0 \in \mathbb{N}$ large enough such that

$$P(R > r_0) \leq \frac{1}{Cr_0}$$

and tile $\mathbb{Z}^2$ (or finite tori) with boxes of side lengths in $\{r_0, r_0 + 1\}$. This is possible in $O(\log^* n)$, or $O(\log^* 1/\varepsilon)$, rounds. Moreover for $\Delta = 100$, we compute a $(\Delta + 1)$-coloring of these boxes, that has the property that if $B, B'$ are boxes of distance less than $3(r_0 + 2)$, then they are assigned different color. The fact that this is possible follows from the observation that for a fixed box $B$ we have that all the boxes of distance at most $3(r_0 + 2)$ are contained in a box of side length $10r_0$. Now a volume argument shows that there are not more than $(10r_0)^2/r_0^2 \leq 100$ many such boxes. By the basic theorem about $\Delta + 1$-colorings, this takes again $O(\log^* n)$, or $O(\log^* 1/\varepsilon)$, rounds. We denote this colored tiling problem as $\square_{r_0}$.

In the remaining part of the proof we describe a constant round rule $A'$ that solves 3-coloring given solution to $\square_{r_0}$ as an input. Inductively on the $\Delta + 1$-many color classes, we deterministically fix randomness in each box in such a way that:
• each vertex can compute its randomness in constant number of rounds independent of \(n\),

• there is \(R \in \mathbb{N}\) independent of \(n\) so that each vertex \(v\) on the boundary of the tiling needs to look to distance at most \(R\) to simulate \(A\) and output a solution,

• the solution given by \(A\) on the boundary can be extended to the interior of each tile.

Having this is clearly enough, as having a solution on the boundary allows to extend it inside each tile in constantly many rounds, once we know that such a solution exists.

Let \(\{B_i\}_{i \in J}\) together with a coloring \(B_i \mapsto c(B_i) \in [\Delta + 1]\) be the input solution of \(\Box_{r_0}\) Write \(V(B)\) for the vertices of \(B\) and \(\partial B\) for the vertices of \(B\) that have distance at most \(t\) from the boundary of \(B\). We have \(|\partial B| < 4 \cdot (r_0 + 1)\). Define \(N(B)\) to be the set of rectangles of distance at most \((r_0 + 2)\) from \(B\).

We are now going to describe how \(A'\) iterates over the boxes in the order of their colors and for each node \(u\) in the box \(B\) it carefully fixes a random string \(r(u)\). This is done in such a way that after fixing the randomness in all nodes, we can apply the original algorithm \(A\) to each vertex in \(\partial B\) for each \(B\) with the randomness given by \(r\) and the local algorithm at \(u \in \partial B\) will not need to see more than its \(r_0\)-hop neighborhood.

Given a partial function \(r_\ell : V(G) \to [0, 1]\) corresponding to nodes such that their randomness was already fixed, every box \(B\) can compute a probability \(P(\mathcal{E}_\ell(B))\), where \(\mathcal{E}_\ell(B)\) is the (bad) event that there is a vertex in \(\partial B\) that needs to look further than \(r_0\) to apply the uniform local algorithm \(A\). Note that for computing \(P(\mathcal{E}_\ell(B))\) it is enough to know the \(r_0\)-neighborhood of \(B\).

The algorithm \(A\) has \(\Delta + 1\) rounds. In the 0-th round, we set \(r_0\) to be the empty function. Note that the probability that a vertex needs to look further than \(r_0\) in the algorithm \(A\) is smaller than \(\frac{1}{C_{r_0}}\) by our choice of \(r_0\). The union bound gives \(P(\mathcal{E}_0(B)) < \frac{1}{(2\Delta)^{2\Delta + 1}}\) for every \(B\).

In each step \(\ell \in [\Delta + 1]\) we define \(r_B : V(B) \to [0, 1]\) for every \(B\) such that \(c(B) = \ell\) and put \(r_\ell\) to be the union of all \(r_D\) that have been defined so far, i.e., for \(D\) such that \(c(D) \leq \ell\). We make sure that the following is satisfied for every \(B\):

(a) the assignment \(r_B\) depends only on \(2(r_0 + t)\)-neighborhood of \(B\) (together with previously defined \(r_{\ell - 1}\) on that neighborhood) whenever \(c(B) = \ell\),

(b) \(P(\mathcal{E}_\ell(B)) < \frac{1}{(2\Delta)^{2\Delta + 1}}\),

(c) restriction of \(r_\ell\) to \(N(B)\) can be extend to the whole \(\mathbb{Z}^d\) such that \(A\) is defined at every vertex, where we view \(N(B)\) embedded into \(\mathbb{Z}^d\).

We show that this gives the required algorithm. It is easy to see that by (a) this defines a local construction that runs in constantly many rounds and produces a function \(r := r_{\Delta + 1} : V(G) \to [0, 1]\). By (b) every vertex in \(\partial B\) for some \(B\) looks into its \(r_0\) neighborhood and use \(A\) to choose its color according to \(r\). Finally, by (c) it is possible to assign colors in \(B \setminus \partial B\) for every \(B\) to satisfy the condition of \(\Pi\) within each \(B\). Note that as the thickness of the boundary is more than \(t\), this depends only on the output of \(A\) on the boundary. Picking minimal such solution for each \(B \setminus \partial B\) finishes the description of the algorithm.

We show how to proceed from \(0 \leq \ell < \Delta + 1\) to \(\ell + 1\). Suppose that \(r_\ell\) satisfies the conditions above. Pick \(B\) such that \(c(B) = \ell + 1\). Note that \(r_B\) that we want to find can influence rectangles \(D \in N(B)\) and these are influenced by the function \(r_\ell\) at distance at most \(r_0\) from them. Also given any \(D\) there is at most one \(B \in N(D)\) such that \(c(B) = \ell + 1\). We make the decision \(r_B\) based on \(2r_0\) neighborhood around \(B\) and the restriction of \(r_\ell\) there, this guarantees that condition (a) is satisfied.
Fix $D \in N(B)$ and pick $r_B$ uniformly at random. Write $\mathcal{G}_{B,D}$ for the random variable that computes the probability of $\mathcal{E}_{r_B \cup r_l}(D)$. We have $\mathbb{E}(\mathcal{G}_{B,D}) = \mathbb{P}(\mathcal{E}_l(D)) < \frac{1}{(2\Delta)^2 - \ell}$. By Markov inequality we have
\[
\mathbb{P}\left(\mathcal{G}_{B,D} \geq 2\Delta \frac{1}{(2\Delta)^2 - \ell}\right) < \frac{1}{2\Delta}.
\]
Since $|N(B)| \leq \Delta$ we have that with non-zero probability $r_B$ satisfies $\mathcal{G}_{B,D} < \frac{1}{(2\Delta)^2 - \ell}$ for every $D \in N(B)$. Since $A$ is defined almost everywhere we find $r_B$ that satisfies (b) and (c) above. This finishes the proof.

References
