Constraining the clustering transition for
colorings of sparse random graphs

Michael Anastos, Alan Frieze∗ and Wesley Pegden†
Department of Mathematical Sciences
Carnegie Mellon University
Pittsburgh PA 15213

December 31, 2017

Abstract

Let Ωq denote the set of proper q-colorings of the random graph G_{n,m}, m = dn/2 and let Hq be the graph
with vertex set Ωq and an edge \{σ, τ\} where σ, τ are mappings \([n] → [q]\) iff h(σ, τ) = 1. Here h(σ, τ)
is the Hamming distance \(|\{v ∈ [n] : σ(v) ≠ τ(v)\}|\). We show that w.h.p. Hq contains a single giant component
containing almost all colorings in Ωq if d is sufficiently large and q ≥ \frac{cd}{\log d} for a constant c > 3/2.

1 Introduction

In this short note, we will discuss a structural property of the set Ωq of proper q-colorings of the random graph
G_{n,m}, where m = dn/2 for some large constant d. For the sake of precision, let us define Hq to be the graph
with vertex set Ωq and an edge \{σ, τ\} iff h(σ, τ) = 1 where h(σ, τ) is the Hamming distance \(|\{v ∈ [n] : σ(v) ≠ τ(v)\}|\). In the Statistical Physics literature the definition of Hq may be that colorings σ, τ are connected by an edge in Hq
whenever h(σ, τ) = o(n). Our theorem holds a fortiori if this is the case.

Heuristic evidence in the statistical physics literature (see for example [15]) suggests there is a clustering transition c_d such that for q > c_d, the graph H_q is dominated by a single connected component, while for q < c_d, an exponential number of components are required to cover any constant fraction of it; it may be that c_d ≈ \frac{d}{\log q}. (Here A(d) ≈ B(d) is taken to mean that A(d)/B(d) → 1 as d → ∞. We do not assume d → ∞, only that d is a

∗Research supported in part by NSF grant DMS1362785
†Research supported in part by NSF grant DMS1363136
sufficiently large constant, independent of $n$.) Recall that $G_{n,m}$ for $m = dn/2$ becomes $q$-colorable around $q \approx \frac{d}{2\log d}$ or equivalently when $d \approx 2q \log q$, [3, 7]. In this note, we prove the following:

**Theorem 1.1.** If $q \geq \frac{cd}{\log d}$ for constant $c > 3/2$, and $d$ is sufficiently large, then w.h.p. $H_q$ contains a giant component that contains almost all of $\Omega_q$.

In particular, this implies that the clustering transition $c_d$, if it exists, must satisfy $c_d \leq \frac{3}{2} \frac{d}{\log d}$.

Theorem 1.1 falls into the area of “Structural Properties of Solutions to Random Constraint Satisfaction Problems”. This is a growing area with connections to Computer Science and Theoretical Physics. In particular, much of the research on the graph $H_q$ has been focused on the structure near the colorability threshold, e.g. Bapst, Coja-Oghlan, Hetterich, Rassman and Vilenchik [4], or the clustering threshold, e.g. Achlioptas, Coja-Oghlan and Ricci-Tersenghi [2], Molloy [13]. Other papers heuristically identify a sequence of phase transitions in the structure of $H_q$, e.g. Krzakala, Montanari, Ricci-Tersenghi, Semerjian and Zdeborová [12], Zdeborová and Krzakala [15]. The existence of these transitions has been shown rigorously for some other CSPs. One of the most spectacular examples is due to Ding, Sly and Sun [8] who rigorously showed the existence of a sharp satisfiability threshold for random $k$-SAT.

An obvious target for future work is improving the constant in Theorem 1.1 to 1. We should note that Molloy [13] has shown that w.h.p. there is no giant component if $q \leq \frac{(1-\epsilon_d)d}{\log d}$, for some $\epsilon_d > 0$. Looking in another direction, it is shown in [9] that w.h.p. $H_q, q \geq d + 2$ is connected. This implies that Glauber Dynamics on $\Omega_q$ is ergodic. It would be of interest to know if this is true for some $q \ll d$.

Before we begin our analysis, we briefly explain the constant $3/2$. We start with an arbitrary $q$-cloring and then re-color it using only approximately $\approx \frac{d}{\log d}$ of the given colors. We then use a disjoint set of approximately $\frac{d}{2\log d}$ colors to re-color it with a target $\chi \approx \frac{d}{2\log d}$ coloring $\tau$.

## 2 Greedily Re-coloring

Our main tool is a theorem from Bapst, Coja-Oghlan and Efthymiou [5] on planted colorings. We consider two ways of generating a random coloring of a random graph. We will let $Z_q = |\Omega_q|$. The first method is to generate a random graph and then a random coloring. In the second method, we generate a random (planted) coloring and then generate a random graph compatible with this coloring.

**Random coloring of the random graph $G_{n,m}$:** Here we will assume that $m$ is such that w.h.p. $Z_q > 0$.

(a) Generate $G_{n,m}$ subject to $Z_q > 0$. 

(b) Choose a \( q \)-coloring \( \sigma \) uniformly at random from \( \Omega_q \).

(c) Output \( \Pi_1 = (G_{n,m}, \sigma) \).

**Planted model:**

1. Choose a random partition of \([n]\) into \( q \) color classes \( V_1, V_2, \ldots, V_q \) subject to
   \[
   \sum_{i=1}^q \left( \frac{|V_i|}{2} \right) \leq \left( \frac{n}{2} \right) - m.
   \]

2. Let \( \Gamma_{\sigma,m} \) be obtained by adding \( m \) random edges, each with endpoints in different color classes.

3. Output \( \Pi_2 = (\Gamma_{\sigma,m}, \sigma) \).

We will use the following result from [5]:

**Theorem 2.1.** Let \( d = 2m/n \) and suppose that \( d \leq 2(q-1) \log(q-1) \). Then \( \text{Pr}(\Pi_2 \in \mathcal{P}) = o(1) \) implies that \( \text{Pr}(\Pi_1 \in \mathcal{P}) = o(1) \) for any graph+coloring property \( \mathcal{P} \).

Consequently, we will use the planted model in our subsequent analysis. Let

\[
q_0 = \frac{q}{q-1} \cdot \frac{d}{\log d - 7 \log \log d} \approx \frac{d}{\log d}.
\]

The property \( \mathcal{P} \) in question will be: “the given \( q \)-coloring can be reduced via single vertex color changes to a \( q_0 \) coloring” where \( \alpha > 1 \) is constant.

In a random partition of \([n]\) into \( q \) parts, the size of each part is distributed as \( \text{Bin}(n, q^{-1}) \) and so the Chernoff bounds imply that w.h.p. in a random partition each part has size \( \frac{n}{q} \left( 1 \pm \frac{\log n}{n^{1/2}} \right) \).

We let \( \Gamma \) be obtained by taking a random partition \( V_1, V_2, \ldots, V_q \) and then adding \( m = \frac{1}{2}dn \) random edges so that each part is an independent set. These edges will be chosen from

\[
N_q = \left( \binom{n}{2} \right) - (1 + o(1))q \left( \binom{n/q}{2} \right) = (1 - o(1)) \frac{n^2}{2} \left( 1 - \frac{1}{q} \right)
\]

possibilities. So, let \( \hat{d} = \frac{mn}{N_q} \approx \frac{dq}{q-1} \) and replace \( \Gamma \) by \( \hat{\Gamma} \) where each edge not contained in a \( V_i \) is included independently with probability \( \hat{p} = \frac{\hat{d}}{n} \). \( V_1, V_2, \ldots, V_q \) constitutes a coloring which we will denote by \( \sigma \). Now \( \hat{\Gamma} \) has \( m \) edges with probability \( \Omega(n^{-1/2}) \) and one can check that the properties required in Lemmas 2.2 and 2.3 below all occur with probability \( 1 - o(n^{-1/2}) \) and so we can equally well work with \( \Gamma \).
Now consider the following algorithm for going from $\sigma$ via a path in $\Omega_q$ to a coloring with significantly fewer colors. It is basically the standard greedy coloring algorithm, as seen in Bollobás and Erdős [6], Grimmett and McDiarmid [10] and in particular Shamir and Upfal [14] for sparse graphs.

In words, it goes as follows. At each stage of the algorithm, $U$ denotes the set of vertices that have not been re-colored. Having used $r - 1$ colors to color some subset of vertices we start using color $r$. We let $W_j = V_j \cap U$ denote the uncolored vertices of $V_j$ for $j \geq 1$. We then let $k$ be the smallest index $j$ for which $W_j \neq \emptyset$. This is an independent set and so we can re-color the vertices of $W_k$, one by one, with the color $r$. We let $U_r \subseteq U$ denote the set of vertices that may possibly be re-colored $r$ by the algorithm i.e. those vertices with no neighbors in $C_r$, the current set of vertices colored $r$. Each time we re-color a vertex with color $r$, we remove its neighbors from $U_r$. We continue with color $r$, until $U_r = \emptyset$. After which, $C_r$ will be the set of vertices that are finally colored with color $r$.

At any stage of the algorithm, $U$ is the set of vertices whose colors have not been altered. The value of $L$ in line D is $n/\log^2 \hat{d}$.

ALGORITHM GREEDY RE-COLOR
begin
  Initialise: $r = 0, U = [n], C_0 \leftarrow \emptyset$;
  repeat;
    $r \leftarrow r + 1, C_r \leftarrow \emptyset$;
    Let $W_j = V_j \cap U$ for $j \geq 1$ and let $k = \min \{j : W_j \neq \emptyset\}$;
    A: $C_r \leftarrow W_k, U \leftarrow U \setminus C_r, U_r \leftarrow U \setminus \{\text{neighbors of } C_r \text{ in } \hat{\Gamma}\}$;
    If $r < k$, re-color every vertex in $C_r$ with color $r$;
    B: repeat (Re-color some more vertices with color $r$);
    C: Arbitrarily choose $v \in U_r, C_r \leftarrow C_r + v, U_r \leftarrow U_r - v$;
    $U_r \leftarrow U_r \setminus \{\text{neighbors of } v \text{ in } \hat{\Gamma}\}$;
    until $U_r = \emptyset$;
    D: until $|U| \leq L$;
    Re-color $U$ with $\frac{\hat{d}}{\log^2 \hat{d}} + 2$ unused colors from our initial set of $q_0$ colors;
end

We first observe that each re-coloring of a single vertex $v$ with line C can be interpreted as moving from a coloring of $\Omega_q$ to a neighboring coloring in $H_q$. This requires us to argue that the re-coloring by GREEDY RE-COLOR is such that the coloring of $\hat{\Gamma}$ is proper at all times. We argue by induction on $r$ that the coloring at line A is proper. When $r = 1$ there have been no re-colorings. Also, during the loop beginning at line B we only re-color vertices with color $r$ if they are not neighbors of the set $U_r$ of vertices colored $r$. This guarantees that the coloring remains proper until we reach line D. The following lemma shows that we can then reason as in Lemma 2 of Dyer, Flaxman, Frieze and Vigoda [9], as will be explained subsequently.
Lemma 2.2. Let \( p = m/n^2 = \Delta/n \) where \( \Delta \) is some sufficiently large constant. With probability \( 1 - o(n^{-1/2}) \), every \( S \subseteq [n] \) with \( s = |S| \leq n / \log^2 \Delta \) contains at most \( s\Delta / \log^2 \Delta \) edges.

The above lemma, is Lemma 7.7(i) of Janson, Łuczak and Ruciński [11] and it implies that if \( \Delta = \tilde{d} \) then w.h.p. \( \hat{\Gamma}_U \) at line D contains no \( K \)-core, \( K = \frac{2\tilde{d}}{\log^2 \tilde{d}} + 1 \). Here \( \Gamma_U \) denotes the sub-graph of \( \hat{\Gamma} \) induced by the vertices \( U \). For a graph \( G = (V,E) \) and \( K \geq 0 \), the \( K \)-core is the unique maximal set \( S \subseteq V \) such that the induced subgraph on \( S \) has minimum degree at least \( K \). A graph without a \( K \)-core is \( K\)-degenerate i.e. its vertices can be ordered as \( v_1, v_2, \ldots, v_n \) so that \( v_i \) has at most \( K - 1 \) neighbors in \( \{v_1, v_2, \ldots, v_{i-1}\} \). To see this, let \( v_n \) be a vertex of minimum degree and then apply induction.

We argue now that we can re-color the vertices in \( U \) with \( K + 1 \) new colors, all the time following some path in \( H_q \). Let \( v_1, \ldots, v_n \) denote an ordering of \( U \) such that the degree of \( v_i \) is less than \( K \) in the subgraph \( \hat{\Gamma}_i \) of \( \hat{\Gamma} \) induced by \( \{v_1, v_2, \ldots, v_i\} \). We will prove the claim by induction. The claim is trivial for \( i = 1 \). By induction there is a path \( \sigma_0, \sigma_1, \ldots, \sigma_r \) from the coloring \( \sigma_0 \) of \( U \) at line B, restricted to \( \hat{\Gamma}_{i-1} \) using only \( K + 1 \) colors to do the re-coloring.

Let \((w_j, c_j)\) denote the \((vertex, color)\) change defining the edge \( \{\sigma_{j-1}, \sigma_j\} \). We construct a path (of length \( \leq 2r \)) that re-colors \( \hat{\Gamma}_i \). For \( j = 1, 2, \ldots, r \), we will re-color \( w_j \) to color \( c_j \), if no neighbor of \( w_j \) has color \( c_j \). Failing this, \( v_i \) must be the only neighbor of \( w_j \) that is colored \( c_j \). This is because \( \sigma_r \) is a proper coloring of \( \hat{\Gamma}_{i-1} \). Since \( v_i \) has degree less than \( K \) in \( \hat{\Gamma}_i \), there exists a new color for \( v_i \) which does not appear in its neighborhood. Thus, we first re-color \( v_i \) to any new (valid) color, and then we re-color \( w_j \) to \( c_j \), completing the inductive step. Note that because the colors used in Step D have not been used in Steps A,B,C, this re-coloring does not conflict with any of the coloring done in Steps A,B,C.

We need to show next that each Loop B re-colors a large number of vertices. Let \( \alpha_1(G) \) denote the minimim size of a maximal independent set of a graph \( G \) i.e. an independent set that is not contained in any larger independent set. The round will re-color at least \( \alpha_1(\Gamma_U) \) vertices, where \( U \) is as at the start of Loop B. The following result is from Lemma 7.8(i) of [11].

Lemma 2.3. Let \( p = m/n^2 = \Delta/n \) where \( \Delta \) is some sufficiently large constant. \( \alpha_1(G_{n,m}) \geq \frac{\log \Delta - 3 \log \log \Delta}{p} \) with probability \( 1 - o(n^{-1/2}) \). (see Lemma 7.8(i)).

Suppose now that we take \( u_0 \) to be the size of \( U \) at the beginning of Step A and that \( u_t \) is the size of \( U \) after \( t \) vertices have been finally colored \( r \). Thus we assume that \( u_{|W_k|} \) is the size of \( U \) at the start of Step B. We observe that,

\[
u_{t+1} \text{ stochastically dominates } u_t - \text{Bin}(u_t, \bar{p}) - 1.
\]

This is because the edges inside \( U \) are unconditioned by the algorithm and because \( v \in V_j \) has no neighbors in \( V_j \) for \( j \geq 1 \). On the other hand, if we apply Algorithm GREEDY RE-COLOR
to \( G_{n,\hat{p}} \) then \( (1) \) is replaced by the recurrence
\[
\hat{u}_{t+1} = \hat{u}_t - \text{Bin}(\hat{u}_t, \hat{p}) - 1. \tag{2}
\]

(Putting \( V_j = \{j\} \) means that GREEDY RE-COLOR is running on \( G_{n,\hat{p}} \).

Comparing \( (1) \) and \( (2) \) we see that we can couple the two applications of GREEDY RE-COLOR so that \( u_t \geq \hat{u}_t \) for \( t \geq 0 \). Now the application of Loop B re-colors a maximal independent set of the graph \( \hat{\Gamma}_U \) induced by \( U \) as it stands at the beginning of the loop. The size of this set dominates the size of a maximal independent set in the random graph \( G_{|U|, \hat{p}} \). So if we generate \( G_{|U|, \hat{p}} \) and then delete some edges, we see that every independent set of \( G_{|U|, \hat{p}} \) will be contained in an independent set of \( \Gamma_U \). And so using Lemma 2.3 we see that w.h.p. each execution of Loop B re-colors at least
\[
\log(\hat{d}/\log\hat{d}) - 3 \log \log(\hat{d}/\log\hat{d}) \geq \frac{q - 1}{q} \cdot \frac{\log d - 6 \log \log d}{d} n
\]
vertices, for \( d \) sufficiently large. We have replaced \( \Delta \) of Lemma 2.3 by \( \hat{d}/\log\hat{d} \) to allow for the fact that we have replaced \( n \) by \( |U| \geq L \). Consequently, at the end of Algorithm GREEDY RE-COLOR we will have used at most
\[
\frac{q}{q - 1} \cdot \frac{d}{\log d - 6 \log \log d} + \frac{\hat{d}}{\log^2 \hat{d}} + 2 \leq \frac{q}{q - 1} \cdot \frac{d}{\log d - 7 \log \log d} = q_0
\]
colors. The term \( \frac{\hat{d}}{\log^2 \hat{d}} + 2 \) arises from the re-coloring of \( U \) at line D.

**Finishing the proof:** Now suppose that \( q \geq \frac{cd}{\log d} \) where \( d \) is large and \( c > 3/2 \). Fix a particular \( \chi \)-coloring \( \tau \). We prove that almost every \( q \)-coloring \( \sigma \) can be transformed into \( \tau \) changing one color at a time. It follows that for almost every pair of \( q \)-colorings \( \sigma, \sigma' \) we can transform \( \sigma \) into \( \sigma' \) by first transforming \( \sigma \) to \( \tau \) and then reversing the path from \( \sigma' \) to \( \tau \).

We proceed as follows. The algorithm GREEDY RE-COLOR takes as input: (i) the coloring \( \sigma \) and (ii) a specific subset of \( q_0 \) colors from \( \{1, ..., q\} \) that are not used in \( \tau \). W.h.p. it transforms the input coloring into a coloring using only those \( q_0 \) colors. Then we process the color classes of \( \tau \), re-coloring vertices to their \( \tau \)-color. When we process a color class \( C \) of \( \tau \), we switch the color of vertices in \( C \) to their \( \tau \)-color \( i_C \) one vertex at a time. We can do this because when we re-color a vertex \( v \), a neighbor \( w \) will currently either have one of the \( q_0 \) colors used by GREEDY RE-COLOR and these are distinct from \( i_C \). Or \( w \) will have already been re-colored with its \( \tau \)-color which will not be color \( i_C \). This proves Theorem 1.1.
References


