

A factor of i.i.d. with uniform marginals and infinite clusters spanned by equal labels

PÉTER MESTER

Alfréd Rényi Institute of Mathematics, Budapest and Tomori Pál College, Budapest, Hungary

(e-mail: pmester@alumni.iu.edu)

(Received 20 October 2020 and accepted in revised form 19 January 2022)

Abstract. We give an example of an FIID vertex-labeling of \mathbb{T}_3 whose marginals are uniform on $[0, 1]$, and if we delete the edges between those vertices whose labels are different, then some of the remaining clusters are infinite. We also show that no such process can be finitary.

Key words: regular trees, hyperfiniteness, non-amenable groups, factor of i.i.d.
2020 Mathematics Subject Classification: 60G10, 82B43 (Primary); 37A50, 22F10, 05C78 (Secondary)

1. Introduction

In this paper, we answer a question asked by Gaboriau in his ICM survey in a slightly different language (see [G10, Question 5.6]). The question concerns factor of i.i.d. processes (FIIDs from now on), which are random elements of S^Γ where Γ is a countable group and S is some measurable space which we will call the *label set*, and arises by applying an equivariant measurable map $f : [0, 1]^\Gamma \rightarrow S^\Gamma$ to an i.i.d. family $\{\omega_\gamma\}_{\gamma \in \Gamma}$ where ω_γ is uniform on $[0, 1]$. In our work, the label set S will be $[0, 1]$ or $\{0, 1\}^n$.

To make the definition of being an FIID precise, we need to say what are the group actions on $[0, 1]^\Gamma$ and S^Γ . A group action $H \curvearrowright \Gamma$ of a group H on Γ naturally extends into a group action of H on M^Γ for any set M : for $x \in M^\Gamma$ and $h \in H$, the action $h \cdot x$ is defined as $(h \cdot x)(\gamma) = x(h^{-1} \cdot \gamma)$ for $\gamma \in \Gamma$. Then we will call f equivariant with respect to this action if it commutes with it. Although it is with respect to the group action itself, not only with respect to H , if in the context it is clear which action we mean, we may say simply that f is H -equivariant. For us, there are two natural group actions on Γ . One is when $H = \Gamma$ and it acts on itself by left multiplication. For the other one, we assume that a Cayley graph of Γ is given considered here as a simple graph on the vertex set $V(G) = \Gamma$, forgetting the orientations and labels on the edges by the group generators), so that the full graph automorphism group $\text{Aut}(G)$ also acts on Γ . As $\text{Aut}(G)$ also acts on the edge set $E(G)$ of G , it makes sense to talk about equivariant measurable maps $f : [0, 1]^\Gamma \rightarrow \{0, 1\}^{E(G)}$ which we will call FIID subgraphs of G . Note that $\Gamma < \text{Aut}(G)$ naturally, so for any f , the property of being $\text{Aut}(G)$ -equivariant is stronger than just being Γ -equivariant. The process

we will construct will be $\text{Aut}(G)$ -equivariant, and consequently also Γ -equivariant. For us, the group itself is relevant as the vertex set of the graph G which, in our work, will be the 3-regular tree \mathbb{T}_3 . For example, we can consider it to be the Cayley graph of the free product of either \mathbb{Z} with \mathbb{Z}_2 or that of the threefold free product of \mathbb{Z}_2 with itself.

An S^Γ -valued FIID process also belongs to the larger class of *invariant* processes which means that the distribution is invariant under the group action in the following sense: if A is any event (in this case, some measurable subset of S^Γ), $h \in H$, then $\mathbf{P}(A) = \mathbf{P}(h \cdot A)$. This is again a notion relative to some group action and while our process will be $\text{Aut}(G)$ invariant, some important properties (where we use the mass-transport principle) will already follow from the weaker fact that the process is Γ -invariant.

If $\Theta = f(\omega)$ is an FIID and $\gamma \in \Gamma$, then $\Theta(\gamma)$ is a random variable whose distribution does not depend on γ by the transitivity of the group action with respect to which f is equivariant, and this distribution will be called the *marginal* of Θ . Note that the marginal is also well defined for any invariant process when the underlying group action is transitive as well.

If $\mathcal{L} \in S^V$ is a labeling of the vertex set V of a graph, then let $\text{clust}(\mathcal{L})$ be the subgraph obtained by deleting any edge whose endpoints got different labels by \mathcal{L} . The connected components of $\text{clust}(\mathcal{L})$ will be called the *clusters* of the labeling.

Gaboriau asked if assuming that the marginals of an FIID on a Cayley graph are uniform on $[0, 1]$ implies that the corresponding clusters are finite. We will show by an example on the 3-regular tree \mathbb{T}_3 that the answer is *no*.

THEOREM 1.1. *There is an FIID labeling Θ of \mathbb{T}_3 whose marginals are uniform on $[0, 1]$ for which $\text{clust}(\Theta)$ contains infinite clusters. Moreover, it is possible to modify Θ into a new labeling Θ^∞ whose marginals are still uniform on $[0, 1]$ and every cluster in $\text{clust}(\Theta^\infty)$ is infinite.*

Actually, \mathbb{T}_3 can be replaced by any non-amenable Cayley graph as [GL09] has shown that they always contain an FIID spanning forest whose components have furcation vertices (defined in §4), and that is the only thing our construction needs.

It is natural to think of FIIDs as the class of random labelings of the vertex set which can be obtained by applying a relabeling algorithm (f in the definition) which does not use any further randomness beyond its input (ω in the definition). For this reason, we sometimes will refer to f as the *code* of the FIID and to ω as its *source*. The measurability of f implies that we can approximate the label $f(\omega)(v)$ from local data with high probability. That is, for any $\epsilon_1, \epsilon_2 > 0$, there is an $r \in \mathbb{N}$ and a $g : [0, 1]^{B(v,r)} \rightarrow [0, 1]$ which ‘guesses’ the value of $f(\omega)(v)$ from the restriction of ω to $B(v, r)$ such that

$$\mathbf{P}\{\omega : |f(\omega)(v) - g(\omega \upharpoonright_{B(v,r)})| < \epsilon_1\} > 1 - \epsilon_2.$$

We will not use this locality property explicitly (we sketch its proof in Remark 1.2). For example, if $f(\omega)(v)$ is the indicator of v being in an infinite cluster of a labeling obtained from ω , then of course f is measurable and $f(\omega)(v)$ could be approximated in the above sense, but we will not need that detour (a similar remark holds for furcation vertices introduced later).

The reason we still emphasize this locality is twofold. First, it provides a fundamental intuition which also helps to appreciate the difficulties in constructing a process as an FIID. In particular, for far away vertices, these approximations are independent and this makes some properties hard to satisfy. Second, there are stronger notions of locality which do not follow from measurability alone, one of them is being *finitary* (which holds if there is a random but finite neighborhood of any vertex so that it is enough to see the source in that neighborhood to determine the label of the vertex), and we will see in §6 that no finitary process can be a counterexample to Gaboriau’s question.

Remark 1.2. This locality property is a simple consequence of the fact that the product σ -algebra on $[0, 1]^{V(\mathbb{T}_3)}$ is generated by the algebra of local events (those determined by finitely many coordinates). If $(\Omega, \mathcal{B}, \mathbf{P})$ is a probability space and \mathcal{A} is an algebra generating the σ -algebra \mathcal{B} , then for any $E \in \mathcal{B}$ and $\epsilon > 0$, there is an $F \in \mathcal{A}$ such that $\mathbf{P}(E \Delta F) < \epsilon$ (to prove this, notice that the collection of sets which can be approximated by elements of \mathcal{A} in this sense forms a σ -algebra).

Thus, for any event $E \subset [0, 1]^{V(\mathbb{T}_3)}$ and $\epsilon > 0$, there is a local event F such that $\mathbf{P}(E \Delta F) < \epsilon$. That is, one can guess whether or not E holds based on whether or not F holds with a probability of error less than ϵ . We can use this to approximate the value $f(\omega)(v)$ with high probability if we apply it for events in the form $a < f(\omega)(v) < b$ for $a, b \in [0, 1]$ (this is where the measurability of f is used). That is, we can figure out with high probability which subinterval of a finite subdivision of $[0, 1]$ contains $f(\omega)(v)$.

FIID processes are the closest to i.i.d. processes, hence there is an obvious interest in deciding if a given naturally defined process is FIID, and what properties FIID processes can or must have. These problems have a rich history in ergodic theory, probability, statistical physics, and theoretical computer science; see, e.g., [B18, L17, NSZ20, OW87, RV17, S10, S20, vdBS99] and the references therein. On amenable groups, Ornstein–Weiss theory [OW87] says that every FIID process is actually isomorphic to an i.i.d. process of the same entropy. Moreover, hyperfiniteness easily implies that every process is a weak limit of FIID processes. However, on non-amenable groups, these tools break down.

In general, it seems hard to decide whether or not an FIID process with a given property exists. The locality mentioned above suggests that the answer to Gaboriau’s question should be affirmative since the conditions imply that the vertices build a label whose specific values have probability zero, through a process governed by local data, so it seems surprising that vertices arbitrarily far away can agree on the same value, yielding infinite clusters with the same label. There is, in particular, a strong correlation decay found by [BSzV15] which states that if u, v are vertices of \mathbb{T}_d at distance k , then the absolute value of the correlation between their labels given by an FIID process is at most $(k + 1 - 2k/d)(1/\sqrt{d-1})^k$. This can be an obstruction for some processes to be FIID which might be seen as a quantitative version of this locality. While we will show that the condition of having uniform marginals does not exclude infinite clusters, there is an intuition that the condition should imply that at least the clusters are *small* in some sense. And indeed, [Ch10, Theorem 1.1] implies that the clusters under the condition that the marginals are uniform on $[0, 1]$ must be *hyperfinitely*. In our context, hyperfiniteness of the forest $\text{clust}(\Theta)$ means that there is a sequence of random invariant forests $\{F_i\}_{i \in \mathbb{N}}$

such that every component of F_i is finite almost surely, and (using the notation $F(v)$ for the component of F containing the vertex v in a forest F) $F_i(v) \subset F_j(v)$ for $i \leq j$ and $\text{clust}(\Theta)(v) = \bigcup_{i \in \mathbb{N}} F_i(v)$ for each vertex v . While it is true for any countable set that it is an increasing union of finite sets, the extra requirement of achieving this with random invariant processes makes hyperfiniteness a strong property. It implies that the clusters of $\text{clust}(\Theta)$ must be finite or must have 1 or 2 ends (see [BLPS99]), where the number of ends of a graph is the supremum of the number of infinite components when an arbitrary finite set of vertices (together with the incident edges) is deleted. It is well known that if F is an invariant random subforest of a Cayley graph, then an infinite component of F can only have 1, 2, or infinitely many ends.

The problem mentioned in the following example will not be pursued in this paper, but the simple construction shown there serves as a quick illustration of this locality phenomenon of FIIDs contrasted with invariant processes in general. It also helps to see through a natural optimization problem that being an FIID is a serious restriction among invariant processes since the optimum when we allow to use any invariant process is $1/2$ on \mathbb{T}_d while if we only allow FIIDs, the optimum is going to zero as d is going to ∞ .

Example 1.3. Assume we want an invariant random vertex set S of \mathbb{T}_d which is *independent* in the graph theoretic sense (that is, S does not contain neighbors). We want the marginal probability p_S of being in S to be high. The optimal $p_S = \frac{1}{2}$ is achieved by the following random invariant process: let $v_1, v_2 \in V(\mathbb{T}_d)$ be equivalent if their distance is even, and pick one of the classes with probability $\frac{1}{2}$ to be S . If we want S to be a FIID (that is, its indicator 1_S to be FIID), then a possible solution is the following code: at a vertex v , let $f(v) := 1$ if and only if for each neighbor w of v , $\omega(v) > \omega(w)$ (where ω was the original source) and let $f(v) = 0$ otherwise, then $S = \{f(v) = 1 : v \in V(\mathbb{T}_d)\}$ is an independent set. In this case, $p_S = 1/(d + 1)$.

While this simple construction can be improved, its basic features are known to hold even for a near-optimal independent set arising as an FIID. Namely, the marginal of any FIID independent vertex set of $V(\mathbb{T}_d)$ is bounded away from $\frac{1}{2}$ for any $d \geq 3$ and it goes to zero as $d \rightarrow \infty$. This follows easily from [B81], although, its focus is on finite graphs. For the connection with FIIDs and references to further research in this direction, see [RV17].

An example similar to the above will be used in our construction.

Example 1.4. Let r be a positive integer and let S be the FIID vertex set defined by the following code (which will be the indicator of S): let $1_S(v) = 1$ if and only if the label $\omega(v)$ of v is maximal of all the labels within the ball of radius r around v (otherwise $1_S(v) = 0$). Then, S has the property that any two vertices of it have distance at least r .

We will need this construction not only directly on \mathbb{T}_3 but also on some locally finite forests associated to it. Then, ‘distance’ will refer to the distance within the forest, which is infinite between vertices of different components. These forests will be random, and S can be sampled independently of it. In this way, S is guaranteed to intersect all infinite component of these forests, as there are only countably many components.

We close this section by an important elementary observation and some remarks on the organization of the paper.

By a $\text{un}[0, 1]$ random variable, we mean one which is uniform on $[0, 1]$. By $x \stackrel{d}{\sim} y$, we mean that the random variable x and y has the same distribution, but with some abuse of notation, we will also denote by $x \stackrel{d}{\sim} \nu$ if x has distribution ν . From a single $x \stackrel{d}{\sim} \text{un}[0, 1]$, we can obtain an i.i.d. family of infinitely many $x_j \stackrel{d}{\sim} \text{un}[0, 1]$ by reorganizing the bits of x . Using this, when we describe the code f , we can assume that it can always reach out for an additional $\text{un}[0, 1]$ random variable independent of any other step of the algorithm. However, importantly, every single random variable the algorithm uses is local in the sense that it must belong to some vertex.

In §2, we develop some tools whose purpose will be to imitate a large degree tree within a small degree one by collecting the vertices into finite connected ‘bags’ (which we will call ‘cells’), whose neighbors will be the other bags. With these tools ready, we will give a high level overview of the whole construction in §3. The first point is that Bernoulli $(1/2)$ percolation on a rooted tree with large degrees will leave the root in an infinite cluster with high probability. This will reduce the degrees of the tree, but this can be counterbalanced by using larger and larger bags—the tools coming from §2. To make this work, we will need special sorts of vertices called furcations, and we have to make sure that we get enough of them using our process—this will be the main content of §4. In §5, we put together everything to prove our main theorem. We also explain directly why the clusters of our labelings are hyperfinite. Finally, in §6, which is largely independent from the rest of the paper, we show that no counterexample to Gaboriau’s question can be finitary. Because it turns out that we could prove our theorem from quite basic results, we made an effort to stay self-contained.

2. Voronoi partitions and other forests

In this section, we define some FID subgraphs which will always be forests since we are working on \mathbb{T}_3 . We have already defined the forest $\text{clust}(L)$ corresponding to a labeling L .

We now define Voronoi partitions. If S is a vertex set, we want to partition all the other vertices into classes according to the closest element of S . We have to deal with the potential ambiguity if a vertex v is at an equal distance from several elements of S , moreover, we want to make the partition classes connected.

Definition 2.1. Let F be any locally finite forest and let $S \subset V(F)$ and a collection of distinct real numbers $\{\alpha(v)\}_{v \in S}$ be given. If $v \in V(F)$, and the F -component which contains v also contains some element from S , then let $S_v \subset S$ be the set of those elements of S which are closest to v , that is, $S_v := \{s : d_F(v, s) = d_F(v, S)\}$. Let $\phi_{(S, \alpha)}(v) := s_0$ be that element of S_v for which $\alpha(s_0)$ is minimal (by the local finiteness of F , S_v is finite). Let two vertices v_1, v_2 be equivalent if $\phi_{(S, \alpha)}(v_1) = \phi_{(S, \alpha)}(v_2)$. If the F -component of v does not contain any element from S , then let the equivalence class of v be the singleton $\{v\}$. Let $\text{Vor}(S, \alpha)$ be the partition corresponding to this equivalence.

The role of the α is to handle the ambiguity if $|S_v| > 1$ and in this way, the partition classes are indeed connected. Note that one can break the tie in many other ways, but it is not automatically true that the partition classes are connected. For example, if each vertex

v chose to be equivalent with a uniform random element from S_v and different vertices did it independently of each other, then the partition classes do not have to be connected.

When we use Voronoi partitions, the forest will be in the form of $\text{clust}(L)$ or something closely related, the S will be an FIID set and the α will be extracted from the source. We will suppress α in the notation and just denote the partition by $\text{Vor}(S)$. We always assume that the hidden α is independent of any other steps of the construction. We will refer to partition classes as *cells* and we will use this terminology in general where we have a forest where every component is finite almost surely. We will see that our Voronoi partitions have this property. When we consider a Voronoi partition as a forest, we mean that we delete edges between vertices of different cells and forget the distinguished vertex; in this way, they are FIID forests.

If we want to produce an FIID labeling with a $\text{un}[0, 1]$ marginal whose clusters are finite but arbitrarily large, that is easy. We can even sample an arbitrary random forest whose components are almost surely finite, and label the vertices independently afterward.

LEMMA 2.2. *Let Π be an FIID forest whose components are almost surely finite. There is an FIID labeling θ with a $\text{un}[0, 1]$ marginal which is constant over each component of Π (in fact, almost surely $\text{clust}(\theta) = \Pi$), and the θ -labels of different components of Π form an independent family.*

Proof. Let $(\alpha(v), \beta(v))_{v \in V(T)}$ be a collection of two independent $\text{un}[0, 1]$ label over each vertex. For a vertex v , let $\Pi(v)$ be the component of Π containing v . Since almost surely $|\Pi(v)| < \infty$ and the $\beta(v)$ labels are all distinct, there will be a unique $v_0 \in \Pi(v)$ for which $\beta(v_0)$ is minimal within $\Pi(v)$ (that is, $\beta(v_0) = \min\{\beta(w); w \in \Pi(v)\}$). Then let v ‘copy’ the α label from v_0 , meaning that $\theta(v) := \alpha(v_0)$. \square

The finiteness of the components above was crucial; when we construct the infinite clusters with uniform labels, then the labels and the clusters will be built together step-by-step and not by selecting the infinite clusters first and labeling them afterward.

We will call a random forest whose components are almost surely finite a *cell-partition* and the components will be called *cells* or Π -cells where the forest is denoted by Π . If Π is an FIID cell-partition, then let $\text{Ber}(\Pi)$ be the FIID $\{0, 1\}$ -labeling λ of the vertex set with the properties that: its marginals are fair bits ($\mathbf{P}(\lambda(v) = 1) = \mathbf{P}(\lambda(v) = 0) = \frac{1}{2}$), the labels are constant over a Π -cell ($\Pi(v_1) = \Pi(v_2)$ implies $\lambda(v_1) = \lambda(v_2)$), and the labels over different cells are independent. By the notation $\lambda \sim \text{Ber}(\Pi)$, we will mean that first Π is sampled, and then we sample λ given Π . We will use $\text{Ber}(\Pi)$ to imitate a Bernoulli percolation on a graph whose vertices are the cells of Π . The fact that a labeling with this distribution can be realized as an FIID labeling is a consequence of the previous lemma: the fair bits needed for $\text{Ber}(\Pi)$ can be obtained from the $\text{un}[0, 1]$ -label guaranteed by the lemma, for example, by defining the bit to be 1 if $\theta > \frac{1}{2}$ and 0 if $\theta \leq \frac{1}{2}$.

One may notice that for a Voronoi-type partition $\text{Vor}(S)$, we do not need to know the finiteness of the cells to label them as claimed in the lemma, since each cell already comes with a single distinguished vertex (the one from S), and the whole cell can copy labels from this distinguished vertex just as in the proof.

So, if a Voronoi partition had non-zero chance of producing infinite clusters, then that already would witness the truth of our Theorem 1.1. However, a simple application of the mass-transport principle shows that every Voronoi cell must be finite. See [LP16, Ch. 8] for more on the mass-transport principle and note that the application we need here is also covered there directly as Example 8.6. For completeness, we also explain this here.

We will show that an invariant process on a Cayley graph cannot determine infinite components with a single distinguished vertex; the precise statement is Lemma 2.3 below. Note that this immediately implies that this is also true if the word ‘single’ is replaced by ‘finitely many’, as from the finitely many vertices we can select a uniform one and this still will be an invariant process if the original one was.

Let G be a Cayley graph of a countable group Γ , we will use $\Omega := \{0, 1\}^{V(G) \cup E(G)}$ to represent various vertex and edge configurations of G . Note that Γ acts on Ω . We interpret the edges with label zero as deleted, the edges with label 1 as kept, and vertices with label 1 as distinguished (and the vertices of label 0 are not), and thus when we talk about a component, we mean that the underlying graph has all the original vertices (distinguished or not), but only the edges which are kept. Thus, we can represent the event that ‘in every component, there is at most one distinguished vertex and there is an infinite component which does contain a distinguished vertex’ as a measurable subset $\mathcal{E} \subset \Omega$.

We have not specified a probability measure on Ω , but now we will say something about all Γ -invariant ones.

LEMMA 2.3. *If \mathcal{P} is any Γ -invariant probability measure on Ω , then the event \mathcal{E} cannot happen, that is*

$$\mathcal{P}(\mathcal{E}) = 0.$$

To prove this, we will need the mass-transport principle for countable groups, which is the following.

PROPOSITION 2.4. *Let Γ be a countable group and o be its identity. If $f : \Gamma \times \Gamma \rightarrow [0, 1]$ is diagonally invariant (meaning that $f(\gamma x, \gamma y) = f(x, y)$ for all $\gamma, x, y \in \Gamma$), then*

$$\sum_{x \in \Gamma} f(o, x) = \sum_{x \in \Gamma} f(x, o).$$

Proof. By f being diagonally invariant, $f(o, x) = f(x^{-1}o, x^{-1}x) = f(x^{-1}, o)$, so $\sum_{x \in \Gamma} f(o, x) = \sum_{x \in \Gamma} f(x^{-1}, o)$. However, inversion is a bijection, so the summation $\sum_{x \in \Gamma} f(x^{-1}, o)$ runs over the same non-negative terms as $\sum_{x \in \Gamma} f(x^{-1}, o)$. \square

Proof of Lemma 2.3. Assume that there exists a Γ -invariant \mathcal{P} probability measure on Ω for which $\mathcal{P}(\mathcal{E}) > 0$. Define the measurable function $F : \Gamma \times \Gamma \times \Omega \rightarrow [0, \infty]$ by: $F(x, y, \omega) := 1$ if x and y are within the same component and x is the distinguished vertex of that component in the configuration ω , $F(x, y, \omega) := 0$ otherwise. This function is invariant under $\text{Aut}(G)$, consequently also under Γ , in the sense that $F(\alpha \cdot x, \alpha \cdot y, \alpha \cdot \omega)$ for any α from either of $\text{Aut}(G)$ or Γ . Here, $F(x, y, \omega)$ is called *the mass sent by x to y* or *the mass received by y from x* in the configuration ω . The Γ -invariance of \mathcal{P} and F implies

that $f(x, y) := \mathbf{E}_{\mathcal{P}} F(x, y, *)$ is diagonally invariant, so Proposition 2.4 implies that for $o \in \Gamma$, the expected overall mass it receives is the same as the expected overall mass it sends out.

The expected mass the origin receives is no more than 1 (this is true even pointwise), but the expected mass it sends out is infinite if $\mathcal{P}(\mathcal{E}) > 0$; the origin would actually send out infinite mass with positive probability, not just in expectation. This contradiction shows that such invariant measure \mathcal{P} is not possible. \square

Recall that if Π is a forest, then for a vertex v , we denoted by $\Pi(v)$ the component of v . If two forests P, F are related in a way that $P(v) \subset F(v)$ for all v , then we denote this relationship by $P < F$ or $F > P$. To such a pair, we associate a new forest.

Definition 2.5. If F, P are forests on the same vertex set and $P < F$, then we associate to this pair a new forest F/P , called the large-scale forest (or when F is a tree, the large-scale tree). The vertices of F/P are the components of P , and two P -components t_1, t_2 are connected in F/P if their distance is 1 in F . For a vertex v , let $F/P(v)$ be the subtree of F/P which contains $P(v)$.

When we use this large scale forest construction, the P -components will be finite (so P is a cell-partition). If there is a further cell-partition Π on F/P , then there is a natural corresponding cell-partition $\text{glue}_{\Pi}(P)$ on F so that $P < \text{glue}_{\Pi}(P) < F$. We just glue together the cells of P according to Π , meaning that if \mathcal{C} is a Π -cell consisting of the P -cells C_1, \dots, C_l , then $\bigcup\{C_i : C_i \in \mathcal{C}\}$ will be a $\text{glue}_{\Pi}(P)$ -cell and, as these cells already partition all the vertices of F , defines $\text{glue}_{\Pi}(P)$.

When F, P are IID subforests of \mathbb{T}_3 and $P < F$ and P is a cell-partition, then by Lemma 2.2, we can assume that the vertices of F/P are equipped with a family of i.i.d. random variables $x(v)_{v \in V(F/P)} \stackrel{d}{\sim} \text{un}[0, 1]$ which we can use to build Voronoi-type partitions on F/P as an IID-forest on the original \mathbb{T}_3 (corresponding to each other via our glue operation).

3. High level overview

We will use the basic theory of Bernoulli percolation on trees, see [LP16] or [P19]. In fact, we will only need some very basic result from which we can directly obtain everything that we need so we made some effort to stay self-contained. In addition to this, we will need at the end the fact that an IID is ergodic, thus, when we prove that a vertex is contained in an infinite cluster with positive probability, we can conclude that there are infinite clusters with probability one.

A Bernoulli- p site percolation on a graph is the $\{0, 1\}$ -labeling where a vertex gets label 1 with probability p and 0 with probability $1 - p$, independently of the others. We will denote this labeling by $\text{Ber}(p)$.

Remark 3.1. In percolation theory, the interpretation is that with probability p , we keep a vertex (or site) *open* and they become *closed* with probability $(1 - p)$, independently from each other. There is a very useful notion called the *branching number* $\text{br}(T)$ of a locally finite infinite tree T with the property that $\text{br}(\mathbb{T}_d) = d - 1$ and a $\text{Ber}(p)$ labeling

on T almost surely has infinite open clusters exactly if $p > 1/\text{br}(T)$. Moreover, when this condition holds and if T is homogeneous enough, for example, \mathbb{T}_d , then the infinite open clusters will have branching number exactly $p\text{br}(T)$. So the big picture is that what guarantees infinite clusters in a Bernoulli percolation is a high enough branching number and the Bernoulli percolation will push down the branching number of the resulting clusters and this pattern continues if we keep applying independent Bernoulli percolations until there are no more infinite clusters.

We will need the following proposition.

PROPOSITION 3.2. *On the 4-regular tree \mathbb{T}_4 , a $\text{Ber}(\frac{1}{2})$ labeling almost surely has infinite clusters.*

Although we will work on more general trees, the above will be sufficient, through the fact that if a tree T has minimal degree $\delta := d_{\min}(T)$, then T contains \mathbb{T}_δ as a subgraph.

The above result has the easy consequence, which we will prove as Lemma 4.4, that for any $\epsilon > 0$, there exists a $D(\epsilon) \in \mathbb{N}$ such that if a tree T with a distinguished root $r \in V(T)$ has minimal degree at least $D(\epsilon)$, then a $\text{Ber}(\frac{1}{2})$ labeling on T will have an infinite cluster containing r with probability at least $(1 - \epsilon)$. In fact, we will need a property which is stronger than being infinite: we need *forking* clusters. In the process that we are going to define, the infinite clusters will automatically be forking; however, we will not prove this, since for our construction to work, it will be enough to know that forking clusters exists with positive probability, and thus some of the infinite clusters will be forking by ergodicity. We will prove forking clusters exists with positive probability through Lemma 4.5 directly from Proposition 3.2.

To highlight the ideas of the construction, we first show the modest claim that for any positive integer n , there is an FIID labeling θ_n on \mathbb{T}_3 whose marginal is uniform on the label set $\{0, 1\}^n$ and $\text{clust}(\theta_n)$ contains infinite clusters.

On the d -regular tree \mathbb{T}_d , a $\text{Ber}(p)$ labeling almost surely has infinite clusters spanned by vertices of label 1 exactly when $p > 1/(d - 1)$. This implies that if a tree T has minimal degree at least δ and if $p(\delta - 1) > 1$, then the clusters of a $\text{Ber}(p)$ -labeling on T will contain infinite ones labeled with 1.

If L_0, \dots, L_{n-1} are independent $\text{Ber}(\frac{1}{2})$ -labelings, and we concatenate them to get the $\{0, 1\}^n$ -labeling $\mathcal{L}_n := (L_0, \dots, L_{n-1})$, then for any $s \in \{0, 1\}^n$, the distribution of vertices whose \mathcal{L}_n -label is s will be the same as the distribution of vertices whose label is 1 in a single $\text{Ber}(1/2^n)$ -labeling. In particular, there will be infinite clusters in $\text{clust}(\mathcal{L}_n)$ on \mathbb{T}_d if $d > 2^n + 1$.

This is not yet the θ_n we promised, as we want to label \mathbb{T}_3 instead of \mathbb{T}_d where d depends on n . However, we can imitate a tree whose minimal degree is at least d within \mathbb{T}_3 using Voronoi partitions and the large scale tree construction. As in Example 1.4, let S be an FIID vertex set in \mathbb{T}_3 for which any two $v_1, v_2 \in S$ have distance at least $2r + 1$, then let $\Pi_0 := \text{Vor}(S)$. Each Π_0 -cell \mathcal{C} contains the ball $B_{\mathbb{T}_3}(v, r)$ around $v \in S \cap \mathcal{C}$. This implies that the large-scale tree \mathbb{T}_3/Π_0 has minimal degree at least $|B_{\mathbb{T}_3}(v, r)| + 2$.

For large enough r , the random tree \mathbb{T}_3/Π_0 has minimal degree at least $2^n + 2$, so using the FIID labeling $\text{Ber}(\Pi_0)$ guaranteed by Lemma 2.2 and concatenating n independent

versions $\lambda_i \sim \text{Ber}(\Pi_0)$ to form θ_n , we get an $\text{FID}\{0, 1\}^n$ -labeling which has infinite clusters and marginals uniform on $\{0, 1\}^n$.

This proves the claim, but how do we get a labeling which has uniform marginals on $[0, 1]$ but still with infinite clusters?

Instead of the ‘static’ sequence $\Pi_0, \dots, \Pi_0, \dots$, with the i.i.d. labels $\lambda_i \sim \text{Ber}(\Pi_0)$, we use a dynamically changing sequence of cell-partitions $\Pi_0 < \dots < \Pi_n < \dots$, and the corresponding sequence of labelings $\Lambda_0 \sim \text{Ber}(\Pi_0), \dots, \Lambda_n \sim \text{Ber}(\Pi_n), \dots$. This way, we will be able to use infinitely many bits and thus get $\text{un}[0, 1]$ marginals, while also having infinite clusters. The essence of how this sequence is constructed and what issues need to be taken care of is already visible in the step from Π_0 to Π_1 .

There will be ‘target degrees’ D_0 and D_1 which, for now, are just large integers.

A target degree was also present in the previous θ_n construction in the fact that \mathbb{T}_3/Π_0 had minimal degree at least $2^n + 2$ and Π_0 in the final construction is defined just as before; this time, we want \mathbb{T}_3/Π_0 to have minimal degree at least D_0 . To Π_0 , we associate the labeling $\Lambda_0 \sim \text{Ber}(\Pi_0)$ and we get the random forest $\mathcal{F}_0 := \text{clust}(\Lambda_0)$, where of course $\Pi_0 < \mathcal{F}_0$.

We want to build Π_1 in such a way that $\Pi_0 < \Pi_1 < \mathcal{F}_0$ and, ‘whenever possible’, the components of the large-scale forest \mathcal{F}_0/Π_1 should have minimal degree at least D_1 . So the goal in this second step is similar to that in the first step, when we wanted \mathbb{T}_3/Π_0 to have minimal degree at least D_0 .

A key difference is that in the first step, we worked with the deterministic tree \mathbb{T}_3 , while now we have to deal with the random forest \mathcal{F}_0 . We can immediately see that the target degree goal cannot be reached for all components of \mathcal{F}_0/Π_1 , as \mathcal{F}_0/Π_1 contains finite clusters, whose minimal degree is always 1, because a finite tree always has leaves. What about the infinite clusters? If there was a bi-infinite path among them, then for that component, the target degree goal cannot be achieved for $D_1 > 2$.

A key concept is a *furcation vertex* discussed in more detail in the next section. These are the vertices whose removal splits their component into a forest containing at least three infinite components. For example, a bi-infinite path contains no furcations at all. We will see that simply because \mathcal{F}_0 is defined by an invariant random process, as soon as a component contains a furcation at all, it contains enough to reach our target degree goal. Invariance alone does not guarantee that the infinite clusters have furcations, as an invariant process could split \mathbb{T}_3 into bi-infinite paths. Our process imitates Bernoulli percolations, and we will see that the existence of furcations is guaranteed. These concerns will be taken care of in the next section.

Now we can clarify for which components of the \mathcal{F}_0/Π_1 we can achieve the target degree goal: those containing furcations, which will be called *forking* components. We sketch what we do within a forking component. Of course, we need to do something with the other components as well, but that is rather arbitrary.

If v is a furcation of $\mathcal{F}_0(v)$, then by the finiteness of the Π_0 cells, $\Pi_0(v)$ will also be a furcation of $\mathcal{F}_0/\Pi_0(v)$. Conversely, if $\Pi_0(v)$ is a furcation of $\mathcal{F}_0/\Pi_0(v)$, then there is a vertex $v_0 \in \Pi_0(v)$ which is a furcation of $\mathcal{F}_0(v)$. Let S be the collection of furcations of \mathcal{F}_0/Π_0 and take the associated Voronoi partition $\text{Vor}(S)$. This partition lives on the large-scale forest \mathcal{F}_0/Π_0 , but it gives a natural way to glue together the corresponding

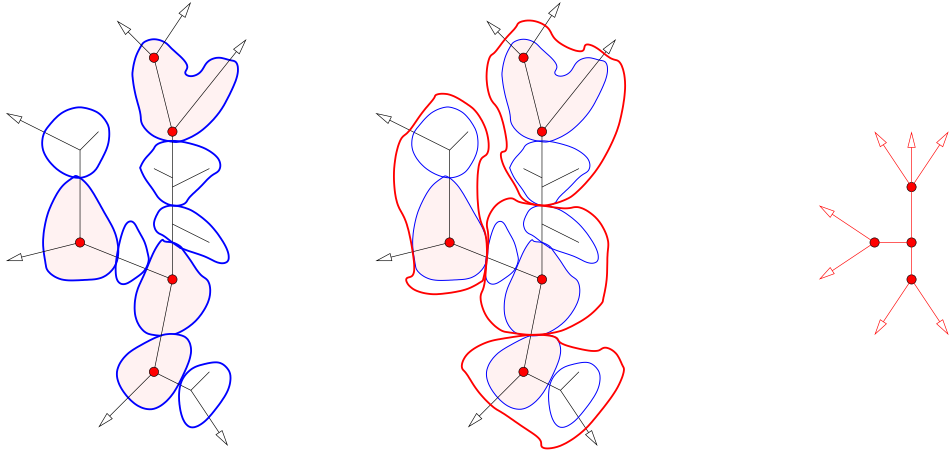


FIGURE 1. First, a forking tree $\mathcal{F}(v)$, together with a cell-partition Π . Arrows represent infinite parts of the tree, furcation vertices are denoted by solid dots. Furcation Π -cells are shaded. Second, the partition $\Pi^{\text{there exists furc}} \succ \Pi$ is also shown. Third, the large-scale tree $\mathcal{F}(v)/\Pi^{\text{there exists furc}}$.

Π_0 cells into $\Pi^{\text{there exists furc}} := \text{glue}_{\text{Vor}(S)}(\Pi_0)$. In $\Pi^{\text{there exists furc}}$, every cell within a forking component contains some furcation, and the cells of $\Pi^{\text{there exists furc}}$ remain finite by the mass-transport principle. See Figure 1. This finiteness implies that in the component $\mathcal{F}_0/\Pi^{\text{there exists furc}}(v)$, every degree will be at least 3 (see what could go wrong if the cells were not finite in Example 4.2). Now that all the degrees are at least 3, we can repeat the same idea that we used in \mathbb{T}_3 to achieve an arbitrarily high target degree, and this is how we get the new partition Π_1 with the properties that $\Pi_0 \prec \Pi_1 \prec \mathcal{F}_0$ and that the forking components of the large-scale forest \mathcal{F}_0/Π_1 have minimal degree at least D_1 . After this, we can proceed to the labeling $\Lambda_1 \sim \text{Ber}(\Pi_1)$, define $\mathcal{F}_1 = \text{clust}(\Lambda_0, \Lambda_1)$, and continue in the same way. The sequence of labelings $\Lambda_0, \dots, \Lambda_n, \dots$ has the property that when we take their concatenation as the binary representation of a real Θ from $[0, 1]$, then Θ has $\text{un}[0, 1]$ marginal.

To show that this labeling has infinite clusters, we need to prove that for any given vertex r , the probability that r is in an infinite cluster is positive, say at least $c > 0$. To do so, we will use Lemma 4.4 mentioned in the beginning of this section. Choose a sequence $\epsilon_0, \epsilon_1, \dots, \epsilon_n, \dots$ such that $\prod_{i \in \mathbb{N}} (1 - \epsilon_i) > c$, and choose the target degree sequence as $D_0 := D(\epsilon_0), D_1 := D(\epsilon_1), \dots$ and so on. Then in the first step, we imitate a $\text{Ber}(\frac{1}{2})$ labeling on a tree whose minimal degree is at least $D(\epsilon_0)$, so the probability that the cell $\Pi_0(r)$ of r will end up within a forking cluster is at least $(1 - \epsilon_0)$ and if that happens, we can achieve our next target degree of $D(\epsilon_1)$ for the component $\mathcal{F}_0(r)$. Then for this component, we will imitate a $\text{Ber}(\frac{1}{2})$ labeling on the tree $\mathcal{F}_0/\Pi_1(r)$ whose minimal degree is at least $D(\epsilon_1)$ so the cell $\Pi_1(r)$ of r will stay within a forking cluster with probability at least $(1 - \epsilon_1)$ conditioned on the event that it stayed within an forking cluster after the first labeling. Thus, the probability that r is within a forking cluster after the first two labelings is at least $(1 - \epsilon_0)(1 - \epsilon_1)$. Continuing in this way, the probability that r is within an infinite cluster in each stage will be at least c and since the cells will increase, it implies that r will be within an infinite cluster even when each label is taken into account.

Why is this construction an FIID? By Lemma 2.2, if the forest Π_i is FIID, then the labeling $\Lambda_i \sim \text{Ber}(\Pi_i)$ is FIID as well. Moreover, as soon as a labeling L is FIID, the indicator function whether or not a vertex v is a furcation of its component in $\text{clust}(L)$ is FIID itself (see the note on measurability related to furcations in the next section right after Definition 4.1). Because of this, when we glue together some of the ‘old’ cells of Π_i to build new cells so that each new cell will contain furcation. This gluing process again can be realized as an FIID. The way we described the process, we need fresh randomness at every stage, but as we already explained at the end of §1, the bits of a single real number can be reorganized to obtain an infinite collection of infinitely many independent bits. So, in fact, the measurable map corresponding to the intuitive description we give can be constructed by using the i.i.d. labels provided originally.

4. Furcations

Furcation vertices will let us reach our target degree goal through Lemma 4.3, and we will find trees containing them in Bernoulli clusters through Lemma 4.5.

Definition 4.1. If T is a tree, we say that $v \in V(T)$ is a *furcation* if after deleting v from T , among the remaining components, there are at least 3 infinite ones. If a tree T has a furcation, we will say that T is *forking*. When \mathcal{F} is a forest and $v \in V(\mathcal{F})$, then we will also say that v is a furcation of \mathcal{F} if it is a furcation of the subtree $\mathcal{F}(v)$ containing it.

This terminology follows [LP16] and is similar but not identical to the *encounter points* of the landmark Burton–Keane argument which are known as *trifurcations* since.

Note that in a bounded degree tree, a vertex v is a furcation if and only if for any $n \in \mathbb{N}$ there exist three paths, \mathcal{P}_1 , \mathcal{P}_2 , and \mathcal{P}_3 , of length at least n which are all emanating from v and are edge and (except for v) vertex disjoint. Thus, the indicator of a vertex being a furcation is measurable, being the infimum of countably many measurable functions.

An infinite example for a tree without furcation is the bi-infinite path. The next example shows trees whose furcations are arranged in an adversarial fashion and if S is the set of their furcations, then the cells of $\text{Vor}(S)$ are not finite. These examples are also worth keeping in mind, as they would be obstacles to our target degree goals. However, they simply cannot occur in an invariant process on a Cayley graph (as we have seen in the application of the mass-transport principle).

Example 4.2. A *ray* emanating from v is a half-infinite path starting from v . Let T_\perp be a tree (defined up to isomorphism) which has a unique vertex of degree 3 and all the other degrees are 2 (that is, three disjoint rays emanating from a single vertex). As a further example, consider the tree $T_{\perp\perp}$ obtained from a bi-infinite path P by attaching to each vertex v a ray R_v (which are not intersecting outside of P).

The following lemma can be proven by induction on r .

LEMMA 4.3. *If C is a finite, connected subset of a tree and contains at least r furcations, then after deleting C from the tree, among the remaining components, there will be at least $r + 2$ which are infinite.*

Recall Theorem 3.2 which said that a $\text{Ber}(\frac{1}{2})$ labeling of \mathbb{T}_4 has infinite clusters. This implies that for a specific vertex v of \mathbb{T}_4 , the probability that v will be in an infinite cluster in a $\text{Ber}(\frac{1}{2})$ -labeling is positive. We, however, need forking clusters. We can get their existence from Theorem 3.2 as follows.

LEMMA 4.4. *For all $\epsilon > 0$, there exists a $D(\epsilon) \in \mathbb{N}$ such that if T is a tree whose minimal degree is at least $D(\epsilon)$ and r is a distinguished vertex (the ‘root’) of T , then in a $\text{Ber}(\frac{1}{2})$ -labeling of T , the cluster of the root r will be a forking one with probability at least $1 - \epsilon$.*

To see this, consider first the rooted tree (T, r) built from rooted copies $(T_1, r_1), \dots, (T_D, r_D)$ of \mathbb{T}_4 (so T_i is a 4-regular tree and r_i is one of its vertices and D will be fixed later) by adding a new vertex r to this collection and make it into a tree by connecting r to r_i for all i (no other edges are added). For r to be in a forking cluster in a $\text{Ber}(\frac{1}{2})$ -labeling of T , it is enough if there are at least three such (T_i, r_i) so that r_i in an infinite cluster of the labeling restricted to T_i and r is connected to r_i . The probability of this clearly goes to 1 as $D \rightarrow \infty$. If $D := D(\epsilon)$ is chosen so that this probability is at least $1 - \epsilon$, then in a tree whose minimal degree is at least D , we can take any vertex to be the root and embed this (T, r) graph into it.

Choose a sequence $\epsilon_0, \epsilon_1, \dots, \epsilon_n, \dots \subset (0, 1)$ tending to 0 fast enough so that $\prod_{n=1}^{\infty} (1 - \epsilon_n) > 0$. Define $D_n := D(\epsilon_n)$ and use this sequence as the target degree in our construction. We got immediately from Lemma 4.4 the following lemma.

LEMMA 4.5. *If, as given above, a sequence $(T_1, r_1), \dots, (T_n, r_n), \dots$ of a rooted trees is given, where the minimal degree of T_n is at least D_n , and each of these trees are independently $\text{Ber}(\frac{1}{2})$ -labeled, then with positive probability, the roots of all of these trees will be in a forking cluster simultaneously.*

We add a more process-oriented corollary to this, which involves random rooted trees (\mathcal{T}_n, r_n) which contains the previous (T_n, r_n) trees as subgraphs by the constraint on their minimal degree. At this point, the distribution of these random trees can be arbitrary as soon as they satisfy the degree constraints. Assume that we start with a random rooted tree (\mathcal{T}_0, r_0) which has minimal degree at least D_0 , and we run the following process. Label the vertices of \mathcal{T}_0 by a $\text{Ber}(\frac{1}{2})$ -labeling \mathcal{L}_0 , and if the cluster of r_0 in $\text{clust}(\mathcal{L}_0)$ is not forking, then stop; otherwise, generate a new random rooted tree (\mathcal{T}_1, r_1) whose minimal degree is at least D_1 , take a $\text{Ber}(1/2)$ -labeling \mathcal{L}_1 of its vertices, and so on. Then, with positive probability, the above process never stops and the generated sequence of bits $\mathcal{L}_0(r_0), \dots, \mathcal{L}_n(r_n), \dots$ will be i.i.d., so the random real number whose bits are this sequence has distribution $\text{un}[0, 1]$.

Remark 4.6. For us, the distinction between an infinite and a forking tree is very important. However, it is known from [LS99] that if λ is any Bernoulli percolation on any Cayley graph, then its infinite clusters are *indistinguishable* by any invariant Borel property (which includes that of being forking). Thus, as soon as there are forking clusters with positive probability, we know that in fact all infinite clusters are forking. Our labelings are not

immediately Bernoulli ones, but cooked up from them in a way that this theorem would likely go through. However, we did not try to use this direction as what we need can be obtained directly from the very basics of percolation theory on a tree.

In our construction, Bernoulli processes on large-scale forests (from Definition 2.5) will be used, and with the aid of Lemma 4.5, we will find forking ones among its clusters. We will put those clusters into use through the following lemma.

LEMMA 4.7. *If Π is an FIID cell-partition of \mathbb{T}_3 and \mathcal{F} is an FIID subforest of \mathbb{T}_3 in such a way that $\Pi \prec \mathcal{F}$, and D is a positive integer, then there is cell partition $\text{furc}_D(\Pi)$ such that $\Pi \prec \text{furc}_D(\Pi) \prec \mathcal{F}$ also holds, and whenever for a vertex v the tree $\mathcal{F}(v)$ is forking, then the $\text{furc}_D(\Pi)(v)$ -cell contains at least D furcations of $\mathcal{F}(v)$.*

Proof. We first show that there exists a partition $\Pi^{\text{there exists furc}}$ for which $\Pi \prec \Pi^{\text{there exists furc}} \prec \mathcal{F}$ and whenever $\mathcal{F}(v)$ is forking, then $\Pi^{\text{there exists furc}}(v)$ contains at least one furcation of $\mathcal{F}(v)$. Let F_Π be the set of those Π -cells which contain at least one furcation of \mathcal{F} . Our goal is achieved if we manage to glue Π -cells within a forking component to form bigger (but still finite) cells in such a way that every new cell contains at least one ‘old’ Π -cell from F_Π .

Voronoi cells on the large-scale forest are just right for this purpose. Move to the large-scale forest \mathcal{F}/Π and build $\text{Vor}(F_\Pi)$. This $\text{Vor}(F_\Pi)$ is ‘almost’ the partition $\Pi^{\text{there exists furc}}$ we seek, except that it lives in \mathcal{F}/Π instead of \mathcal{F} . We bring it back to \mathcal{F} in the obvious way as $\Pi^{\text{there exists furc}} := \text{glue}_{\text{Vor}(F_\Pi)}(\Pi)$. See Figure 1.

Note that the finiteness of the new cells are guaranteed by induction, as every new cell either contains the distinguished finite subset which was an old cell from F_Π , or (in case the \mathcal{F} -component of a cell does not contain any furcation) the new cell is just equal to the old one.

Now that we have $\Pi^{\text{there exists furc}}$, we can define $\text{furc}_D(\Pi)$. Since every $\Pi^{\text{there exists furc}}$ -cell within a forking \mathcal{F} -cluster contains at least one furcation, it is enough if we manage to glue together $\Pi^{\text{there exists furc}}$ -cells in such a way that every new cell of a forking cluster contains at least D ‘old’ $\Pi^{\text{there exists furc}}$ -cells. To achieve this, we can use the same idea as in the very first step described in the high level overview in constructing Π_0 and the associated large-scale tree \mathbb{T}_3/Π_0 , but this time, we work within the forking components of $\mathcal{F}/\Pi^{\text{there exists furc}}$. In the large-scale forest $\mathcal{F}/\Pi^{\text{there exists furc}}$, every forking component has minimal degree at least 3. In $\mathcal{F}/\Pi^{\text{there exists furc}}$, select an FIID vertex set S where the minimal distance between distinct vertices is at least $2D + 1$ and S has at least one element in every forking component of $\mathcal{F}/\Pi^{\text{there exists furc}}$. In the corresponding Voronoi partition $\text{Vor}(S)$, every cell C belonging to a forking component of $\mathcal{F}/\Pi^{\text{there exists furc}}$ will contain at least $|\mathcal{B}_{\mathbb{T}_3}(o, D)| \geq D$ many vertices of $\mathcal{F}/\Pi^{\text{there exists furc}}$ (o denotes an arbitrarily fixed vertex of \mathbb{T}_3). Thus, we can define $\text{furc}_D(\Pi) := \text{glue}_{\text{Vor}(S)}(\Pi^{\text{there exists furc}})$. The new cells are finite again by induction. □

An important property of the cell partition $\text{furc}_D(\Pi)$ follows from Lemma 4.3 and from the finiteness of the $\text{furc}_D(\Pi)$ -cells: the minimal degree of any forking component

of $\mathcal{F}/\text{fur}_{\mathcal{C}_D}(\Pi)$ is at least $D + 2$. So Lemma 4.7 is a natural tool to achieve the previously described target degree goals.

5. The main construction

This section is largely devoted to the proof of our main theorem.

First we define the IID labeling Θ witnessing the truth of our theorem. We define the sequence of cell-partitions $\Pi_0 < \Pi_1 < \dots < \Pi_n < \dots$; this will give us also the sequence of $\Lambda_i \sim \text{Ber}(\Pi_i)$ labels, where, conditioned on Π_i , the label Λ_i will be independent from the previous labels. Of course, Π_i itself depends on $\{(\Pi_j, \Lambda_j)\}_{j < i}$.

Here, Π_0 and Λ_0 are as defined before in §3. Assume that $\Pi_0 < \dots < \Pi_n$ and $\Lambda_0, \dots, \Lambda_n$ are defined. Let $\mathcal{F}_n := \text{clust}(\Lambda_0, \dots, \Lambda_n)$, where $(\Lambda_0, \dots, \Lambda_n)$ is the $\{0, 1\}^{n+1}$ -label obtained by concatenating the Λ_i terms.

We want to define Π_{n+1} in such a way that $\Pi_n < \Pi_{n+1} < \mathcal{F}_n$, and if for a vertex v the tree $\mathcal{F}_n(v)$ is forking, then the $\Pi_{n+1}(v)$ -cell should contain at least D_{n+1} furcations of \mathcal{F}_n . We use Lemma 4.7 for the pair $\Pi_n < \mathcal{F}_n$ and define $\Pi_{n+1} := \text{fur}_{\mathcal{C}_{D_{n+1}}}(\Pi_n)$. As we mentioned at end of the previous section, we know by Lemma 4.3 that the minimal degree of a forking component of $\mathcal{F}_{n+1}/\Pi_{n+1}$ is at least $D_{n+1} + 2$.

This concludes the construction of $\Pi_0 < \dots < \Pi_n < \dots$ and thus also that of $\Lambda_i \sim \text{Ber}(\Pi_i)$, with the specification that, conditioned on Π_i , the Λ_i must be independent of the previous steps. This implies that for a any vertex o fixed in advance, the sequence $\Lambda_0(o), \Lambda_1(o), \dots$ of fair bits is i.i.d., and thus if we define $\Theta(o)$ to be the real number from $[0, 1]$ whose consecutive bits are $\Lambda_0, \dots, \Lambda_n, \dots$, then $\Theta(o)$ has distribution $\text{un}[0, 1]$. Thus, Θ has $\text{un}[0, 1]$ marginals.

Now we turn to the proof of our main theorem.

Proof of Theorem 1.1. Because $\Pi_{n+1} < \mathcal{F}_n := \text{clust}(\Lambda_0, \dots, \Lambda_n)$, if we define $\Pi_\infty(o) := \bigcup_{i \in \mathbb{N}} \Pi_i(o)$, then for any $v_1, v_2 \in \Pi_\infty(o)$ and $m \in \mathbb{N}$, we have $\Lambda_m(v_1) = \Lambda_m(v_2)$. Thus, $v_1, v_2 \in \Pi_\infty(o)$ also implies $\Theta(v_1) = \Theta(v_2)$, and thus $\Pi_\infty(o)$ will be contained within a single cluster of $\text{clust}(\Theta)$. Moreover, if $\mathcal{F}_n(o)$ is a forking cluster, then $|\Pi_{n+1}(o)| \geq D_{n+1}$ as it contains at least D_{n+1} furcation of $\mathcal{F}_n(o)$. So if $\mathcal{F}_i(o)$ is forking for every i , then $\Pi_\infty(o) \supset \Pi_m(o)$ contains at least D_m element for any m , and as $D_m \rightarrow \infty$, this implies the part of Theorem 1.1 which claims the existence of an IID labeling Θ , for which $\text{clust}(\Theta)$ contains infinite clusters.

It indeed happens with positive probability that $\mathcal{F}_n(o)$ is forking for all n , because of the corollary to Lemma 4.5 using the process of generating rooted trees. The correspondence is as follows. Start with the rooted tree $(T_0, r_0) := (\mathbb{T}_3/\Pi_0(o), \Pi(o))$ whose minimal degree is greater than D_0 . Use our $\Lambda_0 \sim \text{Ber}(\Pi_0)$ which is a $\text{Ber}(\frac{1}{2})$ -labeling of T_0 . Stop the process if the cluster $\mathcal{F}_0(o)$ of $\Pi_0(o)$ is not forking, otherwise continue by creating the next random rooted tree $(\mathcal{F}_0/\Pi_1(o), \Pi_1(o))$, whose minimal degree is greater than D_1 (the way we use Lemmas 4.3 and 4.7, this minimal degree actually is at least $D_1 + 2$). In general, if the process has not stopped, then the rooted tree (T_n, r_n) is constructed as the random rooted tree $(\mathcal{F}_n/\Pi_{n+1}(o), \Pi_{n+1}(o))$. Notice that—by the finiteness of the cells—moving from the tree $\mathcal{F}_n(o)$ to its large-scale version $\mathcal{F}_n/\Pi_{n+1}(o)$ does not change its being forking or not (while it increases its minimal degree), so the correspondence

between our construction and the process oriented corollary to Lemma 4.5 is complete. This finishes the proof that $\text{clust}\Theta$ contains infinite clusters with positive probability, and thus by ergodicity, it contains them almost surely.

It is easy to modify the construction so that every cluster will be infinite. Let us define a new Θ^∞ labeling from Θ as follows: if T is a finite Θ cluster which is at distance 1 from at least one infinite Θ cluster, then, since T is finite, the collection \mathcal{C} of such clusters is finite, and thus T can randomly choose one of these infinite clusters $\text{choice}(T) \in \mathcal{C}$ (we say a few words about this at the end). Then we define a new FIID labeling Θ^1 by replacing the labels of the vertices of every such T by the Θ -label of $\text{choice}(T)$. Then we can progressively do the same thing with Θ^1 and obtain the new FIID labeling Θ^2 and so on. Thus, we get a sequence of FIID labelings Θ, Θ^1, \dots with the property that for any given vertex v , the sequence of labels $\Theta(v), \Theta^1(v), \dots$ is constant from one index (in fact, it changes at most once), and thus its limit $\Theta^\infty(v)$ is well defined. If a finite cluster is at distance r from an infinite one under the original Θ labeling, then the vertices of that cluster will be in an infinite cluster of Θ^r at the latest. So in Θ^∞ , there are only infinite clusters. In our original Θ labeling, knowing whether or not the cluster of a vertex is finite or infinite does not change the distribution of its Θ -label, the conditional distribution is still $\text{un}[0, 1]$, so Θ^∞ still has uniform marginals.

Now we explain how a finite cluster T can choose one of its infinite neighbors. As we can always access to new fresh randomness, we can obtain a random variable $\text{vote}(u) \sim \text{un}[0, 1]$ at every vertex u (independently from each other). Since T is finite, there are only finitely many vertices of the whole tree which are at distance 1 from T and which are themselves contained in an infinite cluster, so one of those, say w , had maximal $\text{vote}(w)$. Then let $\text{choice}(T)$ be the infinite cluster containing w . □

We mentioned in §1 that [Ch10, Theorem 1.1] implies that the clusters in Θ and Θ^∞ must be hyperfinite. Here we show their hyperfiniteness directly.

To see this for the Θ -clusters, notice that not only $\Pi_\infty(o) \subset \text{clust}(\Theta)$ for any o , which already means $\Pi_\infty \prec \text{clust}(\Theta)$, but actually $\Pi_\infty = \text{clust}(\Theta)$ and $\Pi_\infty(o)$ is an increasing union of the finite $\Pi_n(o)$ s. To see this, consider an edge e connecting a vertex $v_{\text{in}} \in \Pi_\infty(o)$ with $v_{\text{out}} \notin \Pi_\infty(o)$. Notice that for any i , the labels $\Lambda_i(v_{\text{in}})$ and $\Lambda_i(v_{\text{out}})$ are independent, so they cannot be all equal. So e is deleted from $\text{clust}(\Theta)$.

To see the hyperfiniteness of the Θ^∞ -clusters, we reorganize the way we build them using some new sequence of cell-partitions Π_n^\bullet . For a finite Θ -cluster K , it is well defined which is the infinite Θ -cluster $\text{goal}(K)$ to which it finally will be attached, by $\Theta^\infty(K) = \Theta^\infty(\text{goal}(K)) = \Theta(\text{goal}(K))$; by being inside a tree, it is also well defined what is the vertex $\text{m}(K)$ of $\text{goal}(K)$ which is closest to K . Moreover, K will be attached to an infinite cluster at a certain stage indexed by the smallest $i =: I(K)$ for which $\Theta^i(K) = \Theta^\infty(K)$.

We will reorganize the original Π_n partitions into a new one using these two parameters $I(K)$ and $\text{m}(K)$. Extend these parameters to vertices in a natural way: for a vertex v within a finite Θ -cluster K , let $I(v) := I(K)$, $\text{m}(v) := \text{m}(K)$, while if v is within an infinite Θ -cluster, then let $I(v) := 0$, $\text{m}(v) := v$. The I index has the following property: if P is a one-sided infinite path which stays completely within an infinite Θ^∞ -cluster and starts at a vertex v whose Θ -cluster is already infinite, then I can only increase along P as we start

from its starting point and move away from it. Moreover, if u_1, u_2 are vertices of P with $I(u_1) = I(u_2)$, then u_1 and u_2 are in the same Θ -cluster.

The cells of the new partition Π_n^\bullet are built as follows: the vertices of an infinite Θ -cluster \mathcal{K} are partitioned into Π_n -cells. For a given Π_n -cell C within \mathcal{K} , we collect together those finite Θ -clusters K whose $m(K)$ is in C and whose index $I(K)$ is at most n to form a single Π_n^\bullet -cell. The finite Θ -clusters K for which $I(K) > n$ will be individual Π_n^\bullet -cells.

To define Π_n^\bullet formally, introduce an auxiliary partitioning Π_n° so that for a vertex $\Pi_n^\circ(o) := \Pi_n(o)$ if $\Pi_\infty(o)$ is infinite and $\Pi_n^\circ(o) := \Pi_\infty(o)$ if $\Pi_\infty(o)$ is finite. Finally, let us define

$$\Pi_n^\bullet(v) := \begin{cases} \Pi_n^\circ(v) & \text{if } I(v) > n, \\ \bigcup\{\Pi_n^\circ(x) \mid I(x) \leq n, \Pi_n^\circ(m(x)) = \Pi_n^\circ(m(v))\} & \text{if } I(v) \leq n. \end{cases}$$

Then every $\Pi_m^\bullet(v)$ -class is finite because if it was infinite, then by Kőnig’s lemma, we could find an infinite path P within $\Pi_m^\bullet(v)$ for which $m(u) =: u_0$ would be the same for every vertex u of P and this u_0 would be the starting point of P . By definition of the $m(u)$ parameter, as we walk away from u_0 , we get out from the infinite Θ -cluster of u_0 immediately, so $I(u) > 0$ for any vertex $u \neq u_0$ of P . However, by the definition of $\Pi_m^\bullet(v)$, the index $I(u)$ is at most m within it, so from one vertex w on, I is constant on P . However, that would mean that the full infinite part of P after w is contained in a single Θ -cluster, so that cluster would be infinite in the first place.

However, the increasing union of $\bigcup_{m \in \mathbb{N}} \Pi_m^\bullet(v)$ is exactly the Θ^∞ -cluster of v . To see this, note that if u is in the Θ^∞ -cluster of v , then there will be some n such that $\Pi_n(m(u)) = \Pi_n(m(v))$, so if $j \geq \max\{n, I(u), I(v)\}$, then $u \in \Pi_j^\bullet(v)$.

6. The case of finitary FIIDs

An FIID process is *finitary* if it can be defined with a code $f : [0, 1]^V \rightarrow [0, 1]^V$ which determines the label of any vertex based on the source in a finite neighborhood of that vertex. That is, for almost all $\omega \in [0, 1]^V$ and each vertex v , there exists a random radius $r = r(\omega, v) \in \mathbb{N}$, such that if ω_1 has the same restriction to $B_r(v)$ as ω , then the label of v is the same using either sources of randomness, that is, $f(\omega)(v) = f(\omega_1)(v)$. We will call the smallest such $r(\omega, v)$ the *level* of v , and denote it by $\text{lev}_f(v)$. If f has this property, then we will call it finitary as well. To show that a given FIID is not finitary, it is not enough to point out that a particular code used in its definition is not finitary as different codes could determine the same process.

We will show that an FIID process which witnesses the truth of Theorem 1.1 cannot be finitary. It was raised as an open question if there exists an FIID process on a non-amenable group which is not finitary during the discussion of a talk by Ray and Spinka (to access the talk, see [RS22], the talk is based on the paper [ARS21]), and our construction thus gives an affirmative answer to this question. The label set in our process is the interval $[0, 1]$, and to our knowledge, the question if there exists a non-finitary FIID with a finite label set on a non-amenable group is still open.

Let L be a finitary FIID labeling with un $[0, 1]$ marginals. We will show that every cluster of L is finite almost surely.

Let us fix a finitary code f for L and assume that L is obtained using this f . If C is a cluster of L , then we will call the set of its vertices whose level is minimal the *base* of C . Let $\text{clust}_1(L)$ be the collection of those infinite clusters whose base is finite, and $\text{clust}_\infty(L)$ be the collection of those infinite clusters whose base is infinite. We will show that both $\text{clust}_1(L)$ and $\text{clust}_\infty(L)$ must be empty almost surely.

If $\text{clust}_1(L)$ was not empty, then we would get a contradiction with Lemma 2.3, since the clusters of $\text{clust}_1(L)$ are infinite with finitely many distinguished vertices, from which a single distinguished vertex can be picked uniformly.

To show that $\text{clust}_\infty(L)$ is empty almost surely, let us partition it further, and for $m \in \mathbb{N}$, let $\text{clust}_\infty^m(L)$ be the collection of those clusters within $\text{clust}_\infty(L)$ whose base consists of vertices of level exactly m . Since this is a countable partition, it is enough to show that $\text{clust}_\infty^m(L)$ is empty almost surely for any fixed m . If $\text{clust}_\infty^m(L)$ was not empty, then infinitely many vertices had level m and the same label (the base of any cluster in $\text{clust}_\infty^m(L)$ would be such a collection of vertices). In particular, there existed a pair of vertices whose distance is at least $2m + 1$, their level is m , and they had the same label. Since the possible pairs of vertices form a countable set, it is enough to show that this has zero probability for any fixed pair. Pick two vertices v_1 and v_2 whose distance is at least $2m + 1$, conditioned on $\text{lev}_f(v_1) = \text{lev}_f(v_2) = m$, their labels are independent, since $B_m(v_1)$ and $B_m(v_2)$ are disjoint. If the event $\text{lev}_f(v) = m$ has positive probability, then conditioned on it, the distribution of the label of v still must be atomless, otherwise the unconditional distribution could not be $\text{un}[0, 1]$. However, independent samples from an atomless distribution are different almost surely.

Being finitary is a locality condition, and for other natural notions of locality, we refer the reader to [MS22] (in their more refined framework, what we called finitary above is called *stop-finitary*). Note also that the classic example of isomorphism between Bernoulli shifts by Meshalkin ([M59]) is also a finitary one and that in [KS79], Keane and Smorodinsky proved the existence of finitary isomorphisms between Bernoulli shifts of the same entropy over \mathbb{Z} , and thus strengthening Ornstein's landmark theorem (for a survey of finitary codings in this direction, see [S06]).

Acknowledgement. This work was partially supported by the ERC Consolidator Grant 772466 'NOISE'. At earlier stages, the work was also partially supported by NSF Grant DMS-1007244 and OTKA Grant K76099. I am grateful to Russell Lyons for suggesting working on this problem and for useful discussions and for helping improving the first version of the text. I also thank the anonymous referee of the first submitted version for useful feedback. I similarly thank the anonymous referee of the current version. I am indebted to Gábor Pete for useful discussions throughout the writing of this version and for Sándor Rokob for useful feedback.

REFERENCES

- [ARS21] O. Angel, G. Ray and Y. Spinka. Uniform even subgraphs and graphical representations of Ising as factors of i.i.d. *Preprint*, 2021, [arXiv:2112.03228](https://arxiv.org/abs/2112.03228).
 [B81] B. Bollobás. The independence ratio of regular graphs. *Proc. Amer. Math. Soc.* **83** (1981), 433–436.

- [B18] L. Bowen. A brief introduction to sofic entropy theory. *Proceedings of ICM (Rio de Janeiro 2018)*. Vol. III. Eds. B. Sirakov, P. Ney de Souza and M. Viana. World Scientific, Singapore, 2018, pp. 1847–1866. Preprint, 2017, available at <https://arxiv.org/abs/1711.02062>.
- [BLPS99] I. Benjamini, R. Lyons, Y. Peres and O. Schramm. Group-invariant percolation on graphs. *Geom. Funct. Anal.* **9** (1999), 29–66.
- [BSzV15] Á. Backhausz, B. Szegedy and B. Virág. Ramanujan graphings and correlation decay in local algorithms. *Random Structures Algorithms* **47** (2015), 424–435.
- [ChI10] I. Chifan and A. Ioana. Ergodic subequivalence relations induced by a Bernoulli action. *Geom. Funct. Anal.* **20** (2010), 53–67.
- [G10] D. Gaboriau. Orbit equivalence and measured group theory. *Proceedings of the ICM (Hyderabad, India, 2010)*. Vol. III. Eds. R. Bhatia, A. Pal, G. Rangarajan, V. Srinivas and M. Vanninathan. Hindustan Book Agency (India), India and World Scientific, Singapore, 2010, pp. 1501–1527.
- [GL09] D. Gaboriau and R. Lyons. A measurable-group-theoretic solution to von Neumann’s problem. *Invent. Math.* **177** (2009), 533–540.
- [KS79] M. Keane and M. Smorodinsky. Bernoulli schemes of the same entropy are finitarily isomorphic. *Ann. of Math. (2)* **109**(2) (1979), 397–406.
- [L17] R. Lyons. Factors of IID on trees. *Comb. Probab. Comput.* **26** (2017), 285–300.
- [LP16] R. Lyons and Y. Peres. *Probability on Trees and Networks*. Cambridge University Press, Cambridge, 2016. Available at <http://mypage.iu.edu/rdlyons>.
- [LS99] R. Lyons and O. Schramm. Indistinguishability of percolation clusters. *Ann. Probab.* **27** (1999), 1809–1836.
- [M59] L. D. Meshalkin. A case of isomorphism of Bernoulli schemes. *Dokl. Akad. Nauk SSSR* **128** (1959), 41–44.
- [MS22] T. Meyerovitch and Y. Spinka. Entropy-efficient finitary codings. Preprint, 2022, [arXiv:2201.06542](https://arxiv.org/abs/2201.06542).
- [NSZ20] D. Nam, A. Sly and L. Zhang. Ising model on trees and factors of IID. *Comm. Math. Phys.* **389** (2022), 1009–1046.
- [OW87] D. S. Ornstein and B. Weiss. Entropy and isomorphism theorems for actions of amenable groups. *J. Anal. Math.* **48** (1987), 1–141.
- [P19] G. Pete. *Probability and Geometry on Groups*. Book in preparation. <http://www.math.bme.hu/gabor/PGG.pdf>.
- [RS22] G. Ray and Y. Spinka. Graphical representations and gradients of Ising as factors of i.i.d. Talk at the Percolation Today Webinar, January 18, 2022. Accessible from <https://percolation.ethz.ch/>.
- [RV17] M. Rahman and B. Virág. Local algorithms for independent sets are half-optimal. *Ann. Probab.* **45** (2017), 1543–1577.
- [S06] J. Serafin. Finitary codes, a short survey. *Dynamics & Stochastics (Institute of Mathematical Statistics Lecture Notes—Monograph Series, 48)*. Eds. D. Denteneer, F. den Hollander and E. Verbitskiy. Institute of Mathematical Statistics, Beachwood, OH, 2006, pp. 262–273.
- [S10] A. Sly. Computational transition at the uniqueness threshold. *Proc. 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, Piscataway, NJ, 2010, pp. 287–296. Available at <https://arxiv.org/abs/1005.5584>.
- [S20] Y. Spinka. Finitary codings for spatial mixing Markov random fields. *Ann. Probab.* **48** (2020), 1557–1591.
- [vdBS99] J. van den Berg and J. E. Steif. On the existence and nonexistence of finitary codings for a class of random fields. *Ann. Probab.* **27** (1999), 1501–1522.