SMOOTHED ANALYSIS FOR GRAPH ISOMORPHISM

MICHAEL ANASTOS, MATTHEW KWAN, AND BENJAMIN MOORE

Abstract. There is no known polynomial-time algorithm for graph isomorphism testing, but elementary combinatorial "refinement" algorithms seem to be very efficient in practice. Some philosophical justification for this phenomenon is provided by a classical theorem of Babai, Erdős and Selkow: an extremely simple polynomial-time combinatorial algorithm (variously known as "naïve refinement", "naïve vertex classification", "colour refinement" or the "1-dimensional Weisfeiler–Leman algorithm") yields a so-called canonical labelling scheme for "almost all graphs". More precisely, for a typical outcome of a random graph $\mathbb{G}(n,1/2)$, this simple combinatorial algorithm assigns labels to vertices in a way that easily permits isomorphism-testing against any other graph.

We improve the Babai–Erdős–Selkow theorem in two directions. First, we consider *randomly perturbed* graphs, in accordance with the *smoothed analysis* philosophy of Spielman and Teng: for any graph G, naïve refinement becomes effective after a tiny random perturbation to G (specifically, the addition and removal of $O(n\log n)$ random edges). Actually, with a twist on naïve refinement, we show that $O(n)$ random additions and removals suffice. These results significantly improve on previous work of Gaudio, Rácz and Sridhar (resolving one of their conjectures), and are in certain senses best-possible.

Second, we complete a long line of research on canonical labelling and automorphisms for random graphs: for any p (possibly depending on n), we prove that a random graph $\mathbb{G}(n,p)$ can typically be canonically labelled in polynomial time. This is most interesting in the extremely sparse regime where p has order of magnitude c/n; denser regimes were previously handled by Bollobás, Czajka–Pandurangan, and Linial–Mosheiff. Our proof also provides a description of the automorphism group of a typical outcome of $\mathbb{G}(n,p)$ (slightly correcting a prediction of Linial–Mosheiff).

1. Introduction

Given a pair of graphs G_1 and G_2 (on the same vertex set $\{1,\dots,n\}$, say), how can we test whether they are isomorphic? Perhaps the most obvious first thought is to try to identify some easily-computable isomorphism-invariant information that distinguishes the two graphs. For example, one can easily compute the degrees of the vertices of G_1 and the degrees of the vertices in G_2, sort these lists and check if they are the same. If they are different, we have successfully determined that the graphs are not isomorphic (and if they are the same our test was inconclusive).

Perhaps the most influential approach along these lines is called *colour refinement*, also known as *naïve refinement* or the *1-dimensional Weisfeiler–Leman algorithm*[1]. This is an algorithm that produces a "colour" for each vertex; the colour of a vertex describes the degree of that vertex, together with all other data that one can obtain by allowing degree information to "percolate through the graph". We will define the algorithm formally in Definition 2.2 (and give an example in Figure 1), but briefly: at the start of the algorithm, every vertex "looks the same". Then, in the first step we distinguish vertices by their degrees (i.e., the colour of each vertex is its degree). In the second step, we distinguish vertices by their *number of neighbours with each degree* (i.e., each vertex now has a colour consisting of its own degree, together with a multiset of the degrees of its neighbours). In general, at each step we update the colour of each vertex by appending the multiset of colours of its neighbours. After some number of iterations of this procedure, it will "stabilise" in the sense that no further vertices can be distinguished from each other.

It turns out that the colour refinement algorithm can be executed very efficiently: in a graph with n vertices, the stable colouring can be computed[2] in time $O(n^2\log n)$. It is not hard to see that *if* the algorithm manages to assign each vertex of G a *distinct label*, then the sequence of colours in the stable

All authors were supported by ERC Starting Grant "RANDSTRUCT" No. 101076777. Michael Anastos was also supported in part by the Austrian Science Fund (FWF) [10.55776/ESP3863424] and by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 101034413. For Open Access purposes, the authors have applied a CC BY public copyright license to any author accepted manuscript version arising from this submission.

[1] The origin of this idea is difficult to pin down, but it seems to have been first proposed by Morgan [64] in 1965 in the context of computational chemistry!

[2] We are brushing over some subtleties here, which we will discuss further in Section 2.

colouring uniquely determines the isomorphism class of G. In fact, more is true: in this case one can use the colours to define a *canonical labelling* of G. This notion will be defined formally in Definition 2.1, but roughly speaking it means that one can label the vertices of G with the integers $\{1, \dots, n\}$ (assuming G has n vertices), in such a way that isomorphic graphs are always labelled the same way.

Unfortunately, colour refinement is not always so effective. For example, in a regular graph, where every vertex has the same degree, colour refinement is useless on its own, as it cannot distinguish any vertices from each other. However, a landmark result of Babai, Erdős and Selkow [8] shows that this situation is "atypical": the proportion of n-vertex graphs which cannot be canonically labelled using colour refinement tends to zero as $n \to \infty$ (see also the improvements in [9, 52, 61]). This result is most easily stated in the language of random graphs, as follows.

Theorem 1.1. *For a random graph*[3] $G \sim \mathbb{G}(n, 1/2)$*, whp*[4] *the colour refinement algorithm distinguishes all vertices from each other. In particular, whp G can be tested for isomorphism with any other graph in polynomial time.*

Theorem 1.1 is widely touted as philosophical justification for why algorithms based on colour refinement seem to be so effective in practice. Indeed, all graph isomorphism algorithms in common usage employ a modification of colour refinement called *individualisation-refinement*[5]. It is well-known that algorithms of this type take exponential time on worst-case inputs (see [67] for recent general results in this direction), but one rarely seems to encounter such inputs in practice. Of course, we would be remiss not to mention that from a theoretical point of view, fully sub-exponential algorithms are now available: there is an amazing line of work due to Babai, Luks, Zemlyachenko and others (see [6] for a survey) applying deep ideas from group theory to the graph isomorphism problem, which culminated in Babai's recent quasipolynomial-time graph isomorphism [5] and canonical labelling [7] algorithms.

Remark. The study of the colour refinement algorithm is of interest beyond its direct utility in graph isomorphism testing. Indeed, if graphs G and H are indistinguishable by colour refinement, we say that G and H are *fractionally isomorphic*; this is an important notion of intrinsic interest in graph theory, that has surprisingly many equivalent formulations (e.g., in terms of first-order logic [23], universal covers [2], tree homomorphism counts [34] and doubly-stochastic similarity of adjacency matrices [76]). See for example the surveys [41, 69] and the monograph [14] for more.

We also remark that the colour refinement algorithm represents the limit of so-called *graph neural networks* for graph isomorphism testing [66, 80]; these connections have recently been of significant interest in the machine learning community (see [59, 65] for surveys).

1.1. **Smoothed analysis.** As our first main direction in this paper, we take the philosophy of Theorem 1.1 much further, combining it with the celebrated *smoothed analysis* framework of Spielman and Teng [74]. To give some context: the *simplex algorithm* for linear optimisation is another fundamental example of an algorithm which seems to perform well in practice but takes exponential time in the worst case. As a very strong explanation for this, Spielman and Teng proved that if one takes *any* linear optimisation problem and applies a slight (Gaussian) random perturbation to its coefficients, then for a typical outcome of the resulting perturbed linear optimisation problem, the simplex algorithm succeeds in polynomial time. This shows that poorly-performing instances are "fragile" or "unstable", and perhaps we should not expect them to appear in practice.

Linear optimisation is a continuous problem, and for discrete problems one needs a different notion of random perturbation. In a later paper, Spielman and Teng [73] suggested that the most natural way to define a discrete random perturbation is in terms of *symmetric difference*: for graphs G, G' on the same vertex set, write $G \triangle G'$ for the graph containing all edges which are in exactly one of G and G' (we can think of G' as the "perturbation graph", specifying where we should "flip" edges of G to non-edges, or vice versa). Significantly strengthening Theorem 1.1, we prove that for *any* graph, randomly perturbing each edge with probability about $\log n/n$ (i.e., adding and removing about $n \log n$ random edges[6]) is sufficient to make colour refinement succeed whp.

[3] In the random graph $\mathbb{G}(n, p)$ (called the *binomial* or sometimes the *Erdős–Rényi* random graph), we fix a set of n vertices and include each of the $\binom{n}{2}$ possible edges with probability p independently.

[4] We say a property holds *with high probability*, or "whp" for short, if it holds with probability tending to 1. Here and for the rest of the paper, all asymptotics are as $n \to \infty$.

[5] It will not be relevant for the present paper to formally describe this paradigm, but for the curious reader: the idea is that occasionally a vertex must be artificially distinguished from the other vertices of its colour in order to "break regularity" (one must then consider all possible ways to make this artificial choice).

[6] Note that $\mathbb{G}(n, p)$ tends to have about $p\binom{n}{2} \approx pn^2$ edges.

Theorem 1.2. *Fix a constant $\varepsilon > 0$ and consider any $p \in [0, 1/2]$ satisfying $p \geq (1 + \varepsilon) \log n/n$. For any graph G_0, and with $G_{\text{rand}} \sim \mathbb{G}(n, p)$, whp the colour refinement algorithm distinguishes all vertices of $G_0 \triangle G_{\text{rand}}$ from each other. In particular, whp $G_0 \triangle G_{\text{rand}}$ can be tested for isomorphism with any other graph in polynomial time.*

Note that the $p = 1/2$ case of Theorem 1.2 is precisely Theorem 1.1. Indeed, when $p = 1/2$, the random perturbation is so extreme that all information from the original graph is lost and we end up with a purely random graph. (We may restrict our attention to $p \leq 1/2$, because perturbing G with a random graph $\mathbb{G}(n, p)$ is the same as perturbing the complement of G with a random graph $\mathbb{G}(n, 1-p)$.)

We remark that we are not the first to consider smoothed analysis for graph isomorphism: this setting was also recently considered by Gaudio, Rácz and Sridhar [39], though only under quite restrictive conditions on G_0, and with a stronger assumption on p (both of which make the problem substantially easier). Specifically, they found an efficient canonical labelling scheme that succeeds whp as long as G_0 satisfies a certain "sparse neighbourhoods" property, and as long as p has order of magnitude between $(\log n)^2/n$ and $(\log n)^{-3}$. Actually, Theorem 1.2 confirms a conjecture in [39] (that randomly perturbed graphs $G_0 \triangle G_{\text{rand}}$ can be efficiently canonically labelled when $G_{\text{rand}} \sim \mathbb{G}(n, p)$ with $p \geq (1+\varepsilon) \log n/n$).

It is not hard to see that the assumption on p in Theorem 1.2 cannot be significantly improved. Indeed, if G_0 is the empty graph and $p \leq (1 - \varepsilon) \log n/n$, then $G_0 \triangle G_{\text{rand}}$ is likely to have many isolated vertices (see e.g. [36, Theorem 3.1]), which cannot be distinguished by colour refinement. This does not necessarily mean that colour refinement fails to provide a canonical labelling scheme (isolated vertices can be labelled arbitrarily), but to illustrate a more serious problem, suppose G_0 is a disconnected graph in which every component is a 3-regular graph on at most 10 vertices. If we perturb with edge probability $p \leq (1/20) \log n/n$, it is not hard to see that whp many of the components of G will be completely untouched by the random perturbation (and therefore colour refinement will be unable to distinguish the vertices in these components, despite the components potentially having very different structure).

Of course, while tiny regular components may foil colour refinement, they are not really a fundamental problem for canonical labelling (as we can afford to use very inefficient canonical labelling algorithms on these tiny components). We are able to go beyond Theorem 1.2 via a modification of the colour refinement algorithm in this spirit. Indeed, with a variation on the colour refinement algorithm called the *2-dimensional Weisfeiler–Leman algorithm*, and a notion of a *disparity graph* (which allows us to define an appropriate generalisation of the notion of a "tiny component"), together with an exponential-time canonical labelling algorithm of Corneil and Goldberg [26] (only applied to the analogues of "tiny components"), we are able to define a combinatorial canonical labelling scheme which becomes effective after extremely mild random perturbation (perturbation probability about $1/n$).

Theorem 1.3. *There is a set of graphs $\mathcal{H}$, and an explicit polynomial-time canonical labelling algorithm for graphs in $\mathcal{H}$ (which can also detect* whether *a graph lies in $\mathcal{H}$), such that the following holds.*

Consider any $p \in [0, 1/2]$ satisfying $p \geq 100/n$. For any graph G_0, and $G_{\text{rand}} \sim \mathbb{G}(n, p)$, whp $G_0 \triangle G_{\text{rand}} \in \mathcal{H}$. In particular, whp $G_0 \triangle G_{\text{rand}}$ can be tested for isomorphism with any other graph in polynomial time.

We highlight that the algorithm for Theorem 1.3 is a "combinatorial" algorithm, using completely elementary refinement/recursion techniques (in particular, the algorithm can be interpreted as falling into the *individualisation-refinement* paradigm, and it does not use any group theory). Actually, up to constant factors, the restriction $p \geq 100/n$ in Theorem 1.3 is essentially best possible for combinatorial algorithms, given state-of-the-art worst-case guarantees: if we were to take $G_{\text{rand}} \sim \mathbb{G}(n, p)$ for $p = o(1/n)$ (i.e., if we were to take a milder random perturbation than in Theorem 1.3), then any polynomial-time canonical labelling algorithm that succeeds whp for graphs of the form $G_0 \triangle G_{\text{rand}}$ would immediately give rise[7] to a sub-exponential-time canonical labelling algorithm for *all* graphs (i.e., an algorithm that runs in time $e^{o(n)}$). Although such algorithms are known to exist (most obviously, we already mentioned Babai's quasipolynomial time algorithm), they all fundamentally use group theory.

The proof of Theorem 1.3 involves a wide range of different ideas, which we outline at some length in Section 1.3. As an extremely brief summary: we first use expansion and anticoncentration estimates, together with a characterisation of colour refinement in terms of *universal covers*, to prove that the colour refinement algorithm (applied to $G_0 \triangle G_{\text{rand}}$) typically assigns distinct colours to the vertices in

[7] For any graph G^*, we can consider a larger graph G_0 with many copies of G^* as connected components, with parameters chosen such that a very mild random perturbation typically leaves some copies of G^* completely unaffected (and therefore a canonical labelling of the randomly perturbed graph can be translated into a canonical labelling of G^*). We omit the details.

the so-called *3-core* of G_{rand}. This is already enough to prove Theorem 1.2, but to prove Theorem 1.3 (with its weaker assumption on p), we need to combine this with a delicate *sprinkling* argument, in which we slowly reveal the edges of G_{rand}, and study how the 3-core changes during this process. Every time a new vertex joins the 3-core (thereby receiving a unique colour by the colour refinement algorithm), this new colour information cascades through the 2-dimensional Weisfeiler–Leman algorithm, breaking up many of the colour classes into smaller parts. This has the effect of partitioning the graph into smaller and smaller parts, which can be treated separately at the end.

1.2. **Sparse random graphs.** After the Babai–Erdős–Selkow theorem (Theorem 1.1) on canonical labelling for $\mathbb{G}(n, 1/2)$, one of the most obvious directions for further study was to consider sparser random graphs $\mathbb{G}(n, p_n)$ (note that it suffices to consider $p_n \leq 1/2$, since canonical labelling is not really affected by complementation).

The first work in this direction was by Bollobás (see [18, Theorem 3.17]), who showed that the proof approach for Theorem 1.1 works as long as p_n does not decay too rapidly with n. Combining this with a later result of Bollobás [17] (which considered a very different type of canonical labelling scheme, under different assumptions on p_n), and a result of Czajka and Pandurangan [28] (which considered colour refinement for an intermediate range of p_n), one obtains polynomial-time canonical labelling schemes as long as $np_n - \log n \to \infty$ as $n \to \infty$. This condition on p_n is significant because it is the same range where random graphs are typically *rigid*: as was famously proved by Erdős and Rényi [35] (see also [79]), such random graphs typically have only trivial automorphisms, whereas if $np_n - \log n \to -\infty$ then there typically exist many automorphisms.

Even sparser graphs were recently considered by Linial and Mosheiff [60]; they handled the regime where $np_n \to \infty$ using another different type of canonical labelling scheme. The regime $np_n \to 0$ is easy, as in this regime (see [36, Section 2.1]) the components of $\mathbb{G}(n, p_n)$ are all trees with size at most $o(\log n)$ (so various different types of trivial canonical labelling schemes suffice; see for example [4]). That is to say, the only regime left unaddressed is the regime where p_n is about c/n for some constant c.

As our next main result, we close this gap, finding a canonical labelling scheme for all p_n.

Theorem 1.4. *There is a set of graphs $\mathcal{H}$, and an explicit polynomial-time canonical labelling algorithm for graphs in $\mathcal{H}$ (which can also detect* whether *a graph lies in $\mathcal{H}$), such that the following holds.*

For any sequence $(p_n)_{n\in\mathbb{N}} \in [0, 1]^{\mathbb{N}}$, and $G_n \sim \mathbb{G}(n, p_n)$, whp $G_n \in \mathcal{H}$. In particular, whp G_n can be tested for isomorphism with any other graph in polynomial time.

In fact, we prove that colour refinement on its own yields a canonical labelling scheme whp, unless p_n has order of magnitude c/n. If p_n has order of magnitude c/n, then colour refinement still *almost* works; the only obstruction is connected components which have a single cycle, which can be easily handled separately in a number of different ways.

Remark 1.5. There are important connections between the colour refinement algorithm and mathematical logic (see for example the monograph [40]). Our proof shows that random graphs of any density typically have *Weisfeiler–Leman dimension* at most 2 (and if we delete components with a single cycle, the Weisfeiler–Leman dimension is exactly equal to 1). That is to say, random graphs of any density can typically be uniquely distinguished by the 2-variable fragment of first-order logic with counting quantifiers.

Remark 1.6. An equivalent version of Theorem 1.4, in the regime $p_n = O(1/n)$, has been independently proved in concurrent work by Oleg Verbitsky and Maksim Zhukovskii [78] (they also study the colour refinement algorithm, but with quite different methods: while we take advantage of our general machinery already used to prove Theorem 1.3, Verbitsky and Zhukovskii take advantage of structural descriptions of the "anatomy" of a sparse random graph [31,32]). We believe both proof approaches to be of independent interest.

Given the machinery developed to prove Theorem 1.3, the proof of Theorem 1.4 is rather simple. Special care needs to be taken in the regime where p is very close to $1/n$ (this is the critical regime for the *phase transition* of Erdős–Rényi random graphs).

A related problem is to characterise the automorphism group of a random graph. Linial and Mosheiff achieved this when $np_n \to \infty$, and asked about the case where p_n has order of magnitude $1/n$. Specifically, they wrote that they "suspect" that the largest 2-connected component of the 2-core has trivial automorphism group. We prove that Linial and Mosheiff's suspicion is mostly (but not exactly) correct. To state our theorem in this direction, we need a further definition.

Definition 1.7. For a graph G, let $\mathrm{core}_k(G)$ be its k-core (its largest subgraph with minimum degree at least k). A *bare path* in a graph is a path whose internal vertices have degree 2. A *closed* bare path is a bare path between a vertex and itself (strictly speaking, this is actually a cycle, not a path).

Note that any graph G can always be obtained by gluing a (possibly trivial) rooted tree to each vertex of $\mathrm{core}_2(G)$, and adding some tree components. As observed by Linial and Mosheiff, in order to characterise the automorphisms of G, it suffices to characterise how such automorphisms act on $\mathrm{core}_2(G)$. Indeed, having specified how an automorphism acts on $\mathrm{core}_2(G)$, all that remains is to specify automorphisms of the rooted trees attached with each vertex, and the tree components. (Note that automorphisms of rooted trees are easy to characterise; we can only permute vertices at the same depth, and only if their corresponding subtrees are isomorphic).

Theorem 1.8. *Consider any sequence $(p_n)_{n\in\mathbb{N}}$ and let $G \sim \mathbb{G}(n, p_n)$. Then, G satisfies the following property whp. Every automorphism of G fixes the vertices of the $\mathrm{core}_2(G)$, with the following (possible) exceptions.*

- *For every cycle component of $\mathrm{core}_2(G)$, automorphisms of this cycle may give rise to automorphisms of G.*
- *For every pair of cycle components in $\mathrm{core}_2(G)$ which have the same length, there may be an automorphism of G which exchanges these two cycles.*
- *For a pair of vertices u, v with degree at least 3 in $\mathrm{core}_2(G)$, such that (in $\mathrm{core}_2(G)$) there are multiple bare paths between u and v, there may be an automorphism of G which exchanges these bare paths.*
- *For a vertex u with degree at least 3 in $\mathrm{core}_2(G)$, such that (in $\mathrm{core}_2(G)$) there is a (closed) bare path from u to itself, there may be an automorphism of G which "flips" this bare path (reversing the order of the internal vertices).*

If $np_n \to \infty$, it is not hard to see that whp none of the above exceptions actually occur (and therefore Theorem 1.8 generalises the main result in [60], which states that the 2-core has no nontrivial automorphisms in this regime). Indeed, in the 2-core, whp: there is no pair of bare paths of the same length between any pair of degree-3 vertices, there are no bare paths from a vertex to itself, and there are no cycle components. However, when say $p_n = 2/n$, there is a non-negligible probability that each of the aforementioned configurations exist (and that the rooted trees attached to these configurations permit automorphisms of G which permute vertices of $\mathrm{core}_2(G)$): the asymptotic distribution of the numbers of each of these configurations can be described by a sequence of independent Poisson random variables with nonzero means. This can be shown by the method of moments (see for example [47, Section 6.1]); we omit the details.

Remark 1.9. In the independent work of Verbitsky and Zhukovskii mentioned in Remark 1.6, they also deduced Theorem 1.8. They also went on to characterise all the automorphisms of $\mathrm{core}_2(G)$ (not just those induced by automorphisms of G); this requires some additional work.

1.3. **Key proof ideas.**

Before presenting the full proofs of Theorems 1.2 to 1.4 and 1.8, we take a moment to describe some of the key ideas in the proofs.

1.3.1. *Exploring universal covers.*

The starting point for all the proofs in this paper is an observation essentially due to Angluin [2], that the stable colouring obtained by the colour refinement algorithm assigns two vertices the same colour if and only if the *universal covers* rooted at those vertices are isomorphic. We will use a minor modification of the notion of a universal cover called a *view*, which encodes the same information but which is slightly more convenient for our purposes. Roughly speaking, the view $\mathcal{T}_G(v)$ rooted at a vertex v in a graph G is a (potentially infinite) tree encoding all possible walks in G starting from v (see Section 3 for a precise definition).

Without random perturbation, we are helpless to deal with the fact that many vertices may have isomorphic views (e.g., if G is d-regular, then $\mathcal{T}_G(v)$ is always isomorphic to the infinite rooted tree where each vertex has d children). The power of random perturbation is that, if two vertices u and v "see different vertices in their walks" (for example, if the set of neighbours of u is very different from the set of neighbours of v), then the random perturbation is likely to affect $\mathcal{T}_G(u)$ and $\mathcal{T}_G(v)$ differently, distinguishing them from each other. One of the key technical results in this paper is a general lemma (Proposition 4.1) which makes this precise.

1.3.2. *Distinguishing vertices via random perturbation: expansion and anticoncentration.* It is difficult to give a quick summary of Proposition 4.1 without the necessary definitions, but to give a flavour: we define sets $\mathcal{S}^i(\{u,v\})$ which describe the vertices which "feature differently" in length-i walks starting from u and length-i walks starting from v. Proposition 4.1 says that even extremely mildly randomly perturbed graphs typically have the property that if for some vertices u,v, and some i, the set $\mathcal{S}^i(\{u,v\})$ contains more than about $\log n$ vertices, then u and v are assigned different colours by colour refinement.

To prove this, we first use the expansion properties of random graphs (via a coupling of the type often appearing in analysis of branching processes) to "boost" the condition on $\mathcal{S}^i(\{u,v\})$, showing that if $\mathcal{S}^i(\{u,v\})$ has more than about $\log n$ vertices then there is typically some $j \geq i$ such that $\mathcal{S}^j(\{u,v\})$ has $n^{1-o(1)}$ vertices.

There is then a huge amount of "space" in $\mathcal{S}^j(\{u,v\})$ to take advantage of fluctuations due to the random perturbation. In particular, we show how to algorithmically divide $\mathcal{S}^j(\{u,v\})$ into "buckets" with different degree statistics, use the edges within these buckets to describe certain fluctuations in $\mathcal{T}_G(u)$ and $\mathcal{T}_G(v)$ via certain essentially independent inhomogeneous random walks on $\mathbb{Z}$, and then apply an anticoncentration inequality (a variant of the classical Erdős–Littlewood–Offord inequality) to show that it is extremely unlikely that these random walks behave in the same way for u and v (so unlikely that we can union bound over all u,v).

Of course, in order to actually apply Proposition 4.1, we need to study the pairs u,v for which $\mathcal{S}^i(\{u,v\})$ has more than about $\log n$ vertices (for some i). If the perturbation probability is greater than about $\log n/n$ (as in Theorem 1.2), it is easy to show that whp *all* pairs of vertices have this property (in this case we can even take $i=1$). For milder random perturbation (or sparser random graphs), we will need to restrict our attention to certain pairs u,v lying in certain special subgraphs, as we discuss next.

1.3.3. *The 2-core and the kernel.* Recall from Definition 1.7 that the k-*core* of a graph G is its maximal subgraph of minimum degree at least k. The *kernel* of G is the smallest multigraph homeomorphic to the 2-core of G (i.e., with the same topological structure as the 2-core). It can be obtained from the 2-core by iteratively replacing bare paths (also defined in Definition 1.7) by single edges.

Especially in the setting of random graphs, cores and kernels are objects of fundamental interest, typically possessing extremely strong expansion properties. In particular, the kernel of a random graph in some sense describes its fundamental underlying expander structure: a celebrated theorem of Ding, Lubetzky and Peres [32] shows that one can in some sense "build a random graph from its kernel" by (very informally) starting from a random expander (the kernel), randomly replacing some edges with bare paths, (to obtain the 2-core), and randomly affixing some trees.

If $k \geq 3$, the expansion properties of the k-core make it quite convenient to apply Proposition 4.1: it is fairly simple to show that, for $k \geq 3$, a mildly randomly perturbed graph typically has the property that colour refinement assigns a unique colour to all vertices of the k-core of G_{rand}. This immediately implies Theorem 1.2 (since when $p \geq (1+\varepsilon)\log n/n$ the 3-core of G_{rand} typically comprises the entire vertex set).

However, in very sparse regimes (in particular, when $p < c/n$ for a certain constant c, famously computed by Pittel, Spencer and Wormald [70]), the 3-core is typically *empty*, and we are forced to turn to the 2-core and the kernel. Unfortunately, this makes everything much more delicate. We prove a somewhat technical lemma (Proposition 5.2) showing that a mildly randomly perturbed graph typically has the property that colour refinement can distinguish the vertices[8] of degree at least 3 in the 2-core.

1.3.4. *Sparse random graphs.* The above considerations on the 2-core apply when $p \geq (1+\varepsilon)/n$ (for any constant $\varepsilon > 0$). Considering the case where the initial graph G_0 is empty, it is straightforward to deduce that simple canonical labelling schemes are typically effective for sparse random graphs $\mathbb{G}(n,p)$, when $p \geq (1+\varepsilon)/n$ (thus proving Theorem 1.4 in this regime).

The significance of this assumption on p is that it corresponds to the celebrated *phase transition* for Erdős–Rényi random graphs: when the edge probability is somewhat above $1/n$, there is typically a *giant component* with good expansion properties, but when the edge probability is somewhat less than $1/n$, there are typically only tiny components with very poor expansion properties (see for example the monographs [18, 36, 47] for more).

In the critical regime $(1-\varepsilon)/n < p < (1+\varepsilon)/n$, we proceed differently. In this regime, every component of $\mathbb{G}(n,p)$ with more than one cycle has quite large *diameter*: exploration processes can run

[8] This is not strictly speaking true; for the purposes of this outline we are ignoring a technical caveat concerning very rare configurations of edges in the 2-core.

for quite a long time without exhausting all the vertices in the graph, and we can accumulate quite a lot of independent randomness over this time. The anticoncentration from this randomness gives us another way to prove that different vertices are assigned different colours by colour refinement (completing the proof of Theorem 1.4). The details of this argument appear in Section 7.

1.3.5. *Sprinkling via the 3-core.* For Theorem 1.3 (on canonical labelling of very mildly randomly perturbed graphs), the role of the k-core (and Proposition 5.2, discussed in Section 1.3.3) is that it provides a kind of "monotonicity" that allows us to use a technique called *sprinkling*.

For the unfamiliar reader: sprinkling, in its most basic form, is the observation that a random graph $G_{\mathrm{rand}} \sim \mathbb{G}(n,p)$ can be interpreted as the union of two independent random graphs $G^1_{\mathrm{rand}} \cup G^2_{\mathrm{rand}}$, where $G^1_{\mathrm{rand}}, G^2_{\mathrm{rand}} \sim \mathbb{G}(n,p')$ with $1-p=(1-p')^2$. This observation allows one to first show that certain properties hold whp for G^1_{rand}, then reveal an outcome of G^1_{rand} satisfying these properties, and use the independent randomness of G^2_{rand} to "boost" these properties.

Sprinkling only really makes sense when we are dealing with properties of G_{rand} that are *monotone*, in the sense that once we have established the property for some subgraph of G_{rand}, adding the remaining edges of G_{rand} cannot destroy the property. Unfortunately, the colour refinement algorithm is highly non-monotone: in general, adding additional edges can allow the algorithm to distinguish more vertices, but can also prevent the algorithm from distinguishing some vertices. In the proof of Theorem 1.3, the critical role played by Proposition 5.2 is that it makes a connection between the colour refinement algorithm and the k-core, which is a fundamentally monotone object. For example, if a vertex is in the k-core of G^1_{rand}, then it is guaranteed to be in the k-core of $G^1_{\mathrm{rand}} \cup G^2_{\mathrm{rand}}$.

We will actually split our random perturbation G_{rand} into many independent random perturbations $G^1_{\mathrm{rand}}, \dots, G^T_{\mathrm{rand}} \in \mathbb{G}(n,p')$ (in the proof of Theorem 1.3 we will take $T=8$). If $p' \geq 10/n$, then one can show that whp the first random perturbation G^1_{rand} already has a giant 3-core[9], and by Proposition 5.2 we can safely assume that each of the vertices in this 3-core will be assigned unique colours by the colour refinement algorithm (applied to the randomly perturbed graph $G = G_0 \triangle (G^1_{\mathrm{rand}} \cup \dots \cup G^T_{\mathrm{rand}})$, which we have not yet fully revealed). Then, in each subsequent random perturbation G^i_{rand}, for $i \geq 2$, additional vertices randomly join the 3-core, and we can assume that they will also receive unique colours.

The upshot is that, whatever properties we are able to prove about the stable colouring produced by the colour refinement algorithm, we can "boost" these properties by randomly assigning unique colours to some vertices (and studying how this new information propagates through the colour refinement algorithm). To describe the types of properties we are interested in, we need to define the notion of a *disparity graph*[10], as follows.

1.3.6. *Small components in disparity graphs.* Recall that Theorem 1.2 cannot hold for $p = o(\log n/n)$. The key obstruction to keep in mind is that if G_0 has many tiny regular connected components, then very mild random perturbation will leave some of these components untouched, and the colour refinement algorithm will not be able to distinguish the vertices in these components.

For the proof of Theorem 1.3, we therefore need an appropriate generalisation of the notion of "tiny component" (taking into account the fact that tiny components in the *complement* of G play the same role as tiny components of G). To this end, we introduce the notion of a *disparity graph*.

Definition 1.10. For a graph G, a set of colours Ω and a colouring $c: V(G) \to \Omega$, define the *majority graph* $M(G,c)$ (on the same vertex set as G) as follows. For any (possibly non-distinct) pair of colours $\omega, \omega' \in \Omega$:

- If at least half of the possible edges between vertices of colours ω and ω' are in fact present as edges of G, then $M(G,c)$ contains every possible edge between vertices of colours ω and ω'.
- Otherwise (if fewer than half of the possible edges between vertices of colours ω and ω' are present), $M(G,c)$ contains no edges between vertices of colours ω and ω'.

Then, define the *disparity graph* $D(G,c) = M(G,c) \triangle G$.

Informally speaking, the majority graph $M(G,c)$ is the best possible approximation to G among all graphs which are "homogeneous" between colour classes (for every pair of colour classes, to decide whether to put all edges between them or no edges between them, we look at the majority behaviour in G among vertices of those colours). Then, the disparity graph identifies the places where the majority graph differs

[9] With a little more work, it would suffice to consider the vertices of degree at least 3 in the 2-core, in which case we only need p' to be slightly larger than $1/n$. However, the 3-core is much more convenient to work with.

[10] This notion has previously been introduced under the name "flip graph" [53,54]; we thank the anonymous referees of *STOC 2025* for bringing this to our attention.

from G. Equivalently, we can define the disparity graph to be the graph obtained by considering every pair of colour classes and deciding whether to complement the edges between those colour classes or not, depending on which of the two choices would make the graph sparser.

If c is the stable colouring obtained from the colour refinement algorithm, it is not hard to show that a canonical labelling of $D(G, c)$ yields a canonical labelling of G (this is stated formally in Proposition 2.5 later in the paper). So, we can canonically label G in polynomial time whenever $D(G, c)$ has sufficiently small components (small enough that we can afford to use known inefficient canonical labelling schemes on each component).

1.3.7. *Percolation for a weaker result.* Let c be the stable colouring obtained by the colour refinement algorithm. To prove Theorem 1.3, it would suffice to prove that whp the disparity graph $D(G_0 \triangle G_{\mathrm{rand}}, c)$ has small components (for a polynomial-time combinatorial algorithm, we need every component to have $O(\log n)$ vertices, so that we can afford to use an exponential-time canonical labelling algorithm of Corneil and Goldberg [26] on each component[11]).

Unfortunately, we were not quite able to manage this when the random perturbation probability is $O(1/n)$ (as demanded by Theorem 1.3). Indeed, our proof of Theorem 1.3 requires a more sophisticated variant of the colour refinement algorithm, as we discuss later in this outline. However, the above goal is achievable if the random perturbation probability is at least (say) $100 \log\log n/n$. For expository purposes we next sketch how to prove this, before moving on to the more sophisticated ideas in the full proof of Theorem 1.3 (the details of this simpler argument can be found in Appendix B).

As discussed in Section 1.3.5, we can interpret our random perturbation G_{rand} as a composition of three random perturbations $G^1_{\mathrm{rand}} \cup G^2_{\mathrm{rand}} \cup G^3_{\mathrm{rand}}$. The first random perturbation G^1_{rand} already whp establishes a giant 3-core (of size at least $n/2$, say); we can assume that the vertices in this 3-core get unique colours (and are hence isolated vertices in the disparity graph), so we only need to worry about the vertices outside the 3-core. Our two additional random perturbations G^2_{rand} and G^3_{rand} each cause an independent random subset of vertices to receive unique colours (as they join the 3-core): if $p \geq 100 \log\log n/n$, we calculate that each vertex receives a unique colour with probability at least $1 - o(1/\log n)$.

With one round of sprinkling (i.e., with the random assignment of unique colours provided by G^2_{rand}, followed by the colour refinement algorithm) we can show that the disparity graph has maximum degree $O(\log n)$ whp[12]. Indeed, the disparity graph describes how the neighbourhood of a vertex differs from the "majority behaviour" among vertices of its colour class, so if there is a vertex with high degree in the disparity graph, then there is a pair of vertices in the same colour class with very different neighbourhoods. With a union bound over pairs of vertices, it is easy to show that no such pairs persist after a round of sprinkling (the random assignment of unique colours typically allows the colour refinement algorithm to distinguish all such pairs).

For our second round of sprinkling (provided by G^3_{rand}), we view the random assignment of unique colours as *percolation*: if a vertex gets a unique colour, then it becomes isolated in the disparity graph, and we can imagine that that vertex is deleted. We are interested in the connected components that remain after these deletions[13]. But if the disparity graph has degree $O(\log n)$ before sprinkling, and each vertex is deleted with probability $1 - o(1/\log n)$, one can show that these deletions usually shatter the disparity graph into small components (the necessary analysis is similar to analysis of subcritical branching processes). This percolation step is the part where we are fundamentally using that the random perturbation probability is at least about $\log\log n/n$: if the perturbation probability were smaller than this, then the deletions would not be severe enough to break the disparity graph into small components.

1.3.8. *Splitting colour classes and components.* In order to go beyond the ideas in the previous subsection, to work with random perturbation probabilities as small as $O(1/n)$, we need to get a much stronger conclusion from our sprinkled random perturbation. Every time a vertex is added to the 3-core (and gets a unique colour), this vertex is not simply removed from the disparity graph: the new colour information

[11] We remark that it does not actually seem to make the problem much easier if we weaken this requirement on the size of each component (which we may, if we are willing to use a more sophisticated group-theoretic algorithm, with better worst-case guarantees, on each connected component).

[12] For this bound we only need that (say) $p \geq 100/n$ (this implies that in each round of sprinkling, every vertex joins the 3-core with probability at least 0.9). Using that $p \geq 100 \log\log n/n$ (so in every round of sprinkling, every vertex joins the 3-core with probability at least $1 - o(1/\log n)$), we can actually show that the degrees are at most $O(\log n/\log\log n)$ whp. But, this stronger bound would not really affect the final result.

[13] Here we are sweeping under the rug some technical issues related to the "consistency" of the disparity graph as the underlying graph changes. This turns out to be quite delicate, and is handled in Lemma 6.3.

has a cascading effect (via the colour refinement algorithm) that affects the colours of many other vertices, indirectly affecting the connected components of the disparity graph.

First, it is instructive to think about the effect of sprinkling on the connected components (of the disparity graph) which intersect a given colour class C. If there is a vertex v which has a neighbour in C (with respect to the disparity graph), then if we were to assign v a unique colour this would cause C to break into multiple colour classes: namely, the colour refinement algorithm would be able to distinguish the vertices in C adjacent to v from the vertices in C which are not adjacent to v (it is a simple consequence of the definition of the disparity graph that v is adjacent to at most half the vertices in C).

So, we can consider an exploration process that starts from the vertices of C, and explores[14] all the vertices which share a connected component (in the disparity graph) with a vertex in C. Each time we consider a new vertex, we reveal whether it is assigned a unique colour, and propagate this information via the colour refinement algorithm[15]. As we continue this exploration/refinement process, the colour classes will start to break up into smaller pieces. There is a limit to how long this process can continue, since colour classes of size 1 cannot be broken up further. Indeed, by considering an auxiliary submartingale (which measures how the number of colours we have discovered so far compares to the number of vertices we have explored so far, at each point in the process), we can prove that our exploration process is likely to terminate after $O(|C|)$ steps, having explored all the vertices that share a component with a vertex in C. That is to say, the sizes of the connected components of the disparity graph after sprinkling are bounded in terms of the sizes of the colour classes before sprinkling. (We emphasise that in this discussion C always refers to our colour class *before* sprinkling, i.e., we are not varying C during the exploration process.)

Unfortunately, we cannot hope to show that the colour classes are small, in general (indeed, if G has many isolated vertices, then all these vertices will be assigned the same colour by any canonical vertex-colouring scheme). However, the above idea can be "localised" to a connected component: We prove a crucial lemma (Lemma 6.10), which tells us that in order to show that the disparity graph has connected components with $O(\log n)$ vertices (after sprinkling), it suffices to show that, before sprinkling, for all colour classes C and connected components X of the disparity graph, we have $|C \cap X| = O(\log n)$.

In order to show that these intersection sizes $|C \cap X|$ are small, we need another round of sprinkling which "shatters large components into small colour classes". In order for this sprinkling to have a strong enough effect, we need to consider a more powerful variant of the colour refinement algorithm, as follows.

1.3.9. *Distinguishing vertices via the 2-dimensional Weisfeiler–Leman algorithm.* The *2-dimensional Weisfeiler–Leman algorithm* refines colourings of *pairs* of vertices: starting with a certain "trivial" colouring $\phi_G : V(G)^2 \to \Omega$, we repeatedly refine ϕ_G based on statistics of 3-vertex configurations, until a stable colouring $f : V(G)^2 \to \{0,1\}$ is reached. This colouring of pairs of vertices then gives rise to a colouring of individual vertices $v \mapsto f(v,v)$, which contains a lot more information than the result of ordinary colour refinement.

In particular, this more sophisticated refinement operation allows us to distinguish vertices based on *distances*: in the final vertex-colouring, every two vertices of the same colour see the same number of vertices of every given colour at any given distance (in the disparity graph). As discussed in Section 1.3.7, after a single round of sprinkling whp the disparity graph has maximum degree $O(\log n)$, so if a connected component X has more than logarithmically many vertices then it is quite sparse, meaning that there is a very rich variety of different pairs of vertices at different distances. So, when sprinkling causes new vertices to receive unique colours, hopefully the 2-dimensional Weisfeiler–Leman algorithm will be able to significantly break up the colour classes in X (recall that our goal is now to prove that the intersections between colour classes and connected components have size $O(\log n)$).

Naïvely, one might hope to prove this via a simple union bound over subsets $Z \subseteq X$ of about $\log n$ vertices: for any such set Z, one might try to prove that after a round of sprinkling, and the 2-dimensional Weisfeiler–Leman algorithm, it is overwhelmingly unlikely that all vertices of Z have the same colour. To prove this, it would suffice to show that for any such Z there are many vertices which see some vertices of Z at different distances (so if any of these many vertices receive a new unique colour, this could be used by the 2-dimensional Weisfeiler–Leman algorithm to give different colours to some of the vertices of Z).

[14] For the purposes of this outline, the reader can imagine that at each step we choose the unexplored vertex which is closest to C in the current disparity graph, though our actual exploration process is a bit more complicated.

[15] The resulting changes to the colouring also change the disparity graph. In particular, vertices which used to be in the same connected component may later be spread over multiple connected components, but this is not really a problem.

Unfortunately, this direct approach does not seem to yield any nontrivial bounds, without making structural assumptions about G. Instead, we use a "fingerprint" technique reminiscent of the method of *hypergraph containers* (see [10, 71]) in extremal combinatorics, which reduces the scope of our union bound. Specifically, if there were a large colour class $C' \subseteq X$ after sprinkling, we show that this would imply the existence of a much smaller "fingerprint" set $S \subseteq X$ which sees many vertices (specifically, many vertices of C') at a variety of different distances. We can then take a cheaper union bound over the smaller fingerprint sets S.

We remark that the above sketch was very simplified, and serves only to illustrate the rough ideas. The full proof of Theorem 1.3 (which appears in Section 6) confronts a number of delicate technical issues and requires seven rounds of sprinkling, each of which gradually reduce the degrees, colour classes and connected components of the disparity graph.

1.4. **Further directions.** There are a very large number of natural directions for further research. First is the question of optimising the probability implicit in the "whp" in each of Theorems 1.2 to 1.4, and improving the runtimes of the relevant algorithms. In the setting of Theorem 1.1, both these issues were comprehensively settled by Babai and Kucera [9]: they showed that, except with exponentially small probability, two steps of colour refinement suffice (yielding a linear-time algorithm).

The algorithm in Theorem 1.2 (colour refinement) runs in time $O((n+m)\log n)$, where n and m are the numbers of vertices and edges of our graph of interest, while the algorithm in Theorem 1.4 naïvely runs in time $O(n(n+m)\log n)$, due to repeated iteration of the colour refinement algorithm. With some careful analysis, it may be possible to bound the necessary number of colour refinement steps, to obtain an optimal linear-time algorithm in the setting of Theorem 1.2. In fact, the work of Gaudio, Rácz and Sridhar [39], mentioned earlier in the introduction, already makes some initial steps in this direction: they essentially show that three steps of colour refinement suffice, as long as p has order of magnitude between $(\log n)^2/n$ and $(\log n)^{-3}$. It seems plausible that in fact three steps of colour refinement suffice as long as $p \geq (1+\varepsilon)\log n/n$, and recent work on "shotgun reassembly" [38, 48] indicates that there should be a phase transition between two steps of colour refinement being necessary, and three steps being necessary, at around $p = (\log n)^2(\log\log n)^{-3}/n$.

With similar considerations on the necessary number of colour refinement steps, and the necessary number of iterations of the colour refinement algorithm, it may also be possible to improve the runtime in the setting of Theorem 1.4 to near-linear-time. Regarding Theorem 1.3: this algorithm also naïvely runs in about quadratic-time, due to the use of the 2-dimensional Weisfeiler–Leman algorithm. However, we do not need the full power of the general algorithms that we cite, and it seems plausible that special-purpose variants could be designed that might also yield a near-linear-time algorithm.

Also, there is the possibility of improving on the amount of random perturbation in Theorem 1.3. In particular, taking advantage of group-theoretic techniques (e.g., using the quasipolynomial-time algorithm of Babai [7]), it might be possible to design a canonical labelling scheme that becomes effective with tiny amounts of random perturbation (e.g., perturbation probability $o(1/n)$). In the case where G_0 is a regular graph, it may also be worth considering alternative models of random perturbation that do not destroy the regularity of G_0, as colour refinement is completely ineffective for regular graphs. Perhaps surprisingly, this may actually make the problem *easier*, as sparse random regular graphs are much better expanders than sparse Erdős–Rényi random graphs (cf. the work of Bollobás [17] showing that distance profiles provide an efficient canonical labelling scheme even for very sparse random regular graphs).

Next, there is the possibility of stronger connections between our results on smoothed analysis for graph isomorphism, and the work of Spielman and Teng on smoothed analysis for linear optimisation. Indeed, (a slight variant of) the colour refinement algorithm is used for *dimension reduction* in linear optimisation (see [42]) and it would be interesting to consider smoothed analysis in this setting.

There is also the possibility of considering smoothed analysis for many different types of graph algorithms other than isomorphism-testing. Some early work in this direction was undertaken by Spielman and Teng [73] (see also [12, 13, 62, 63]), but somewhat surprisingly, despite the algorithmic origin of the smoothed analysis framework, there is now a much larger body of work on randomly perturbed graphs in extremal and probabilistic graph theory (see for example [1, 3, 11, 15, 16, 20–22, 29, 30, 43, 44, 49, 56–58]) than on algorithmic questions.

In particular, it is worth remarking that canonical labelling is closely related to the so-called "network alignment problem" (also known as the "graph matching problem"), in which the objective is to find a mapping between the vertex sets of two graphs such that the number of adjacency disagreements between the two graphs is minimised. See for example [33] (and the references therein) for recent work on alignment of random graphs; it would be interesting to explore to what extent this can be extended to the smoothed analysis stetting.

1.5. **Notation.** Our graph-theoretic notation is for the most part standard. For a graph G, we write $V(G)$ for its set of vertices. We write $G[S]$ for the subgraph of G induced by the vertex subset S, write $G[S,T]$ for the bipartite subgraph of G induced between S and T (assuming S,T are disjoint), and write $G\triangle G'$ for the *symmetric difference* of two graphs G,G' on the same vertex set (i.e., e is an edge of $G\triangle G'$ if it is an edge of exactly one of G,G'). We write $\operatorname{dist}_G(u,v)$ for the distance between two vertices u,v, and we write $\mathbb{G}(n,p)$ for the Erdős–Rényi random graph on n vertices with edge probability p.

We (slightly abusively) conflate vertex-colourings with vertex-partitions. In particular, for vertex-colourings $c: V(G)\to\Omega$ and $c': V(G)\to\Omega'$, we say that c is a *refinement* of c' (or c' is a *coarsening* of c) if each colour class of c is a subset of a colour class of c'.

Our use of asymptotic notation is standard as well. For functions $f=f(n)$ and $g=g(n)$, we write $f=O(g)$ to mean that there is a constant C such that $|f|\le C|g|$, $f=\Omega(g)$ to mean that there is a constant $c>0$ such that $|f|\ge c|g|$ for sufficiently large n, and $f=o(g)$ to mean that $f/g\to 0$ as $n\to\infty$. We say that an event occurs with high probability ("whp") if it holds with probability $1-o(1)$.

For a real number x, the floor and ceiling functions are denoted $\lfloor x\rfloor=\max\{i\in\mathbb{Z}: i\le x\}$ and $\lceil x\rceil=\min\{i\in\mathbb{Z}: i\ge x\}$. We will however sometimes omit floor and ceiling symbols and assume large numbers are integers, wherever divisibility considerations are not important. All logarithms in this paper are in base e, unless specified otherwise.

1.6. **Organisation of the paper.** The structure of the paper is as follows. We start with some preliminaries in Section 2; in particular, this section features the formal definitions of many of the notions informally discussed in this introduction. Some of the more routine proofs in this section are deferred to Appendix A.

Then, in Section 3 we introduce some machinery for working with *views*, and in Section 4 we use this machinery to prove a general lemma (Proposition 4.1) giving an expansion-type condition under which two vertices are typically assigned different colours by colour refinement.

In Section 5 we apply Proposition 4.1 to prove that (under suitable random perturbation) colour refinement typically assigns unique colours to all of the vertices of degree 3 in the 2-core of G_{rand} which satisfy a certain technical condition. We deduce (in Proposition 5.5) that colour refinement typically assigns unique colours to all the vertices in the 3-core of G_{rand}. This immediately implies Theorem 1.2.

In Section 6 we prove Theorem 1.3, combining the results in Section 5 with a large number of additional ideas. This section can be read independently of most of the rest of the paper (treating Proposition 5.5 as a black box). A subset of the ideas in Section 6 can also be used to give a much simpler proof of a slightly weaker result (assuming that the random perturbation probability p is at least about $\log\log n/n$); we include the details in Appendix B.

Finally, in Section 7 we prove Theorems 1.4 and 1.8, via the results in Section 5 and some additional considerations for near-critical random graphs.

1.7. **Acknowledgements.** We would like to thank Oleg Verbitsky and Maksim Zhukovskii for several insightful comments, and for alerting us to their independent work (see Remarks 1.6 and 1.9). The first two authors would also like to thank Marc Lelarge for insightful discussions which inspired some of the directions in this paper, and the second author would like to thank Oliver Riordan for interesting comments on the necessary number of colour refinement steps (see Section 1.4). Finally, we would like to thank the anonymous reviewers of previous versions of this paper, for their careful reading and a large number of comments and suggestions that have significantly improved the paper.

2. Basic notions and preliminary lemmas

In this section we formally define some of the notions informally discussed in the introduction, and we state and prove some general-purpose lemmas that will be used throughout the paper. In Section 2.1 we present some general definitions and results about canonical labelling, in Section 2.2 we collect some basic facts about colour refinement, in Section 2.3 we define the 2-dimensional Weisfeiler–Leman algorithm and collect some basic results about it, and in Section 2.4 we collect some probabilistic estimates.

2.1. **Canonical labelling.** We start by formally defining the notion of canonical labelling.

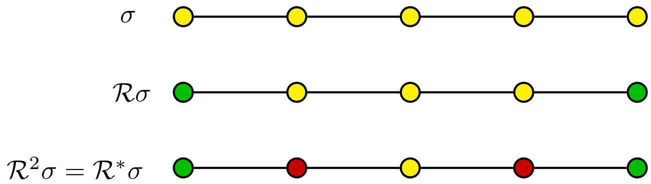


Figure 1. An illustration of the colour refinement algorithm on the 4-edge path.

Definition 2.1. Let $\mathcal{G}_n$ be the set of all graphs on the vertex set $\{1, \dots, n\}$, and let $\mathcal{S}_n$ be the set of all permutations of $\{1, \dots, n\}$. A *labelling scheme* is a map $\Phi : \mathcal{G}_n \to \mathcal{S}_n$. For a graph $G \in \mathcal{G}_n$, we write $\Phi((G)) \in \mathcal{G}_n$ for the graph obtained by applying the permutation $\Phi(G)$ to the vertices and edges of G.

We say that a family of graphs $\mathcal{F} \subseteq \mathcal{G}_n$ is *isomorphism-closed* if it is a disjoint union of isomorphism classes in $\mathcal{G}_n$. For an isomorphism-closed family of graphs $\mathcal{F} \subseteq \mathcal{G}_n$, we say that $\Phi : \mathcal{G}_n \to \mathcal{S}_n$ is a *canonical* labelling scheme for $\mathcal{F}$ if $\Phi((G)) = \Phi((G'))$ whenever $G, G' \in \mathcal{F}$ are isomorphic.

Intuitively speaking, one should think of a canonical labelling scheme as a way to assign labels to an *unlabelled* graph G, where we are not allowed to make "arbitrary choices" (e.g., arbitrarily breaking ties), unless different outcomes of those choices correspond to automorphisms of G.

Note that a canonical labelling scheme Φ for an isomorphism-closed family $\mathcal{F} \subseteq \mathcal{G}_n$ gives rise to a simple way to test for isomorphism between a graph $G \in \mathcal{F}$ and any other graph $G' \in \mathcal{G}_n$: simply compute $\Phi((G))$ and $\Phi((G'))$, and compare the two labelled graphs (e.g., by their adjacency matrices).

The primary engine underlying all canonical labelling schemes in this paper is the so-called *colour refinement algorithm*. It was informally introduced in the introduction; here we define it more formally.

Definition 2.2. For a graph G and a colouring $c : V(G) \to \Omega$, the *refinement* $\mathcal{R}_G c : V(G) \to \Omega \times \mathbb{N}^\Omega$ of c is defined by $\mathcal{R}_G c(v) = (c(v), (d_\omega(v))_{\omega \in \Omega})$, where $d_\omega(v)$ denotes the number of neighbours of v which have colour ω (with respect to c). Note that every colouring $c : V(G) \to \Omega$ defines a partition of $V(G)$ into colour classes. If c, c' define the same vertex partition, then $\mathcal{R}_G c, \mathcal{R}_G c'$ also define the same vertex partition. So, we can actually view $\mathcal{R}_G$ as an operator on vertex partitions.

Also, denote the t-fold iteration of $\mathcal{R}_G$ by

$$\mathcal{R}_G^t c = \underbrace{\mathcal{R}_G \dots \mathcal{R}_G}_{t \text{ times}} c.$$

Note that for any colouring c, the partition defined by $\mathcal{R}_G c$ is always a refinement of the partition defined by c. For a vertex set of size n, the longest possible chain of proper refinements has length n, so if we start with any colouring c in an n-vertex graph, and repeatedly apply the refinement operation $\mathcal{R}_G$ (at most n times), we always end up with a "stable colouring". That is to say, there is always some $t \le n$ such that $\mathcal{R}_G^t c$ and $\mathcal{R}_G^{t+1} c$ define the same vertex partition. We denote the corresponding stable colouring by $\mathcal{R}_G^* c = \mathcal{R}_G^t c$.

We will write $\sigma_G : V(G) \to \{0\}$ for the trivial colouring in which every vertex is given the same "0" colour.

So, $\mathcal{R}_G \sigma_G$ distinguishes vertices by their degree, and $\mathcal{R}_G^* \sigma_G$ is the stable colouring obtained by the colour refinement algorithm as described in the introduction. See Figure 1 for an example. For all the notation introduced in Definition 2.2, we will often omit the subscript "G" when it is clear from context.

If $\mathcal{R}^* \sigma$ assigns every vertex in G a distinct colour, then it is easy to obtain a canonical labelling for G from $\mathcal{R}^* \sigma$: we simply fix some total order of all the potential colours of vertices (e.g. lexicographical order), and assign the labels $\{1, \dots, n\}$ in increasing order. However, it is not completely obvious that this gives rise to an efficient canonical labelling scheme: naïvely, after many steps of refinement the sizes of the labels become extremely large. In order to address this, after each step of colour refinement we can re-encode all colours with some subset of the integers $\{1, \dots, n\}$ (i.e., we do not actually care about the identities of the colours, only the induced partition of the vertices), but we must be careful to do this "in a canonical way" (i.e., all choices can only depend on the isomorphism class of our graph). This issue is comprehensively handled by Berkholz, Bonsma and Grohe [14] (building on ideas in [24, 45, 68]), who present an algorithm that computes a canonical stable colouring of G in time $O((n + m) \log n)$ (where m is the number of edges in G).

Theorem 2.3. *Let $\mathcal{F} \subseteq \mathcal{G}_n$ be the set of all graphs on the vertex set $\{1, \dots, n\}$ for which $\mathcal{R}^* \sigma$ assigns each vertex a distinct colour. Then, the colour refinement algorithm defined in [14] is a canonical labelling* *scheme for $\mathcal{F}$. It computes a canonical labelling for a graph $G \in \mathcal{F}$ in time $O((n+m)\log n)$ (where m is the number of edges in G).*

We will use Theorem 2.3 to prove Theorem 1.2, but for Theorems 1.3 and 1.4 we will need some more robust lemmas on canonical labelling. First, recall the definition of the *disparity graph* from Definition 1.10. As mentioned in Section 1.3.6, we obtain an efficient canonical labelling scheme when the disparity graph can itself be efficiently labelled; we formalise this in Proposition 2.5 below, after introducing another definition.

Definition 2.4. A *canonical colouring scheme* $(c_G)_{G\in\mathcal{G}_n}$ is an assignment of a colouring $c_G : V(G) \to \Omega$ to each graph $G \in \mathcal{G}_n$, in such a way that for any graphs $G, H \in \mathcal{G}_n$, and any isomorphism ψ from G to H, we have $c_H(\psi(v)) = c_G(v)$ for all $v \in V(G)$.

Note that the colour refinement algorithm, starting from the trivial colouring, defines a canonical colouring scheme (i.e., let $c_G = \mathcal{R}^*_G\sigma$). However, in our proof of Theorem 1.3 we will end up needing a slightly more sophisticated canonical colouring scheme (to be defined in Section 2.3).

Proposition 2.5. *Let $\Phi_\mathcal{H}$ be a canonical labelling scheme for a graph family $\mathcal{H} \subseteq \mathcal{G}_n$, and let $(c_G)_{G\in\mathcal{G}_n}$ be a canonical colouring scheme, such that $\Phi_\mathcal{H}(G)$ and c_G can both be computed in time T.*

Let $\mathcal{F}$ be the family of all graphs $G \in \mathcal{G}_n$ such that $D(G, c_G) \in \mathcal{H}$. Then there is a canonical labelling scheme Φ for $\mathcal{F}$, such that for every $G \in \mathcal{F}$, we can compute $\Phi(G)$ in time $O(n^2 + T)$.

Proof. For a graph $G \in \mathcal{F}$, our canonical labelling $\Phi(G) \in \mathcal{S}_n$ is computed as follows. First compute the colouring c_G (in time T). Then, compute the disparity graph $D(G, c_G)$ (this can be done in time $O(n^2)$). To each vertex v associate the pair

$$\Big(c_G(v),\ \Phi_\mathcal{H}(D(G, c_G))(v)\Big).$$

Then, obtain $\Phi(G)$ by ordering the vertices lexicographically by their associated pairs. To see that this indeed describes a canonical labelling scheme, note that every automorphism of $D(G, c_G)$ which fixes the colouring c_G is an automorphism of G. □

We will need to combine Proposition 2.5 with an efficient canonical labelling scheme for graphs with small connected components (it is easy to deal with each connected component separately, so this really comes down to worst-case guarantees for graphs on a fixed number of vertices). There are algorithms based on group-theoretic ideas with very strong theoretical guarantees (due to recent work of Babai [7]), but as the purpose of this paper is to study the typical performance of *combinatorial* algorithms, we will instead use the following theorem of Corneil and Goldberg [26]: with an ingenious recursion scheme, it is possible to canonically label graphs in exponential time (improving on naïve factorial-time algorithms).

Theorem 2.6. *There is a canonical labelling scheme Φ for $\mathcal{G}_n$ such that for every $G \in \mathcal{G}_n$, we can compute $\Phi(G)$ in time $\exp(O(n))$.*

We record the following corollary of Proposition 2.5 and Theorem 2.6.

Corollary 2.7. *Fix a polynomial-time computable canonical colouring scheme $(c_G)_{G\in\mathcal{G}_n}$. Fix a constant C, and let $\mathcal{F}_C$ be the set of all graphs G on the vertex set $\{1, \dots, n\}$, for which $D(G, c_G)$, has connected components with at most $C \log n$ vertices. Then there is a polynomial-time-computable canonical labelling scheme for $\mathcal{F}_C$.*

For the reader unfamiliar with canonical labelling, we provide some details in Appendix A for how exactly to deduce Corollary 2.7 from Proposition 2.5 and Theorem 2.6 (i.e., how to handle connected components separately).

Next, for Theorem 1.4, we need a generalisation of Theorem 2.3 to a much larger class of graphs, as follows.

Definition 2.8. A graph G is *CR-determined* if the multiset of colours in $\mathcal{R}^*\sigma$ distinguishes G from all non-isomorphic graphs (i.e., if there is no other graph H, not isomorphic to G, for which colour refinement produces the same multiset of colours).

We remark that there is a combinatorial characterisation of CR-determined graphs (see [4,54]), though we will not need this.

Theorem 2.9 (see [4], Corollary 18)**.** *Let $\mathcal{F} \subseteq \mathcal{G}_n$ be the set of all CR-determined graphs on the vertex set $\{1, \dots, n\}$. Then there is a canonical labelling algorithm that computes a canonical labelling for a graph $G \in \mathcal{F}$ in polynomial time.*

The canonical labelling algorithm for Theorem 2.9 is a slight variation on the colour refinement algorithm (due to Tinhofer [77] and Immerman and Lander [46], and made canonical by Arvind, Köbler, Rattan and Verbitsky [4]; see also [54] for similar independent results). The idea is that if the colour refinement algorithm reaches a stable colouring in which two vertices have the same colour, we arbitrarily give one of those two vertices a unique colour, and continue.

To use Theorem 2.9 in our proof of Theorem 1.4, we will need a way to show that graphs are CR-determined. To this end, it *almost* suffices to study the vertices in the kernel of G (i.e., the vertices with degree at least 3 in the 2-core), as follows.

Definition 2.10. For a graph G, let $V_{2,3}(G)$ denote the vertices in the 2-core of G which have degree at least 3.

Proposition 2.11. *Let G be a connected graph with $V_{2,3}(G) \neq \emptyset$. If $\mathcal{R}^*\sigma$ assigns all vertices of $V_{2,3}(G)$ distinct colours, then G is CR-determined.*

It is fairly straightforward to prove Proposition 2.11 by reasoning carefully about how to reconstruct G given $\mathcal{R}^*\sigma$ (first put bare paths between vertices of $V_{2,3}(G)$ to obtain $\mathrm{core}_2(G)$, then attach trees to $\mathrm{core}_2(G)$ to obtain G). For completeness, we provide some details in Appendix A.

Remark 2.12. With the same considerations, it is straightforward to characterise the automorphisms of a graph G for which all vertices of $V_{2,3}(G)$ are assigned distinct colours. Indeed, the vertices of $V_{2,3}(G)$ are fixed by any automorphism, so the only flexibility comes from components with at most one cycle (i.e., components with no vertices of $V_{2,3}(G)$), bare paths between vertices of $V_{2,3}(G)$, and the trees attached to $\mathrm{core}_2(G)$. This is relevant for Theorem 1.8.

Proposition 2.11 does not tell us anything about graphs with $V_{2,3}(G) = \emptyset$, but these graphs are easy to handle separately. Indeed, every connected graph with $V_{2,3}(G) = \emptyset$ either has no cycles (i.e., is a tree) or has exactly one cycle. Such graphs have very simple structure, and there are many ways to handle them; for example, all such graphs are *outerplanar* (i.e., they have a planar embedding for which all vertices belong to the outer face of the embedding), so we can appeal to the following classical theorem of Sysło [75].

Theorem 2.13. *There is an $O(n)$-time algorithm to canonically label the family of n-vertex outerplanar graphs.*

Combining Theorems 2.9 and 2.13 and Proposition 2.11 yields the following corollary (the details of the deduction appear in Appendix A).

Corollary 2.14. *Let $\mathcal{F} \subseteq \mathcal{G}_n$ be the set of graphs G on the vertex set $\{1, \dots, n\}$ for which $\mathcal{R}^*\sigma$ assigns vertices of $V_{2,3}(G)$ distinct colours. Then there is a polynomial-time computable canonical labelling scheme for $\mathcal{F}$.*

2.2. **Basic facts about colour refinement.** We collect some simple observations about colour refinement. First, we observe that stable colourings of G are always *equitable*[16]: G is regular between each pair of colour classes.

Definition 2.15. For a graph G, a colouring $c : V(G) \to \Omega$ is *equitable* if for any two distinct colour classes A, B of G, the induced subgraph $G[A]$ is regular, and the induced bipartite subgraph $G[A, B]$ is a biregular bipartite graph.

Fact 2.16. *Consider any graph G and any colouring $c : V(G) \to \Omega$ then $\mathcal{R}^*c$ is equitable.*

In fact, $\mathcal{R}^*c$ is the coarsest *equitable partition refining c, in the sense that any equitable colouring c' refining c is also a refinement of $\mathcal{R}^*c$.*

The uniqueness of $\mathcal{R}^*c$ as the coarsest equitable partition refining c (in Fact 2.16) has the following consequence for the "consistency" of the colour refinement algorithm: if we start the colour refinement algorithm from any coarsening of $\mathcal{R}^*\sigma$, it will still end up at $\mathcal{R}^*\sigma$.

Fact 2.17. *Consider any graph G, and let $c : V(G) \to \Omega$ be a colouring which is a coarsening of $\mathcal{R}^*\sigma$. Then, $\mathcal{R}^*\sigma$ and $\mathcal{R}^*c$ define the same partition of $V(G)$.*

[16] This should not be confused with a different meaning of the term "equitable colouring" more common in extremal graph theory (namely, that the colouring has a roughly equal number of vertices in each colour).

Finally, we observe that the colour refinement algorithm can "see" which pairs of colours in a graph G are connected by a path. This is also true for the disparity graph, and for a more general class of graphs in which we can prescribe for each pair of colour classes whether to complement the edges between those colour classes.

Definition 2.18. For a graph G, a vertex-colouring $V(G) \to \Omega$, and a matrix $L \in \{0,1\}^{\Omega\times\Omega}$, let $M_L(G,c)$ be the graph defined as follows. For any (possibly non-distinct) pair of colours $\omega, \omega' \in \Omega$:

- If $L_{\omega,\omega'} = 1$, then $M_L(G,c)$ contains every possible edge between vertices of colours ω and ω'.
- If $L_{\omega,\omega'} = 0$, then $M_L(G,c)$ contains no edges between vertices of colours ω and ω'.

Then, define the *generalised disparity graph* $D_L(G,c) = M_L(G,c) \triangle G$.

Fact 2.19. *Fix a graph G, let $c = \mathcal{R}^*\sigma$, and let Ω be the set of colours used by c. Consider any $L \in \{0,1\}^{\Omega\times\Omega}$, any two vertices u, v for which $c(u) = c(v)$, and any colour $\omega \in \Omega$. Then, with respect to the graph $D_L(G,c)$, there is a path from u to an ω-coloured vertex if and only if there is a path from v to an ω-coloured vertex.*

Proof sketch. The key point is that c induces an equitable partition of $D_L(G,c)$ (indeed, c is an equitable partition of G by Fact 2.16, and complementing edges between pairs of colour classes does not change this property). So, if there is a path from u to a colour class C_ω via some sequence of colour classes $C_1, \dots, C_\ell$, then by biregularity, through the same sequence of colour classes it is also possible to find a path from v to C_ω. □

We remark that the main reason we introduced the general notion of L-disparity graphs is that if we have two colourings c, c' of a graph G, such that c' is a refinement of c, then the disparity graph $D(G,c)$ with respect to c can be interpreted as a generalised disparity graph $D_L(G,c')$ with respect to c'.

2.3. **The 2-dimensional Weisfeiler–Leman algorithm.** In this section we introduce a variant of colour refinement called the *2-dimensional Weisfeiler–Leman algorithm*, which is a refinement algorithm for colourings of pairs of vertices, instead of colourings of single vertices[17].

Definition 2.20. For a graph G and a colouring $f : V(G)^2 \to \Omega$ of the ordered pairs of vertices of G, the *2-dimensional refinement* $(2\mathcal{R})f : V(G)^2 \to \Omega \times \mathbb{N}^\Omega$ of f is defined by $(2\mathcal{R})f(u,v) = (f(u,v), (d_{\omega_1,\omega_2}(u,v))_{(\omega_1,\omega_2)\in\Omega^2})$, where $d_{\omega_1,\omega_2}(u,v)$ denotes the number of vertices $w \in V(G)$ such that $(f(u,w), f(w,v)) = (\omega_1, \omega_2)$. As in Definition 2.2, let $(2\mathcal{R})^*f$ be the "stable colouring", obtained by iterating the refinement operation $2\mathcal{R}$ until we see the same vertex-pair partition twice in a row.

For a vertex colouring $c : V(G) \to \Omega$, we write $\phi_{G,c} : V(G)^2 \to \Omega \sqcup \{0,1\}$ for the colouring defined by $\phi_{G,c}(u,v) = 1$ if uv is an edge, $\phi_{G,c}(u,v) = c(v)$ if $u = v$, and $\phi_G(u,v) = 0$ otherwise (here we write $\sqcup$ for disjoint union). We write ϕ_G as shorthand for ϕ_{G,σ_G}, recalling that σ_G is the trivial vertex-colouring of $V(G)$ assigning every vertex the same colour.

Also, for a colouring $f : V(G)^2 \to \Omega$, define the "vertex-projection" $\Pi f : V(G) \to \Omega$ by $\Pi f(v) = f(v,v)$. Let $\mathcal{V}^* = \Pi(2\mathcal{R})^*$ be the vertex-projection of the 2-dimensional Weisfeiler–Leman algorithm.

Remark 2.21. There are (at least) two slightly different definitions of the 2-dimensional Weisfeiler–Leman algorithm, with different properties. The definition above was the original one introduced by Cai, Fürer and Immerman [23], and is sometimes called the "2-folklore-Weisfeiler–Leman" algorithm (see [66]).

The first crucial fact we need about the 2-dimensional Weisfeiler–Leman algorithm is that it is efficiently computable. This was first observed by Cai, Fürer and Immerman [23].

Fact 2.22. *For any graph G, the stable vertex-pair-colouring $(2\mathcal{R})^*\phi_G$ can be computed in polynomial time. Moreover, this can be done "canonically" in the sense that different relabellings of the same graph always yield the same colouring $(2\mathcal{R})^*\phi_G$.*

Specifically, Cai, Fürer and Immerman [23] proved that the stable vertex-pair-colouring $(2\mathcal{R})^*\phi_G$ can be computed in time $O(n^3 \log n)$, but we will need an additional $O(\log n)$ "overhead" in order to make sure that all choices are made canonically (as discussed before Theorem 2.3, to prevent the sizes of the labels becoming too large we need to re-label the colours at each step; if we do not wish to make "arbitrary choices" then for this re-labelling we need a lexicographic sorting step). In the setting of the colour

[17] As the reader may have guessed, the colour refinement algorithm (also known as the 1-dimensional Weisfeiler–Leman algorithm) and the 2-dimensional Weisfeiler–Leman algorithm are both special cases of a more general *k-dimensional Weisfeiler–Leman algorithm*, first defined in an influential paper of Cai, Fürer and Immerman [23]. This will not be relevant for the present paper.

refinement algorithm, Bonsma, Berkholz and Grohe showed how to remove this "$O(\log n)$ overhead"; this may also be possible for the 2-dimensional Weisfeiler–Leman algorithm, but we were not able to find this in the literature.

The next crucial fact we need about the 2-dimensional Weisfeiler–Leman algorithm (really, the entire reason we need to consider this algorithm, instead of the simpler colour refinement algorithm) is that it can "detect distances between vertices", generalising Fact 2.19. Specifically, $\mathcal{V}^*\phi_G(u) = (2\mathcal{R})^*\phi_G(u,u)$ tells us the number of vertices of each colour at every given distance from u. This can be easily proved by induction (see also [25, Theorem 2.6.7] for a proof in a somewhat more general framework). We will need a version of this fact for the generalised disparity graphs defined in Definition 2.18, which can be proved in the same way.

Fact 2.23. *Fix a vertex-colouring c of a graph G, and let Ω be the set of colours used by $\mathcal{V}^*c$. Consider any $L \in \{0,1\}^{\Omega\times\Omega}$, any two vertices u,v for which $\mathcal{V}^*c(u) = \mathcal{V}^*c(v)$, any colour $\omega \in \Omega$ and any $i \in \mathbb{N}\cup\{\infty\}$. Then, with respect to the graph $D_L(G,\mathcal{V}^*c)$, the number of ω-coloured vertices at distance i from u is exactly the same as the number of ω-coloured vertices at distance i from v.*

A special case of Fact 2.23 (where L is the all-0 matrix, and $i = 1$) is that $\mathcal{V}^*\phi_{G,c}(u)$ is always an equitable partition. Since $\mathcal{V}^*\phi_{G,c}(u)$ is a refinement of c, it is a refinement of the stable colouring $\mathcal{R}^*c$ produced by the colour refinement algorithm, by Fact 2.16. (Informally, the 2-dimensional Weisfeiler–Leman algorithm is "at least as powerful" as the colour refinement algorithm).

Fact 2.24. *For any vertex-colouring c of a graph G, the vertex-colouring $\mathcal{V}^*\phi_{G,c}$ is a refinement of $\mathcal{R}^*c$.*

Finally, we need an analogue of Fact 2.17, on the "consistency" of the 2-dimensional Weisfeiler–Leman algorithm: if we start the 2-dimensional Weisfeiler–Leman algorithm from any coarsening of $(2\mathcal{R})^*f$, we will end up with a coarsening of $(2\mathcal{R})^*f$. This is a direct consequence of, e.g., the discussion in [25, Section 2.6.1] (see also [37, Proposition 2.1]), which characterises the vertex-pair-partition defined by $(2\mathcal{R})^*f$ as the unique coarsest coherent configuration refining f (here "coherent configurations" are the 2-dimensional analogues of the equitable partitions defined in Definition 2.15).

Fact 2.25. *Fix any graph G and any vertex-pair colouring $f : V(G)^2 \to \Omega$. If $g : V(G)^2 \to \Omega$ is a coarsening of $(2\mathcal{R})^*f$, then $(2\mathcal{R})^*g$ is a coarsening of $(2\mathcal{R})^*f$.*

2.4. **Probabilistic estimates.** At several points in the paper we will need some general-purpose probabilistic estimates. First, we need a Chernoff bound for sums of independent random variables; the following can be deduced from [47, Theorem 2.1].

Lemma 2.26 (Chernoff bound)**.** *Let X be a sum of independent random variables, each of which take values in $\{0,1\}$. Then for any $\delta > 0$ we have*

$$\mathbb{P}[X \le (1-\delta)\mathbb{E}X] \le \exp(-\delta^2\mathbb{E}X/2), \qquad \mathbb{P}[X \ge (1+\delta)\mathbb{E}X] \le \exp(-\delta^2\mathbb{E}X/(2+\delta)).$$

We also need a Littlewood–Offord type *anticoncentration estimate*, showing that certain sums of independent random variables are unlikely to take any particular value. Specifically, we need a generalisation of the classical Erdős–Littlewood–Offord theorem to the "p-biased" setting, essentially due to Juškevičius and Kurauskas [50] and Singhal [72].

Definition 2.27. Let $M(n,p)$ be the modal probability of a Binomial(n,p) random variable, i.e.,

$$M(n,p) = \binom{n}{x}p^x(1-p)^{n-x},$$

for $x = \lfloor (n+1)p \rfloor$. (It is well-known—see for example [51]—that $\lfloor (n+1)p \rfloor$ is always a mode for Binomial(n,p)).

Theorem 2.28. *Fix a positive integer n and a probability $p \in (0,1/2)$. Let $a_1,\dots,a_n \in \mathbb{R}\setminus\{0\}$ be nonzero real numbers, and let $\xi_1,\dots,\xi_n \in \{0,1\}^n$ be independent* Bernoulli(p) *random variables (meaning $\mathbb{P}[\xi_i = 1] = p$ and $\mathbb{P}[\xi_i = 0] = 1-p$). Then*

$$\sup_{x\in\mathbb{R}} \mathbb{P}[a_1\xi_1 + \dots + a_n\xi_n = x] \le M(\lfloor n/2 \rfloor, p) = O\left(\frac{1}{\sqrt{np}}\right).$$

The estimate $O(1/\sqrt{np})$ was first proved by Costello and Vu [27, Lemma 8.2]. The sharper bound $M(\lfloor n/2 \rfloor, p)$ can be deduced from a recent result due to Juškevičius and Kurauskas [50] and independently Singhal [72], as follows.

Proof of Theorem 2.28. First note that we may assume n is even (otherwise, condition on any outcome of ξ_{n-1} to reduce to the same question about $a_1\xi_1+\dots+a_{n-1}\xi_{n-1}$, noting in this case that $\lfloor (n-1)/2\rfloor = \lfloor n/2\rfloor$).

The main result of [50,72] is that (given our assumption that n is even), the quantity $\sup_{x\in\mathbb{R}}\mathbb{P}[a_1\xi_1+\dots+a_n\xi_n=x]$ is maximised when all of the a_i are equal to 1 or -1. By symmetry, we can assume $a_1,\dots,a_{\lfloor n/2\rfloor}=1$. Conditioning on arbitrary outcomes of $a_{\lfloor n/2\rfloor+1},\dots,a_n$, we see that the probability in question is bounded by

$$\sup_{x\in\mathbb{R}}\mathbb{P}\big[a_1\xi_1+\dots+a_{\lfloor n/2\rfloor}\xi_{\lfloor n/2\rfloor}=x\big]=M(\lfloor n/2\rfloor,p).$$

(here we used that $a_1,\dots,a_{\lfloor n/2\rfloor}=1$ to see that $a_1\xi_1+\dots+a_{\lfloor n/2\rfloor}\xi_{\lfloor n/2\rfloor}\sim\mathrm{Binomial}(\lfloor n/2\rfloor,p)$). □

3. Encoding colour refinement via views

There are many (quite different) lenses from which to view colour refinement. In particular, an important way to understand the performance of the colour refinement algorithm (around a vertex v in a graph G) is via a tree $\mathcal{T}_G(v)$ called the *view* (first defined by Yamashita and Kameda [81]).

Definition 3.1. Let G be a connected graph and $v\in V(G)$. Let $\mathcal{T}_G(v)$ be the (infinite, unless v is an isolated vertex) rooted tree defined as follows.

- The vertex set of $\mathcal{T}_G(v)$ is the set of all *walks* $W=(x_0,\dots,x_n)$ starting at $x_0=v$ (i.e., we require that x_ix_{i+1} is an edge of G for all $i\in\{0,\dots,n-1\}$).
- The root vertex of $\mathcal{T}_G(v)$ is the trivial walk $W=(v)$.
- We say that a walk W is a child of another walk W' (and put an edge between W and W') if W' is a one-step extension of W (i.e., if we can write $W=(x_0,\dots,x_n)$ and $W'=(x_0,\dots,x_n,x_{n+1})$).

Note that the set of vertices at depth ℓ in $\mathcal{T}_G(v)$ correspond precisely to the set of walks of length ℓ starting from v in G. For $i\ge 0$, we write $\mathcal{T}^i_G(v)$ for the subtree of $\mathcal{T}_G(v)$ restricted to walks of length at most i. See Figure 2 for an example.

We note that it will sometimes be convenient to view each vertex $u=(x_0,\dots,x_n)$ of $\mathcal{T}_G(v)$ as a "copy" of the vertex x_n.

The following lemma provides a connection between views and colour refinement. It was first observed by Angluin [2] in a slightly different context (the particular statement here follows directly from [55, Lemma 2.6]).

Lemma 3.2. *Let G be a connected graph, and u,v two vertices. Then we have the rooted tree isomorphism $\mathcal{T}^i_G(u)\cong\mathcal{T}^i_G(v)$ if and only if $\mathcal{R}^i\sigma(u)=\mathcal{R}^i\sigma(v)$ (i.e., if u and v have the same colour after i steps of colour refinement starting from the trivial colouring σ). In particular, $\mathcal{T}_G(u)\cong\mathcal{T}_G(v)$ if and only if u and v have the same colour in the stable colouring $\mathcal{R}^*\sigma$.*

Remark 3.3. [55, Lemma 2.6] is stated for *universal covers*, which have the same definition as views (as in Definition 3.1) except that one only considers *non-backtracking* walks $W=(x_0,\dots,x_n)$ with $x_{i+1}\neq x_{i-1}$ for all $i\in\{1,\dots,n-1\}$. As discussed in [55], it is easy to see that views and universal covers encode the same information.

For our purposes, it will not be very convenient to work with views directly. Instead, we consider sequences of multisets $\mathcal{L}^i_G(u,v)$ which describe the difference between two views $\mathcal{T}^i_G(u)$ and $\mathcal{T}^i_G(v)$ at depth i.

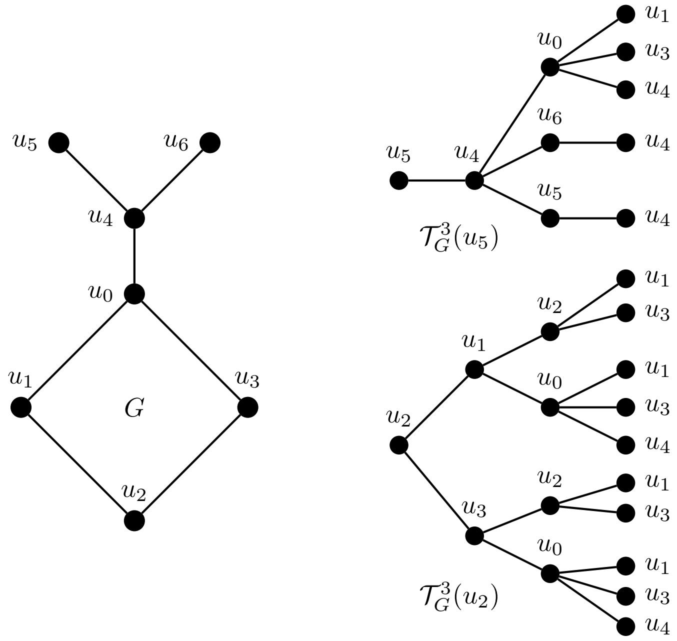
Definition 3.4. Let G be a graph. For distinct $u,v\in V(G)$ and $i\ge 0$, we recursively define multisets $\mathcal{L}^i_G(u,v)$, as follows.

- For all u,v, let $\mathcal{L}^0_G(u,v):=\{u\}$.
- For $u,v,w\in V(G)$ and $i\ge 1$, let $\ell^i_G(w,u,v)$ be the number of neighbours of w in $\mathcal{L}^{i-1}_G(u,v)$.
- For $u,v\in V(G)$ and $i\ge 1$, we define $\mathcal{L}^i_G(u,v)$ as follows. For each $w\in V(G)$, if $\ell^i_G(w,u,v)>\ell^i_G(w,v,u)$, then put $\ell^i_G(w,u,v)-\ell^i_G(w,v,u)$ copies of w in $\mathcal{L}^i_G(u,v)$.

Then, write $\operatorname{supp}\mathcal{L}^i_G(u,v)$ for the support of $\mathcal{L}^i_G(u,v)$, that is, the set of elements appearing in $\mathcal{L}^i_G(u,v)$, and write

$$\mathcal{S}^{\le i}_G(\{u,v\}):=\bigcup_{j\le i}\Big(\operatorname{supp}\mathcal{L}^j_G(u,v)\cup\operatorname{supp}\mathcal{L}^j_G(v,u)\Big),\quad \mathcal{S}^i_G(\{u,v\}):=\mathcal{S}^{\le i}_G(\{u,v\})\setminus\mathcal{S}^{\le i-1}_G(\{u,v\}).$$

Note that $\operatorname{supp}\mathcal{L}^i_G(u,v)\cup\operatorname{supp}\mathcal{L}^i_G(v,u)$ is a *disjoint* union: by definition, there are no vertices with positive multiplicity in both $\mathcal{L}^i_G(u,v)$ and $\mathcal{L}^i_G(v,u)$. We think of $\mathcal{S}^i_G(\{u,v\})$ as being the set of vertices which "appear differently" in the views $\mathcal{T}_G(u)$ and $\mathcal{T}_G(v)$ for the first time at depth i.



$\mathcal{L}^0(u_5, u_2) = \{u_5\}$
$\mathcal{L}^1(u_5, u_2) = \{u_4\}$
$\mathcal{L}^2(u_5, u_2) = \{u_5, u_6\}$
$\mathcal{L}^3(u_5, u_2) = \{u_4\}$

$\mathcal{L}^0(u_2, u_5) = \{u_2\}$
$\mathcal{L}^1(u_2, u_5) = \{u_1, u_3\}$
$\mathcal{L}^2(u_2, u_5) = \{u_0, u_2, u_2\}$
$\mathcal{L}^3(u_2, u_5) = \{u_1, u_1, u_1, u_3, u_3, u_3\}$

$\mathcal{S}^0(\{u_2, u_5\}) = \{u_2, u_5\}$
$\mathcal{S}^1(\{u_2, u_5\}) = \{u_1, u_3, u_4\}$
$\mathcal{S}^2(\{u_2, u_5\}) = \{u_0, u_6\}$
$\mathcal{S}^3(\{u_2, u_5\}) = \emptyset$

Figure 2. An example graph G, and the depth-3 views for u_5 and u_2. As these trees are non-isomorphic, one can deduce that u_2 and u_5 are assigned different colours after three steps of the colour refinement algorithm (in fact, they already receive different colours after the first step). In the figure we show the multisets $\mathcal{L}^i(u_2, u_5)$, $\mathcal{L}^i(u_5, u_2)$ and the sets $\mathcal{S}^i(\{u_2, u_5\})$ for $i \in \{0, 1, 2, 3\}$.

See Figure 2 for an example. For the notation introduced in Definitions 3.1 and 3.4, we will often omit the subscript "G" when it is clear from context.

Lemma 3.2 has the following consequence in terms of our multisets $\mathcal{L}^i(u, v)$.

Lemma 3.5. *Let G be a graph and consider distinct $u, v \in V(G)$. If there exist positive integers i and d such that the number of copies of vertices of degree d in $\mathcal{L}^i(u, v)$ and $\mathcal{L}^i(v, u)$ are not equal then $\mathcal{R}^*\sigma(u) \neq \mathcal{R}^*\sigma(v)$.*

Lemma 3.5 is in some sense self-evident (the multisets $\mathcal{L}^i(u, v)$ describe the discrepancy between the ith layers of $\mathcal{T}(u)$ and $\mathcal{T}(v)$), but for completeness we provide a formal proof.

Proof of Lemma 3.5. Recalling that the vertices of each $\mathcal{T}(v)$ can be viewed as copies of vertices of G, for each d, i, u, v let $\mathcal{M}^i_d(v)$ be the multiset of vertices of degree d appearing on the i-th layer of $\mathcal{T}(v)$, and let $\mathcal{L}^i_d(u, v)$ be the multiset of vertices of degree d in $\mathcal{L}^i(u, v)$.

We will prove the contrapositive of the lemma statement. Fix u, v and suppose that $\mathcal{R}^*\sigma(u) = \mathcal{R}^*\sigma(v)$; our objective is to prove that $|\mathcal{L}^i_d(u, v)| = |\mathcal{L}^i_d(v, u)|$ for all i, d. To this end, we will prove (by induction on i) that for all d there is a multiset $W^i_d(\{u, v\})$ (of degree-d vertices common to the i-th layers of $\mathcal{T}(u), \mathcal{T}(u)$) such that $\mathcal{M}^i_d(u) = \mathcal{L}^i_d(u, v) \cup W^i_d(\{u, v\})$ and $\mathcal{M}^i_d(v) = \mathcal{L}^i_d(v, u) \cup W^i_d(\{u, v\})$ (these are multiset unions, respecting multiplicity). By Lemma 3.2, we have $|\mathcal{M}^i_d(u)| = |\mathcal{M}^i_d(v)|$, so this suffices.

As the $(i = 0)$ base case for the induction, note that $\mathcal{R}^*\sigma(u) = \mathcal{R}^*\sigma(v)$ implies that u, v have the same degree, thus the statement holds with $W^i_d(\{u, v\}) = \emptyset$. So, consider $i \geq 1$, and assume that for all d there exists a set $W^{i-1}_d(\{u, v\})$ such that

$$\mathcal{M}^{i-1}_d(u) = W^{i-1}_d(\{u, v\}) \cup \mathcal{L}^{i-1}_d(u, v) \text{ and } \mathcal{M}^{i-1}_d(v) = W^{i-1}_d(\{u, v\}) \cup \mathcal{L}^{i-1}_d(v, u) \text{ for } d \geq 1.$$

Now, define multisets of vertices $X^i_d(\{u, v\})$ and $Y^i_d(\{u, v\})$ as follows. For $w \in V(G)$, of degree $d(w) = d$, the number of copies of w in $X^i_d(\{u, v\})$ is precisely $\min\{\ell^i_G(w, u, v), \ell^i_G(w, v, u)\}$, and the number of copies of w in $Y^i_d(\{u, v\})$ is the number of neighbours of w (counted with multiplicity) in the multiset union $\bigcup_{d \geq 1} W^{i-1}_d(\{u, v\})$. Recalling the definitions of $\mathcal{L}^i(u, v)$ and $\mathcal{L}^i(u, v)$ (as the "surplus" of vertices in the i-th layer of $\mathcal{T}(u)$ compared to $\mathcal{T}(v)$, and vice versa), and recalling the inductive assumption, we have $\mathcal{M}^i_d(v) = \mathcal{L}^i_d(v, u) \cup X^i_d(\{u, v\}) \cup Y^i_d(\{u, v\})$ and $\mathcal{M}^i_d(u) = \mathcal{L}^i_d(u, v) \cup X^i_d(\{u, v\}) \cup Y^i_d(\{u, v\})$. So we can take $W^i_d(\{u, v\}) = X^i_d(\{u, v\}) \cup Y^i_d(\{u, v\})$, for the desired conclusion. □

4. Expansion for views

Recall the notation $\mathcal{S}^i(\{u,v\})$ defined in Definition 3.4, describing the set of vertices which appear differently in the views $\mathcal{T}(u)$ and $\mathcal{T}(v)$ for the first time at depth i.

In this section we prove that, for two vertices u, v in an appropriately randomly perturbed graph G, it is very likely that whenever $\mathcal{S}^{\le i}(\{u,v\})$ grows reasonably large, u and v end up with different colours in the stable colouring $\mathcal{R}^*\sigma$. This is formalised in the following proposition.

Proposition 4.1. *Let G_0 be a graph on the vertex set $\{1,\dots,n\}$ and let $G_{\mathrm{rand}} \sim \mathbb{G}(n,p)$, for some $p \in [0,1/2]$ satisfying*

$$\frac{1+(\log n)^{-40}}{n} \le p \le \frac{100\log n}{n}.$$

Then whp $G = G_0 \triangle G_{\mathrm{rand}}$ satisfies the following property. For every pair of vertices u, v for which there is some i satisfying

$$|\mathcal{S}_G^{\le i}(\{u,v\})| \ge \frac{100\log n}{\min\{(np-1)^2,1\}}, \tag{4.1}$$

we have $\mathcal{R}_G^*\sigma(u) \ne \mathcal{R}_G^*\sigma(v)$.

We remark that, while we did not necessarily make an effort to optimise the assumption on p, there is some significance to the “$1/n$” essentially appearing in Proposition 4.1. Indeed, this corresponds to the celebrated *phase transition* for Erdős–Rényi random graphs (as briefly discussed in Section 1.3.4).

Morally speaking, the main reason why Proposition 4.1 is true is that random graphs are extremely good *expanders*: starting from any reasonably large set of vertices S, the set of vertices within distance i of S grows geometrically in i until it spans nearly the whole graph. The following lemma (a key ingredient in our proof of Proposition 4.1) is a counterpart to this fact for the sets $\mathcal{S}^i(\{u,v\})$ in a randomly perturbed graph: whenever $\mathcal{S}^{\le i}(\{u,v\})$ is reasonably large, there is some j for which $\mathcal{S}^j(\{u,v\})$ is huge, spanning a significant proportion of the whole graph.

Lemma 4.2. *Let $G_0, p, G_{\mathrm{rand}}$ be as in Proposition 4.1. Then whp $G = G_0 \triangle G_{\mathrm{rand}}$ satisfies the following property. For every pair of vertices u, v for which there is some i satisfying (4.1), there is some j satisfying*

$$\left|\mathcal{S}_G^j(\{u,v\})\right| \ge \frac{n}{4(\log n)^{140}}. \tag{4.2}$$

We defer the proof of Lemma 4.2 to the end of the section. Given Lemma 4.2, the strategy for the proof of Proposition 4.1 is as follows. For each u, v satisfying (4.1), and j as guaranteed by Lemma 4.2, we first observe that it is possible to reveal the set $\mathcal{S}_G^j(\{u,v\})$ without revealing too much information about the random perturbation G_{rand} inside this set. We wish to use the remaining randomness, together with a Littlewood–Offord-type anticoncentration inequality (Theorem 2.28), to show that it is very unlikely that the multisets $\mathcal{L}_G^j(u,v)$ and $\mathcal{L}_G^j(v,u)$ have the same degree statistics.

This final step is actually rather delicate. Indeed, if we let $Q_d^i(u,v)$ be the number of vertices of degree d in $\mathcal{L}^i(u,v)$, minus the corresponding number in $\mathcal{L}^i(v,u)$, it is not too hard to see that $\sum_d dQ_d^i(u,v)$ (which is the number of edges incident to $\mathcal{L}^i(u,v)$ plus the number of edges spanned by $\mathcal{L}^i(u,v)$) can be expressed as a weighted sum of Bernoulli random variables (as required for Theorem 2.28). So, we can use Theorem 2.28 to show that usually we have $\sum_d dQ_d^i(u,v) \ne 0$ (in which case some $Q_d^i(u,v) \ne 0$, so the degree statistics of $\mathcal{L}^i(u,v)$ and $\mathcal{L}^i(v,u)$ must be different). However, this estimate is not good enough for a union bound over choices of u, v, and it is necessary to understand in much more detail the *joint* anticoncentration of $dQ_d^i(u,v)$, between different d. This is accomplished by first revealing almost all information about G_{rand} except the edges inside a tiny subset of $\mathcal{S}_G^j(\{u,v\})$ (which we call a “hole”). In the resulting conditional probability space, we obtain fairly weak anticoncentration bounds on the $dQ_d^i(u,v)$, but we can show that $dQ_d^i(u,v)$ is essentially independent from $d'Q_{d'}^i(u,v)$ whenever d and d' are reasonably far apart (so we can multiply weak anticoncentration bounds together for a much stronger final bound).

The details of the proof of Proposition 4.1 are as follows.

Proof of Proposition 4.1. Fix vertices u, v. Our objective is to prove that with probability $1 - o(n^{-2})$, there is no i satisfying (4.1), or the property in Lemma 4.2 fails to hold, or there are some j, d such that the number of copies of vertices of degree d in $\mathcal{L}^j(u,v)$ and $\mathcal{L}^j(v,u)$ are not equal. The desired result will then immediately follow by a union bound over choices of u, v, and Lemmas 3.5 and 4.2.

Step 1: Exploration. We reveal edges of G_{rand} according to the following procedure. For each j (starting at $j = 0$ and increasing j by one at each step), if (4.2) does not yet hold, then reveal all edges of G_{rand} incident to $\mathcal{S}^j(\{u, v\})$. The procedure terminates at the first step j for which either (4.2) holds or $\mathcal{S}^j(\{u, v\}) = \emptyset$. We fix this j for the rest of the proof; from now on, we will treat $j, \mathcal{S}^j(\{u, v\}), \mathcal{S}^{\leq j-1}(\{u, v\})$ as non-random objects (they are determined by the aforementioned revelation process). Note that we have not yet revealed anything about the edges of G_{rand} inside $\mathcal{S}^j(\{u, v\})$.

If there is i satisfying (4.1), and if the property in Lemma 4.2 holds, then the latter of the two termination conditions cannot happen: we can assume that (4.2) holds.

Step 2: Setting up a small "hole". Recall from Definition 3.4 that $\mathcal{S}^j(\{u, v\})$ is defined in terms of the disjoint supports $\operatorname{supp}\mathcal{L}^j(u, v)$ and $\operatorname{supp}\mathcal{L}^j(v, u)$. Let $A = \mathcal{S}^j(\{u, v\}) \cap \operatorname{supp}\mathcal{L}^j(u, v)$, and assume without loss of generality A is larger than $\mathcal{S}^j(\{u, v\}) \cap \operatorname{supp}\mathcal{L}^j(v, u)$, so by (4.2) we have

$$|A| \geq \frac{n}{8(\log n)^{140}}. \tag{4.3}$$

Let S be a subset of $n^{0.75}$ vertices of A. Let $G_1 = G_0 \triangle (G_{\text{rand}} - G_{\text{rand}}[S])$ be the graph obtained by randomly perturbing G_0 via all the edges of G_{rand} except those inside S.

The idea is that, since S is so small, the degrees of vertices in S with respect to G_1 are a very good prediction of the degrees with respect to G. So, if we reveal G_1 (leaving only the randomness inside S), we might hope to be able to find disjoint "buckets" of vertices in S, whose degrees with respect to G_1 are so far apart from each other that it is highly unlikely that two distinct buckets will have vertices with the same degree in G. This means we can handle each bucket essentially independently, which will yield strong joint anticoncentration bounds.

Step 3: Specifying degree buckets. The following claim will be used to construct these buckets (essentially, it shows that the degree distribution of the vertices of S with respect to G_1 is "spread out"). Let $V(d)$ be the set of vertices in S which have degree d with respect to G_1, and for a set $D \subseteq \mathbb{N}$, let $V(D) = \bigcup_{d \in D} V(d)$. Also let $n(d) = |V(d)|$ and $n(D) = |V(D)|$.

Claim 4.3. *With probability $1 - o(n^{-2})$ (over the remaining randomness of G_1; i.e., conditioned on the edges that have already been revealed), for every set $D \subseteq \{0, \dots, n\}$ of size 100^2 we have $|S| - n(D) > n^{0.7}$ (i.e., there are many vertices whose degrees do not lie in D).*

Proof of claim. Fix a set $D \subseteq \{0, \dots, n\}$ of size 100^2. We will show that the desired property holds for this D with probability at least $1 - o(n^{-100^2-2})$, so the desired result will follow from the union bound.

Let T be a subset of $n/(10(\log n)^{140})$ vertices of A disjoint from S (such a T exists by (4.3)).

First, reveal all edges of G_1 except the edges between S and T. In the resulting conditional probability space, the degree of each vertex $x \in S$ (with respect to G_1) only depends on the edges between x and T in G_{rand} (and thus these degrees are independent): we can write

$$\deg_{G_1}(x) = b_x + c_x + E_x^+ - E_x^-,$$

where

- b_x is the number of neighbours of x outside T with respect to G_1,
- c_x is the number of neighbours of x inside T with respect to G_0,
- E_x^+ is the number of neighbours of x inside T with respect to G_{rand}, among non-neighbours of x with respect to G_0.
- E_x^- is the number of neighbours of x inside T with respect to G_{rand}, among neighbours of x with respect to G_0.

Note that $E_x^+ \sim \text{Binomial}(e_x^+, p)$ and $E_x^- \sim \text{Binomial}(e_x^-, p)$, where e_x^+ (respectively e_x^-) is the number of non-neighbours (respectively, neighbours) of x inside T with respect to G_0.

Now, for each $x \in S$, we want to show that it is reasonably likely that $\deg_{G_1}(x) \notin D$. Fix some x, and assume without loss of generality that $e_x^+ \geq e_x^-$ (so $e_x^+ = n^{1-o(1)}$). Since $|D| = 100^2$, there is some $d_x \notin D$ such that $0 \leq d_x - (b_x + c_x) \leq 100^2$, meaning that one way to have $\deg_{G_1}(x) \notin D$ is to have $E_x^- = 0$ and $E_x^+ = d_x - (b_x + c_x)$. We can compute the probability that this occurs (in our conditional probability space, given information revealed so far).

$$\begin{aligned}\mathbb{P}[\deg_{G_1}(x) \notin D] &\geq \mathbb{P}[\deg_{G_1}(x) = d_x] \geq \mathbb{P}[E_x^- = 0 \text{ and } E_x^+ = d_x - (b_x + c_x)]\\ &\geq \min_{0 \leq k \leq 100^2} (1-p)^{e_x^-} \binom{e_x^+}{k} (1-p)^{e_x^+ - k} p^k = n^{-o(1)} \geq n^{-0.01},\end{aligned}$$

recalling that $p = n^{o(1)}/n$ and $e_x^+ = n^{1-o(1)}$.

Now, $|S|-n(D)$ is the number of $x \in S$ for which $\deg_{G_1}(x) \notin D$. In our conditional probability space, recall that the events $\deg_{G_1}(x) \notin D$ are independent. We have $\mathbb{E}[|S|-n(D)] \geq |S|n^{-0.01} \geq n^{0.73}$, so by a Chernoff bound (Lemma 2.26) we have

$$\mathbb{P}[|S|-n(D) < n^{0.7}] \leq \exp(-\Omega(n^{0.73})) = o(n^{-100^2-2}),$$

as desired. ■

Now, we use Claim 4.3 to construct many large sets of vertices with very distinct degrees with respect to G_1, as follows.

Claim 4.4. *If n is sufficiently large and G_1 satisfies the conclusion of Claim 4.3, then there exists a sequence of pairwise disjoint sets $D_1, \dots, D_{33} \subseteq \mathbb{N}$ such that $n(D_t) \geq n^{0.6}$ for all t, and such that for each $s \neq t$ and each $d \in D_s, d' \in D_t$ we have $|d-d'| > 200$.*

Proof of claim. Let D_{big} be the set of integers d such that $n(d) \geq n^{0.6}$. We split into cases depending on the size of D_{big}.

Case 1: $|D_{\text{big}}| \geq 100^2$. In this case, we can simply order the elements of D_{big} as $d_1 \leq d_2 \leq \cdots \leq d_\ell$, and take $D_t = \{d_{201t}\}$ for each $t \in \{1, \dots, 33\}$.

Case 2: $|D_{\text{big}}| < 100^2$. In this case we greedily construct a sequence of nonnegative integers $a_1 \leq \cdots \leq a_{100}$, such that $n(\{a_r, a_r+1, a_r+2, \dots, a_{r+1}-1\}) \geq n^{0.6}$ and $a_{r+1} - a_r \geq 101$ for each $r \in \{1, \dots, 99\}$; this suffices, as we can then take $D_t = \{a_{3t}, a_{3t}+1, \dots, a_{3t+1}-1\}$ for each $t \in \{1, \dots, 33\}$.

Specifically, we let $a_1 = 0$, and for $r \geq 1$ we recursively define a_{r+1} to be the minimal integer larger than $a_r + 100$ such that $n(\{a_r, a_r+1, a_r+2, \dots, a_{r+1}-1\}) \geq n^{0.6}$. To see that such an a_{r+1} always exists, note that

$$n(\{0,1,2,\dots,a_r-1\}) \leq 2rn^{0.6} + 100(r-1)n^{0.6} + n(D_{\text{big}}) \leq 102rn^{0.6} + n(D_{\text{big}}),$$

so for $r \leq 100$ we have $n(\{a_r, a_{r+1} \dots, n\}) \geq n - n(D_{\text{big}}) - 102rn^{0.6} \geq n^{0.7} - 102 \cdot 100 \cdot n^{0.6} \geq n^{0.6}$ by the property in Claim 4.3, and an appropriate choice for a_{r+1} exists. ■

Now, reveal all the edges in G_{rand} except those inside S (i.e., we have now revealed G_1, so from now on we treat quantities of the form $n(D)$ as being non-random). Assume that the conclusion of Claim 4.4 holds, giving us a sequence of disjoint intervals $D_1, \dots, D_{33}$.

Step 4: Independence between the buckets. We want to be able to study each $V(D_t)$ separately, so we need to make sure that the degrees of vertices in the different $V(D_t)$ cannot coincide. The following claim will be used for this purpose. (Recall that S is our small "hole" that each of the D_t lie inside).

Claim 4.5. *With probability $1-o(n^{-2})$, the induced subgraph $G_{\text{rand}}[V(S)]$ has maximum degree at most 100.*

Proof of claim. The probability there is a vertex in $G_{\text{rand}}[V(S)]$ with at least 100 neighbours is at most

$$n^{0.75}\binom{n^{0.75}}{100}p^{100} = o(n^{-2}),$$

recalling that $p = n^{o(1)}/n$. ■

For each $t \in \{1, \dots, 33\}$, let $B(D_t, 100)$ be the set of all $b \in \mathbb{N}$ which differ from some element of D_t by at most 100. So, the property in Claim 4.5 implies that for each t and each vertex $w \in V(D_t)$, the degree of w with respect to G lies in $B(D_t, 100)$. By construction (of the D_t), the sets $B(D_t, 100)$ are disjoint from each other.

Now, we also reveal all the edges in G_{rand} except those inside the sets $V(D_1), \dots, V(D_{33})$. So, the only remaining randomness is inside these sets.

Step 5: Anticoncentration of degree statistics. Let Q_d be the number of vertices in $\mathcal{L}^j(u,v)$ which have degree d with respect to G, minus the corresponding number in $\mathcal{L}^j(v,u)$ (here, the vertices are counted with multiplicity). Given the information revealed so far, our objective is to prove that with probability $1-o(n^{-2})$, either the property in Claim 4.5 fails to hold, or there is some d such that Q_d is nonzero (or both).

For each $t \in \{1, \dots, 33\}$, let

$$Z_t = \sum_{d \in B(D_t, 100)} dQ_d.$$

Note that if $Z_t \neq 0$, then there is some d such that Q_d is nonzero. These random variables Z_t have been carefully defined in such a way that they can be expressed in the form required for Theorem 2.28.

Indeed, note that the information we have revealed so far already determines the degree (with respect to G) of every vertex outside the sets $V(D_1), \dots, V(D_{33})$, and recall that $\mathcal{L}^j(u,v)$ and $\mathcal{L}^j(v,u)$ are disjoint. For all t:

- Let b_t be the contribution to dQ_d from all vertices outside $V(D_1) \cup \cdots \cup V(D_{33})$ (i.e., b_t is a weighted sum of degrees, with respect to G_1, of vertices which do not lie in $V(D_1) \cup \cdots \cup V(D_{33})$).
- Let c_t be the sum of the degrees *with respect to* G_1 of all vertices in $V(D_t)$, multiplied by the number of times that vertex appears in $\mathcal{L}^j(u,v)$.
- For each pair of vertices $\{x,y\} \subseteq V(D_t)$, let $\xi_{\{x,y\}}$ be the indicator random variable for the event that xy is an edge in G_{rand}.
- For each $\{x,y\} \subseteq V(D_t)$, let $\chi(\{x,y\}) = -1$ if xy is an edge of G_0, and $\chi(\{x,y\}) = 1$ if xy is not an edge of G_0.
- For each $x \in V(D_t)$, let a_x be the number of times that x appears in $\mathcal{L}^j(u,v)$.

Define

$$Z_t^* = b_t + c_t + \sum_{\{x,y\} \subseteq V(D_t)} \chi(\{x,y\})(a_x + a_y)\xi_{\{x,y\}},$$

and note that whenever the property in Claim 4.5 holds, we have $Z_t = Z_t^*$.

Note that $\chi(\{x,y\})(a_x + a_y)$ is nonzero for any distinct $x, y \in V(D_t)$, and note that the different Z_t^* are independent from each other. Recalling Claim 4.4, we have $\binom{|V(D_t)|}{2} \geq \binom{n^{0.6}}{2} = \Omega(n^{1.2})$. Also recall that $p = n^{o(1)}/n$. So, by Theorem 2.28, we have

$$\mathbb{P}[Z_t^* = 0 \text{ for all } t] \leq O\left(\left(\frac{1}{\sqrt{pn^{1.2}}}\right)^{33}\right) = o(n^{-2}).$$

It follows that with probability $1 - o(n^{-2})$ there is some t such that $Z_t^* = Z_t \neq 0$, meaning that there is some d such that Q_d is nonzero, meaning that u and v receive different colours in the stable colouring $\mathcal{R}^*\sigma$. □

We have now completed the proof of Proposition 4.1 modulo Lemma 4.2, which we prove in the next subsection.

4.1. **Exploring views: proof of Lemma 4.2.** As mentioned earlier, roughly speaking Lemma 4.2 is proved using the expansion properties of G_{rand}. As a naïve approach to prove Lemma 4.2, it is not hard to prove that whenever $\mathcal{S}^i(\{u,v\})$ is reasonably large, then $\mathcal{S}^{i+1}(\{u,v\})$ tends to be even larger (by a factor of about np), and one can iterate this to show that $\mathcal{S}^j(\{u,v\})$ eventually reaches the desired size. However, assumption (4.1) in Lemma 4.2 does not guarantee that any *individual* $\mathcal{S}^i(\{u,v\})$ is large, so we must be more careful. We proceed via a delicate coupling argument comparing sets of the form $\mathcal{S}^{\leq q}(\{u,v\})$ and $\mathcal{S}^{\leq q+1}(\{u,v\})$.

Proof of Lemma 4.2. Recall that we are trying to prove that for all pairs of distinct vertices u, v, if (4.1) holds for some i then (4.2) holds for some j.

Step 1: Reducing to single-step expansion. Our main objective will be to prove that whp G satisfies the following property: for every u, v, q for which

$$\frac{100 \log n}{\min\{(np-1)^2, 1\}} \leq |\mathcal{S}^{\leq q}(\{u,v\})| \leq \frac{n}{(\log n)^{100}}, \tag{4.4}$$

we have

$$|\mathcal{S}^{\leq q+1}(\{u,v\})| \geq \left(1 + \frac{np-1}{2}\right)|\mathcal{S}^{\leq q}(\{u,v\})|, \tag{4.5}$$

or equivalently

$$|\mathcal{S}^{q+1}(\{u,v\})| \geq \frac{np-1}{2}|\mathcal{S}^{\leq q}(\{u,v\})|.$$

To see that this suffices to prove the lemma statement, consider u, v, i satisfying (4.1), and suppose that the above property holds. Let $j \geq i$ be minimal such that $|\mathcal{S}^{\leq j}(\{u,v\})| > n/(\log n)^{100}$. Taking $q = j-1$, we obtain

$$\frac{2}{np-1}|\mathcal{S}^j(\{u,v\})| \geq |\mathcal{S}^{\leq j-1}(\{u,v\})|,$$

$$\left(1+\frac{2}{np-1}\right)|\mathcal{S}^j(\{u,v\})| \geq |\mathcal{S}^{\leq j}(\{u,v\})| > \frac{n}{(\log n)^{100}},$$

so recalling that $np-1 \geq (\log n)^{-40}$, we have $1/(1+2/(np-1)) \geq 1/(4(\log n)^{40})$ and

$$\left|\mathcal{S}^j_G(\{u,v\})\right| \geq \frac{n}{4(\log n)^{140}},$$

meaning that j satisfies (4.2).

So, from now on we focus on showing that the above property holds whp. In fact, fix distinct vertices u,v; we will prove that with probability $1-o(n^{-2})$, each q satisfying (4.4) also satisfies (4.5). This suffices; the desired result will then follow from a union bound over u,v.

In order to prove this, we need to study how $\mathcal{S}^{\leq q+1}(\{u,v\})$ is defined in terms of random edges of G_{rand} incident to $\mathcal{S}^{\leq q}(\{u,v\})$. The idea, roughly speaking, is that we expect $\mathcal{S}^{\leq q+1}(\{u,v\})$ to have size at least about $(1+np)|\mathcal{S}^{\leq q}(\{u,v\})|$, so having size at most $(1+(np-1)/2)|\mathcal{S}^{\leq q}(\{u,v\})|$ is very unlikely.

Step 2: Exploration. As in the proof of Proposition 4.1 we consider the following procedure to iteratively reveal edges of G_{rand}. For each t (starting at $t=0$ and increasing t by one at each step), reveal all edges of G_{rand} incident to $\mathcal{S}^t(\{u,v\})$; this determines the set $\mathcal{S}^{t+1}(\{u,v\})$, but all the edges incident to $\mathcal{S}^{t+1}(\{u,v\})$ remain unrevealed.

We also define a sub-procedure to break down further the process of revealing edges of G_{rand} incident to $\mathcal{S}^t(\{u,v\})$. To define this sub-procedure we need some more notation.

- Let G^t_{rand} be the graph of edges of G_{rand} which have been revealed so far (i.e., G^t_{rand} is the subgraph of G_{rand} consisting of edges incident to $\mathcal{S}^{\leq t-1}(\{u,v\})$).
- Let $G^t = G_0 \triangle G^t_{\text{rand}}$.
- Let $W^t = \{1,\dots,n\} \setminus \mathcal{S}^{\leq t}(\{u,v\})$ be the set of vertices outside $\mathcal{S}^{\leq t}(\{u,v\})$ (i.e., the set of candidate vertices for $\mathcal{S}^{t+1}(\{u,v\})$).
- Recall the notation in Definition 3.4, and let Z^t be the set of vertices $w \in W^t$ such that $\ell^{t+1}_{G^t}(w,u,v) \neq \ell^{t+1}_{G^t}(w,v,u)$ (i.e., such that w has a different number of neighbours in $\mathcal{L}^t(u,v)$ than $\mathcal{L}^t(v,u)$, with respect to G^t).

Note that the vertices in Z^t are the vertices which (intuitively speaking) seem to be most likely to end up in $\mathcal{S}^{t+1}(\{u,v\})$. Specifically, we have the following observation.

Fact 4.6. *For every $z \in Z^t$, if G_{rand} has no edges between $\mathcal{S}^t(\{u,v\})$ and z, then $z \in \mathcal{S}^{t+1}(\{u,v\})$.*

The vertices in $W^t \setminus Z^t$ have an equal number of neighbours in $\mathcal{L}^t(u,v)$ and $\mathcal{L}^t(v,u)$ with respect to G^t, so they will not end up in $\mathcal{S}^{t+1}(\{u,v\})$ unless G_{rand} has certain edges between $\mathcal{S}^t(\{u,v\})$ and z. The simplest way this can happen is as follows.

Fact 4.7. *For every $w \in W^t \setminus Z^t$, if G_{rand} has exactly one edge between $\mathcal{S}^t(\{u,v\})$ and w, then $w \in \mathcal{S}^{t+1}(\{u,v\})$.*

With the above two facts in mind, our revelation sub-procedure is defined as follows.

- First, fix an arbitrary ordering of the vertices $z \in Z^t$. For each such z:
    - reveal all edges between z and $\mathcal{S}^t(\{u,v\})$, determining whether z is in $\mathcal{S}^{t+1}(\{u,v\})$. We say this is a *type-Z critical moment*, and if $z \in \mathcal{S}^{t+1}(\{u,v\})$ we say the critical moment *succeeds*.
- Then, fix an arbitrary ordering of the vertex pairs $s \in \mathcal{S}^t(\{u,v\})$ and $w \in W^t \setminus Z^t$. For each such s,w:
    - if we have not yet revealed whether sw is an edge of G_{rand}, then we say this is a *type-W critical moment* for sw, and reveal the status of sw. If sw is an edge of G_{rand} we say the critical moment *advances*; in this case, also reveal the rest of the edges between w and $\mathcal{S}^t(\{u,v\})$, determining whether w is in $\mathcal{S}^{t+1}(\{u,v\})$. If $w \in \mathcal{S}^{t+1}(\{u,v\})$, then we say the critical moment *succeeds*.

It is convenient to continue the entire revelation process only as long as $|\mathcal{S}^{\leq t}(\{u,v\})| \leq n/(\log n)^{100}$; i.e., once we have finished revealing a set $\mathcal{S}^{t+1}(\{u,v\})$, and see that it is larger than $n/(\log n)^{100}$, then we terminate the entire procedure and stop revealing further edges (these further edges will not be relevant for the property we are trying to study). This ensures that, for the rest of the proof, we can assume that $\mathcal{S}^t(\{u,v\})$ constitutes a relatively small proportion of the vertices.

Going forward, the broad strategy is as follows. Recall from the discussion at the end of Step 1 that we are trying to prove that, with probability $1-o(n^{-2})$, (4.4) always implies (4.5), by proving that

$\mathcal{S}^{q+1}(\{u,v\})$ tends to have size at least about $(1+np)|\mathcal{S}^{\leq q}(\{u,v\})|$ for all q. We will prove this by considering the above exploration process for $t \leq q$.

First, we will see that type-Z critical moments almost always succeed. That is to say, for all t, typically almost all of the vertices of Z^t end up in $\mathcal{S}^{t+1}(\{u,v\})$. So, if for any step $t \leq q$ the set Z^t occupies a sufficiently large fraction of W^t, it is easy to show that $\mathcal{S}^{\leq q+1}(\{u,v\})$ is large enough to satisfy (4.5). We can therefore assume that most vertices in W^t do not lie in Z^t, and it essentially suffices to restrict our attention to the vertices in $W^t \setminus Z^t$ throughout the revelation procedure.

Second, we will see that type-W critical moments succeed with probability (at least) about p. For a vertex *not* to end up in $\mathcal{S}^{\leq q+1}$, we must have (roughly speaking, ignoring the effect of the sets Z^t) failed at least $|\mathcal{S}^{\leq q}(\{u,v\})|$ type-W critical moments, meaning that we expect $\mathcal{S}^{\leq q+1}(\{u,v\}) = \mathcal{S}^{\leq q}(\{u,v\}) \cup \mathcal{S}^{q+1}(\{u,v\})$ to have size at least about $(1+np)|\mathcal{S}^{\leq q}(\{u,v\})|$.

Step 3: Coupling with a sequence of coin flips. We first make some observations about the success probabilities in critical moments.

Claim 4.8. *Let $p' = 1 - 400/(\log n)^{99}$.*

- *Every type-Z critical moment succeeds with probability at least p' (conditional on all information revealed until that moment).*
- *Every type-W critical moment advances with probability p. Given that we advance, we then succeed with probability at least p'.*

Proof of claim. For the first bullet point: at a type-Z critical moment, we are revealing the status of $|\mathcal{S}^t(\{u,v\})| < n/(\log n)^{100}$ edges of G_{rand} (recalling that we continue the revelation procedure only as long as $|\mathcal{S}^{\leq t}(\{u,v\})| \leq n/(\log n)^{100}$), and by Fact 4.6, the critical moment succeeds if none of these edges are present. So, the probability that the critical moment succeeds is at least

$$(1-p)^{n/(\log n)^{100}} \geq e^{-2np/(\log n)^{100}} \geq e^{-200/(\log n)^{99}} \geq \left(1 - \frac{400}{(\log n)^{99}}\right) = p',$$

recalling that $p \leq 100 \log n/n$.

The second bullet point is very similar: at a type-W critical moment, we advance with probability p (as that is the probability any particular edge appears in G_{rand}), and then we succeed if no edges are present among a certain set of at most $|\mathcal{S}^{\leq t}(\{u,v\})| - 1 \leq n/(\log n)^{100}$ potential edges; by the same calculation as above this happens with probability at least p'. ■

Now, let $(\alpha_i)_{i\in\mathbb{N}} \in \{0,1\}^{\mathbb{N}}$ be a sequence of independent Bernoulli(p') random variables, and let $(\beta_i)_{i\in\mathbb{N}} \in \{0,1\}^{\mathbb{N}}$ be a sequence of independent Bernoulli(p) random variables (such that $(\alpha_i)_{i\in\mathbb{N}}$ and $(\beta_i)_{i\in\mathbb{N}}$ are independent). We can couple these random variables with our revelation procedure in such a way that:

- whenever we encounter a Z-critical moment, we check the next unrevealed random variable α_i, and if $\alpha_i = 1$ then the critical moment succeeds;
- whenever we encounter a W-critical moment, we check the next unrevealed random variable β_j, and if $\beta_j = 1$ then the critical moment advances. We then check the next unrevealed random variable α_i, and if $\alpha_i = 1$ then the critical moment succeeds.

Let $R_\alpha^{\leq q}$ be the total number of α_i that we reveal in the first q steps of the revelation procedure (i.e., up to the point where we have determined $\mathcal{S}^{\leq q+1}(\{u,v\})$). Let $N_\alpha^{\leq q}$ be the number of times we see $\alpha_i = 1$, among these $R_\alpha^{\leq q}$ different α_i. Similarly, let $R_\beta^{\leq q}$ be the total number of β_j that we reveal in the first q steps of the revelation procedure, and let $N_\beta^{\leq q}$ be the number of these β_j that are equal to 1.

Step 4: Interpreting the coin flips. Every time we see $\alpha_i = 1$ (at step t, say), we add a new vertex to $\mathcal{S}^{t+1}(\{u,v\})$. So,

$$|\mathcal{S}^{\leq q+1}(\{u,v\})| \geq N_\alpha^{\leq q}. \tag{4.6}$$

Every time we see $\beta_j = 1$, we reveal a new α_i, so

$$R_\alpha^{\leq q} \geq N_\beta^{\leq q}. \tag{4.7}$$

Also, note that at step t, exactly one α_i is revealed for each $z \in Z^t$, so $|Z^t| \leq R_\alpha^{\leq q}$ for each $t \leq q$. We can also relate $R_\beta^{\leq q}$ to $|\mathcal{S}^{\leq q}(\{u,v\})|$, as follows.

Claim 4.9. *If $|\mathcal{S}^{\leq q}(\{u,v\})| \leq n/(\log n)^{100}$ then*

$$R_\beta^{\leq q} \geq |\mathcal{S}^{\leq q}(\{u,v\})| \cdot \left(n - 2R_\alpha^{\leq q} - \frac{n}{(\log n)^{100}}\right).$$

Proof of claim. For every $t \leq q$ and every pair of vertices $s \in \mathcal{S}^t(\{u,v\})$ and $w \in W^t \setminus Z^t$, the only reason there might not be a type-W critical moment for sw at step t (i.e., the only reason sw might not contribute to $R_\beta^{\leq q}$) is if sw has already been revealed in a type-W critical moment for some other pair $s'w$ (which advanced, and may or may not have succeeded; in particular it contributed to $R_\alpha^{\leq q}$). So,

$$\begin{aligned}
R_\beta^{\leq q} &\geq \sum_{t \leq q} |\mathcal{S}^t(\{u,v\})| \cdot |W^t \setminus Z^t| - \max_{t \leq q} |\mathcal{S}^t(\{u,v\})| \cdot R_\alpha^{\leq q} \\
&\geq \sum_{t \leq q} |\mathcal{S}^t(\{u,v\})| \cdot \big(n - |\mathcal{S}^{\leq t}(\{u,v\})| - |Z^t|\big) - |\mathcal{S}^{\leq q}(\{u,v\})| \cdot R_\alpha^{\leq q} \\
&\geq |\mathcal{S}^{\leq q}(\{u,v\})| \cdot \big(n - |\mathcal{S}^{\leq q}(\{u,v\})| - R_\alpha^{\leq q}\big) - |\mathcal{S}^{\leq q}(\{u,v\})| \cdot R_\alpha^{\leq q}.
\end{aligned}$$

The desired result follows, recalling our assumption $|\mathcal{S}^{\leq q}(\{u,v\})| \leq n/(\log n)^{100}$. ■

Step 5: Concluding with Chernoff bounds. Let $\mathcal{E}_\alpha$ be the event that for every $r \geq n/(\log n)^{100}$ there are at most $800r/(\log n)^{99}$ different "0"s among the first r of the α_i. If $\mathcal{E}_\alpha$ holds, then for all q satisfying $R_\alpha^{\leq q} \geq n/(\log n)^{100}$, we have

$$N_\alpha^{\leq q} \geq (1 - 800/(\log n)^{99}) R_\alpha^{\leq q}. \tag{4.8}$$

Recall that $p' = 1 - 400/(\log n)^{99}$, so by a Chernoff bound (Lemma 2.26) and a union bound over r, we see that

$$1 - \mathbb{P}[\mathcal{E}_\alpha] \leq \sum_{r = n/(\log n)^{100}}^{\infty} \exp\left(-\Omega\left(\frac{r}{((\log n)^{99})^2}\right)\right) = o(n^{-2}),$$

i.e., $\mathcal{E}_\alpha$ holds with probability $1 - o(n^{-2})$.

Similarly, let $\mathcal{E}_\beta$ be the event that for every $s \geq 100 \log n / \min\{(np-1)^2, 1\}$, there are at least $(1 + 2(np-1)/3)s$ different "1"s among the first sn of the β_i. By a Chernoff bound and a union bound over s, we see that

$$1 - \mathbb{P}[\mathcal{E}_\beta] \leq \sum_{s = 100 \log n / \min\{(np-1)^2, 1\}}^{\infty} \exp\left(-\left(\frac{np-1}{3}\right)^2 spn/2\right) = o(n^{-2}).$$

i.e., $\mathcal{E}_\beta$ holds with probability $1 - o(n^{-2})$.

Now, recall that we are trying to prove that, with probability $1 - o(n^{-2})$, whenever q satisfies (4.4) it also satisfies (4.5). It suffices to show that this property follows from $\mathcal{E}_\alpha \cap \mathcal{E}_\beta$. So, for the purpose of contradiction, suppose that $\mathcal{E}_\alpha \cap \mathcal{E}_\beta$ holds, and suppose that there is some q for which

$$\frac{100 \log n}{\min\{(np-1)^2, 1\}} \leq |\mathcal{S}^{\leq q}(\{u,v\})| \leq \frac{n}{(\log n)^{100}},$$

but

$$|\mathcal{S}^{\leq q+1}(\{u,v\})| < \left(1 + \frac{np-1}{2}\right) |\mathcal{S}^{\leq q}(\{u,v\})| \leq \frac{50n}{(\log n)^{99}}, \tag{4.9}$$

recalling that $p \leq 100 \log n / n$. First, note that (4.6), (4.8), and (4.9) imply that

$$R_\alpha^{\leq q} \leq \max\left\{\frac{n}{(\log n)^{100}}, 2N_\alpha^{\leq q}\right\} \leq \max\left\{\frac{n}{(\log n)^{100}}, 2|\mathcal{S}^{\leq q+1}(\{u,v\})|\right\} \leq 100n/(\log n)^{99}.$$

This and Claim 4.9 imply that

$$R_\beta^{\leq q} \geq |\mathcal{S}^{\leq q}(\{u,v\})| \cdot \left(n - \frac{200n}{(\log n)^{99}} - \frac{n}{(\log n)^{100}}\right) \geq |\mathcal{S}^{\leq q}(\{u,v\})| \cdot \left(n - \frac{201n}{(\log n)^{99}}\right),$$

while (4.6) to (4.9) imply that

$$N_\beta^{\leq q} \leq R_\alpha^{\leq q} \leq \frac{N_\alpha^{\leq q}}{1 - 800/(\log n)^{99}} \leq \frac{|\mathcal{S}^{\leq q+1}(\{u,v\})|}{1 - 800/(\log n)^{99}} \leq \frac{(1 + (np-1)/2)|\mathcal{S}^{\leq q}(\{u,v\})|}{1 - 800/(\log n)^{99}}.$$

Recalling that $np - 1 \geq (\log n)^{-40}$, these inequalities contradict $\mathcal{E}_\beta$ (taking $s = |\mathcal{S}^{\leq q}(\{u,v\})|$ in the definition of $\mathcal{E}_\beta$). □

5. Distinguishing vertices in the 2-core

In the previous section we showed that given two vertices u, v in an appropriately randomly perturbed graph G, it is very likely that whenever the sets $\mathcal{S}^i(\{u,v\})$ grow reasonably large, u and v end up with different colours in the stable colouring $\mathcal{R}^*\sigma$. In this section we prove that this condition is typically satisfied for pairs of vertices u, v satisfying certain combinatorial conditions (defined in terms of the *2-core* of the random perturbation graph G_{rand}). In particular, the main result of this section will directly imply Theorem 1.2 (as we will see at the end of this section).

Definition 5.1. Recall from Definition 1.7 that the k-core of a graph G is its maximum subgraph with minimum degree at least k.

- Write $V_k(G)$ for the vertex set of this k-core (so the k-core itself is $\mathrm{core}_k(G) = G[V_k(G)]$).
- As in Definition 2.10, let $V_{2,3}(G) \subseteq V_3(G)$ be the set of vertices which have degree at least 3 in the 2-core $G[V_2(G)]$.
- Let $V_{2,3}^{\mathrm{safe}}(G) \subseteq V_{2,3}(G)$ be the set of vertices $v \in V_{2,3}(G)$ satisfying the following property. If we delete any two vertices (other than v), the connected component of v still contains at least three vertices of $V_{2,3}(G)$ (including v).

In words, note that the 2-core can be viewed as a network of "hubs" (vertices with degree at least 3), joined by bare paths in which every interior vertex has degree 2. Each hub has at least three bare paths emanating from it, but it could happen that several of these bare paths lead to the same hub. We say a hub v is "safe" if it is not possible to delete two vertices to create a tiny connected component containing v and at most one other hub.

Proposition 5.2. *Let G_0 be a graph on the vertex set $\{1, \dots, n\}$ and let $G_{\mathrm{rand}} \sim G(n,p)$ for some $p \in [0, 1/2]$ satisfying*

$$p \geq \frac{1 + (\log n)^{-40}}{n}.$$

Let $G = G_0 \triangle G_{\mathrm{rand}}$. Then whp G has the property that $\mathcal{R}^*_G\sigma(u) \neq \mathcal{R}^*_G\sigma(v)$ for every pair of distinct vertices u, v satisfying one of the following assumptions.

A1 *$u \in V_{2,3}^{\mathrm{safe}}(G_{\mathrm{rand}})$, or*

A2 *$u, v \in V_{2,3}(G_{\mathrm{rand}})$, and G_0 is the empty graph.*

The idea of the proof of Proposition 5.2 is as follows. We fix two distinct vertices u, v, and consider the same exploration procedure that featured in the proofs of Proposition 4.1 and Lemma 4.2: at step t we reveal the edges of G_{rand} incident to $\mathcal{S}^t(\{u,v\})$ to determine the next set $\mathcal{S}^{t+1}(\{u,v\})$. Vertices of $\mathcal{S}^t(\{u,v\})$ which lie in $V_{2,3}(G_{\mathrm{rand}})$ are typically connected, via bare paths in $V_2(G)$, to at least three vertices of $V_{2,3}(G_{\mathrm{rand}})$. Our exploration process will follow these paths, and typically each of these "neighbouring" vertices in $V_{2,3}(G_{\mathrm{rand}})$ will end up in some future $\mathcal{S}^{t'}(\{u,v\})$. That is to say, each time our exploration process reaches a hub, it branches out into at least two more hubs; the resulting exponential growth means it will not take long for $\mathcal{S}^{\leq t}(\{u,v\})$ to grow large enough to apply Proposition 4.1. Indeed, if the exploration process "gets stuck" (with $\mathcal{S}^{i+1}(\{u,v\}) = \emptyset$) before (4.1) is satisfied, some very unlikely events need to have happened: we need to have been in a subset of the 2-core with very poor expansion properties, or some edges must have shown up in atypical places during the iterative revelation process.

Proof of Proposition 5.2. Before we do anything else, the first step is to reduce to the case where p is in a suitable range to apply Proposition 4.1.

Step 1: Making assumptions on p. We can assume $p \leq 2\log n/n$, because for larger p we can simply view the random perturbation as a composition of two random perturbations. Specifically, we can choose $p_1 \in [0,1]$ in such a way that, if we take $G'_{\mathrm{rand}} \sim \mathbb{G}(n, 2\log n/n)$ and $G_1 \sim \mathbb{G}(n, p_1)$, then G_{rand} has the same distribution as $G_1 \triangle G'_{\mathrm{rand}}$. (Namely, we need p_1 to satisfy the equation $p_1(1 - 2\log n/n) + (2\log n/n)(1-p_1) = p$). Then, letting $G'_0 = G_0 \triangle G_1$ we have $G = G'_0 \triangle G'_{\mathrm{rand}}$, so after conditioning on an outcome of G'_0 we are considering a randomly perturbed graph with edge perturbation probability exactly $2\log n/n$. Moreover, whp the sets $V_{2,3}^{\mathrm{safe}}(G'_{\mathrm{rand}})$ and $V_{2,3}^{\mathrm{safe}}(G_{\mathrm{rand}})$ are identical: in this regime whp both, G'_{rand} and G_{rand} are 3-connected (see for example [36, Theorem 4.3]) hence $V_{2,3}^{\mathrm{safe}}(G'_{\mathrm{rand}}) = V_3(G'_{\mathrm{rand}}) = V(G) = V_3(G_{\mathrm{rand}}) = V_{2,3}^{\mathrm{safe}}(G_{\mathrm{rand}})$.

Step 2: Expansion. We observe a (very weak) expansion property of G_{rand}: whp there are no small vertex cuts that disconnect a small dense subgraph.

Claim 5.3. *Whp G_{rand} has the following property. Consider disjoint vertex sets T, U, where $|T| \le 2$ and $|U| \le (\log n)^{100}$. Suppose that G_{rand} has no edges between U and $V(G) \setminus T$ (i.e., T disconnects U from the rest of the graph), and suppose that $G_{\mathrm{rand}}[T \cup U]$ is connected. Then the number of edges in $G_{\mathrm{rand}}[U \cup T]$ is at most $|U| + |T|$.*

Proof of claim. If the desired conclusion were to fail, there would be a pair of vertex sets U, T, with $|T| \le 2$ and $|U| \le (\log n)^{100}$, such that G_{rand} has no edges between U and $V(G) \setminus T$, and such that $G_{\mathrm{rand}}[U \cup T]$ has at least $|U| + |T| + 1$ edges.

Summing over possibilities for $t = |U| + |T|$, the probability that such a pair of vertex sets exists is at most

$$\sum_{t=4}^{(\log n)^{100}+2} \binom{n}{t} t^2 t^{t-2} (t^2)^2 p^{t+1} (1-p)^{(t-2)(n-t)} \le 2 \sum_{t=4}^{n^{o(1)}} \left(\frac{en}{t}\right)^t t^{t+4} p^{t+1} e^{-pnt} e^{2np}$$
$$\le 2 \sum_{t=4}^{n^{o(1)}} (enpe^{-pn})^t e^{2np} pt^4 = o(1).$$

In the last equality we used that if $np \le 1000$ then e^{2np}, whereas if $np \ge 1000$ then $(enpe^{-pn})^t e^{2np} \ge e^{-tnp/10}$, for $t \ge 4$.

To provide a bit of explanation for the above calculation: first note that $U \cup T$ always spans at most $|U| + |T|$ edges if $|U| + |T| \le 3$, so we only need to consider $t \ge 4$. The number of possible ways to choose U and T is at most $\binom{n}{t} t^2$ (first choose $U \cup T$, then choose T). Then, note that $G_{\mathrm{rand}}[U \cup T]$, being connected, must contain a spanning tree (which has $t-1$ edges, and can be chosen in t^{t-2} different ways, by Cayley's formula), plus two additional edges (which can be chosen in at most $(t^2)^2$ different ways). These $(t-1) + 2 = t + 1$ edges are present in G_{rand} with probability p^{t+1}. Also, since T disconnects U from the rest of the graph, there are $(t-2)(n-t)$ specific pairs of vertices which are *not* edges of G_{rand}, namely the edges from U to $G - (U \cup T))$ (which happens with probability $(1-p)^{(t-2)(n-t)}$).

In the last line, we used the inequality $xe^{-x} \le 1/e$ (which holds for all $x \in \mathbb{R}$), and the assumption $p = n^{o(1)-1}$. ■

Step 3: Exploration. Define

$$f(n) = \frac{100 \log n}{\min\{(np-1)^2, 1\}} \le 100 \log^{81} n \le (\log n)^{100}.$$

Fix distinct vertices u, v. We will prove that with probability at least $1 - o(n^{-2})$, either the property in Claim 5.3 fails to hold or there is i such that $|\mathcal{S}^{\le i}(\{u, v\})| \ge f(n)$. The desired result will then directly follow from a union bound over u, v, and Proposition 4.1.

As in the proofs of Proposition 4.1 and Lemma 4.2, we consider an exploration/revelation procedure. In each step t, if we do not already have $|\mathcal{S}^{\le t+1}(\{u, v\})| \ge f(n)$ (this is condition (4.1) in Proposition 4.1), then we reveal the edges of G_{rand} incident to $\mathcal{S}^t(\{u, v\})$ to determine the next set $\mathcal{S}^{t+1}(\{u, v\})$. This process terminates at the end of the first step i for which $|\mathcal{S}^{\le i+1}(\{u, v\})| \ge f(n)$ or $\mathcal{S}^{i+1}(\{u, v\}) = \emptyset$.

Our goal is to rule out (with probability $1 - o(n^{-2})$) the possibility that **A1** or **A2** holds, $\mathcal{S}^{i+1}(\{u, v\}) = \emptyset$ and $|\mathcal{S}^{\le i}(\{u, v\})| < f(n)$.

Step 4: Atypical events during exploration. Recall the definitions of the sets $Z^t \subseteq W^t$ from Step 2 of the proof of Lemma 4.2 (in Section 4.1). Namely, W^t is the set of "candidate vertices" for $\mathcal{S}^{t+1}(\{u, v\})$, and Z^t is a subset of vertices that are especially likely to end up in $\mathcal{S}^{t+1}(\{u, v\})$.

Recalling Fact 4.7, if there is exactly one edge between $\mathcal{S}^t(\{u, v\})$ and $w \in W^t \setminus Z^t$ (with respect to G_{rand}), then we have $w \in \mathcal{S}^{t+1}(\{u, v\})$. That is to say, an edge of G_{rand} between $\mathcal{S}^t(\{u, v\})$ and $w \in W^t$ always contributes to the growth of $\mathcal{S}^{t+1}(\{u, v\})$, unless $w \in Z^t$, or unless there is more than one edge between w and $\mathcal{S}^t(\{u, v\})$. The following claim shows that these two possibilities happen very rarely (unless $|\mathcal{S}^{\le i}(\{u, v\})|$ is large, in which case we are done).

Claim 5.4. *For $t \le i$, let $Z^t_{\mathrm{bad}} \subseteq Z^t$ be the set of vertices in Z^t which are adjacent to $\mathcal{S}^t(\{u, v\})$ with respect to G_{rand}, and let W^t_{bad} be the set of vertices in $W^t \setminus Z^t$ with more than one neighbour in $\mathcal{S}^t(\{u, v\})$, again with respect to G_{rand}. Then, with probability $1 - o(n^{-2})$, we have*

$$|\mathcal{S}^{\le i+1}(\{u, v\})| \ge f(n) \quad \text{or} \quad \sum_{t=0}^{i} \left(|W^t_{\mathrm{bad}}| + |Z^t_{\mathrm{bad}}|\right) \le 2.$$

Proof of claim. First, note that with probability $1-o(n^{-2})$, we have $|Z^t| < 2f(n)$ for all $t \le i$ such that $|\mathcal{S}^{\le t+1}(\{u,v\})| < f(n)$. Indeed, consider the *first* step t such that $|Z^t| \ge 2f(n)$ (if such a step exists), and suppose that $|\mathcal{S}^{\le t}(\{u,v\})| < f(n)$. Conditioning on the information revealed up to step t, by the same argument as given in Claim 4.8, each vertex in Z^t independently ends up in $\mathcal{S}^{t+1}(\{u,v\})$ with probability $1-o(1)$, so by a Chernoff bound (Lemma 2.26), we have $|\mathcal{S}^{t+1}(\{u,v\})| \ge |Z^t|/2 \ge f(n)$ with probability at least $1-o(n^{-2})$, in which case the desired property holds.

Now, let $X_W = \sum_t |W^t_{\text{bad}}|$, where the sum is over all $t \le f(n)$ for which $|\mathcal{S}^t(\{u,v\})| < f(n)$, and let $X_Z = \sum_t |Z^t_{\text{bad}}|$, where the sum is over all $t \le f(n)$ for which $|Z^t| < 2f(n)$ and $|\mathcal{S}^t(\{u,v\})| < 100\log n$. It suffices to show that $X_W + X_Z \le 2$ with probability $1-o(n^{-2})$.

Note that, if we condition on the information revealed up to step t (i.e., after we have revealed $\mathcal{S}^t(\{u,v\})$, but before we have revealed $\mathcal{S}^{t+1}(\{u,v\})$), then each vertex $w \in W^t \setminus Z^t$ ends up in W^t_{bad} with probability at most $p^2|\mathcal{S}^t(\{u,v\})|^2 = n^{-2+o(1)}$, and each vertex $z \in Z^t$ ends up in Z^t_{bad} with probability at most $p|\mathcal{S}^t(\{u,v\})| = n^{-1+o(1)}$. So, $X_W + X_Z$ is stochastically dominated by a sum of at most $f(n)\cdot n = n^{1+o(1)}$ Bernoulli trials with probability $n^{-2+o(1)}$, plus a sum of at most $f(n)\cdot 2f(n) = n^{o(1)}$ Bernoulli trials with probability $n^{-1+o(1)}$ (all independent). We conclude that

$$\begin{aligned}\mathbb{P}[X_W + X_Z \ge 3] &\le \sum_{q=0}^{3} \mathbb{P}[X_W \ge q,\, X_Z \ge 3-q] \\ &\le \sum_{q=0}^{3} \binom{n^{1+o(1)}}{q} (n^{-2+o(1)})^q \cdot \binom{n^{o(1)}}{3-q} (n^{-1+o(1)})^{3-q} = n^{-3+o(1)} = o(n^{-2}),\end{aligned}$$

as desired. ■

Step 5: Concluding. We now wish to prove that if **A1** or **A2** are satisfied (i.e., $u \in V^{\text{safe}}_{2,3}(G_{\text{rand}})$, or $G_0 = \emptyset$ and $u, v \in V_{2,3}(G_{\text{rand}})$), then the properties in Claims 5.3 and 5.4 imply that $|\mathcal{S}^{\le i+1}(\{u,v\})| \ge f(n)$. So, suppose for the purpose of contradiction that **A1** or **A2** is satisfied, and the properties in Claims 5.3 and 5.4 hold, and for some i, $|\mathcal{S}^{\le i}(\{u,v\})| < f(n)$, and $\mathcal{S}^{i+1}(\{u,v\}) = \emptyset$.

Let $T = \bigcup_{t=0}^{i}(W^t_{\text{bad}} \cup Z^t_{\text{bad}})$, so $|T| \le 2$ by the property in Claim 5.4. As discussed in Step 3, every edge of G_{rand} between $\mathcal{S}^t(\{u,v\})$ and $w \in W^t$ contributes to $\mathcal{S}^{\le t+1}(\{u,v\})$, unless $w \in W^t_{\text{bad}} \cup Z^t_{\text{bad}}$. Since $\mathcal{S}^{i+1}(\{u,v\}) = \emptyset$, this means T disconnects $\mathcal{S}^{\le i}(\{u,v\})$ from the rest of G_{rand}. Let U_u and U_v be the vertex sets of the connected components of u and v, with respect to $G_{\text{rand}} - T$. Note that $U_u \cup U_v$, being a subset of $\mathcal{S}^{\le i}(\{u,v\})$, has fewer than $f(n) \le (\log n)^{100}$ vertices.

First, suppose **A1** is satisfied (i.e., $u \in V^{\text{safe}}_{2,3}(G_{\text{rand}})$). By the definition of $V^{\text{safe}}_{2,3}(G_{\text{rand}})$, the connected component of u in $G_{\text{rand}} - T$ always contains at least three vertices in $V_{2,3}(G_{\text{rand}})$. So, $G_{\text{rand}}\big[(T \cup U_u) \cap V_2(G_{\text{rand}})\big]$ has at least three vertices of degree at least 3, and at most two vertices of degree 1 (these can only be vertices in T). It follows that $G_{\text{rand}}\big[(T \cup U_u) \cap V_2(G_{\text{rand}})\big]$ has more edges than vertices, and the same holds for $G_{\text{rand}}\big[T \cup U_u\big]$ (recall that a graph is obtained from its 2-core by adding "dangling trees", which do not change the balance of vertices and edges), contradicting Claim 5.3.

Second, suppose **A2** is satisfied (i.e., $G_0 = \emptyset$ and $u, v \in V_{2,3}(G_{\text{rand}})$). Since G_0 is the empty graph, we have $Z^t_{\text{bad}} = Z^t = \emptyset$ for each t. So, every vertex in T lies in some W^t_{bad}, meaning it has at least two neighbours in $\mathcal{S}^{\le i}(\{u,v\})$ (which in turn are connected to u or v, via some path). So, all vertices in $G_{\text{rand}}\big[(T \cup U_u \cup U_v) \cap V_2(G_{\text{rand}})\big]$ have degree at least 2; since u and v have degree at least 3, it follows that $G_{\text{rand}}\big[(T \cup U_u \cup U_v) \cap V_2(G_{\text{rand}})\big]$ has more edges than vertices, and the same holds for $G_{\text{rand}}\big[T \cup U_u \cup U_v\big]$. This graph may not be connected, but at least one of its connected components contradicts Claim 5.3. □

Before ending this section, we record a simple consequence of Proposition 5.2: for any $k \ge 3$, whp the vertices in the 3-core of G_{rand} are all assigned unique colours by the colour refinement algorithm. Recall that $V_k(G)$ is the vertex set of the k-core of a graph G, and note that $V_k(G) \subseteq V_{k'}(G)$ when $k' \le k$.

Proposition 5.5. *Let G_0 be a graph on the vertex set $\{1,\dots,n\}$, let $G_{\text{rand}} \sim G(n,p)$ for some $p \in [0,1/2]$, and let $G = G_0 \triangle G_{\text{rand}}$. Then whp G has the property that $\mathcal{R}^*_G\sigma(u) \ne \mathcal{R}^*_G\sigma(v)$ for every pair of distinct vertices u, v such that $u \in V_3(G_{\text{rand}})$.*

Proof. It was famously proved by Pittel, Spencer and Wormald [70] that there is a constant $c \approx 3.35$ such that if $p \le c/n$ then $V_3(G_{\text{rand}}) = \emptyset$ whp. So, we can assume that $p \ge 3/n$.

Also, it is easy to prove (see for example [36, Theorem 3.2]) that if $p \ge (1+\varepsilon)\log n/n$ for any fixed $\varepsilon > 0$, then whp every vertex of G_{rand} has degree at least 4; that is, $V_{2,3}(G) = V_4(G) = V(G)$. Note that if a vertex $v \in V_{2,3}(G)$ has at least four neighbours in $V_{2,3}(G)$, then trivially $v \in V_{2,3}^{\mathrm{safe}}$, so in this case the desired result follows directly from Proposition 5.2. We can therefore assume that $p \leq 2\log n/n$.

Now, it suffices to prove that whp $V_3(G_{\mathrm{rand}}) \subseteq V_{2,3}^{\mathrm{safe}}(G_{\mathrm{rand}})$. Consider $v \in V_3(G_{\mathrm{rand}}) \subseteq V_{2,3}(G_{\mathrm{rand}})$. Note that v has at least three neighbours with degree at least 3 in $V_{2,3}(G_{\mathrm{rand}})$. The only way that deletion of two vertices other than v could possibly cause the connected component of v to contain fewer than three vertices of $V_{2,3}(G_{\mathrm{rand}})$ is if the two deleted vertices x, y are both neighbours of v, and there is a third neighbour of v whose neighbours in $V_{2,3}(G_{\mathrm{rand}})$ are precisely x, y, v. If this situation occurs, then G has a subgraph with four vertices and at least five edges. It is easy to see that whp no such subgraph exists in G_{rand}, given our assumption that $p \leq 2\log n/n$ (indeed, the expected number of such subgraphs is $O(p^5 n^4) = o(1)$). □

As observed in the proof of Proposition 5.5, for any fixed $\varepsilon > 0$, if $G_{\mathrm{rand}} \sim \mathbb{G}(n, p)$ with $p \geq (1+\varepsilon)\log n/n$, then whp every vertex of G_{rand} has degree at least 3, so $V_3(G_{\mathrm{rand}}) = V(G)$. So, Proposition 5.5 implies that $\mathcal{R}^*\sigma$ gives every vertex a different colour. That is to say, Theorem 1.2 follows directly from Proposition 5.5 and Theorem 2.3.

6. Small components with very mild perturbation

In this section we prove Theorem 1.3. The proof is fairly involved, so we start with a rough description of the overall structure of the proof, supplementing the discussion in Section 1.3. In fact, the reader may benefit from re-reading Section 1.3.8 before starting with this section. We also remark that a much simpler implementation of similar ideas can be found in Appendix B, where we prove a weaker version of Theorem 1.3. Recall the definitions of the colour refinement and 2-dimensional Weisfeiler–Leman algorithms, from Definitions 2.2 and 2.20.

Recalling the notation in Definitions 1.10 and 2.20 (on disparity graphs and the 2-dimensional Weisfeiler–Leman algorithm), our goal is to prove that if $G = G_0 \triangle G_{\mathrm{rand}}$ is a randomly perturbed graph with perturbation probability as small as $O(1/n)$, then the connected components of the disparity graph $D(G, \mathcal{V}^*\phi_G)$ have size $O(\log n)$. Indeed, we saw in Corollary 2.7 and Fact 2.22 that there is an efficient canonical scheme for graphs satisfying this property.

Towards this goal, we view the random perturbation G_{rand} as the union of eight slightly sparser independent random perturbations $G^1_{\mathrm{rand}}, \dots, G^8_{\mathrm{rand}} \sim \mathbb{G}(n, p')$ (this is called *sprinkling*). Assuming $p \geq 100/n$ (thus $p' > 10/n$), standard results show that the 3-core of G^1_{rand} (which is a subgraph of the 3-core of G_{rand}) has at least $n/2$ vertices, and by Proposition 5.2, whp all these vertices are assigned unique colours by $\mathcal{R}^*_G\sigma$ (therefore by $\mathcal{V}^*\phi_G$, recalling Fact 2.24).

So, we first reveal the 3-core of G^1_{rand}, and fix a set V_{core} of $n/2$ vertices in this 3-core. We then define the complementary set $U = V(G) \setminus V_{\mathrm{core}}$, and reveal all edges of G_{rand} except those between U and V_{core}. For the rest of the proof we work with the remaining seven rounds of random edges (in $G^2_{\mathrm{rand}}, \dots, G^8_{\mathrm{rand}}$) between U and V_{core}.

Specifically, whenever a vertex in U has at least three neighbours in V_{core}, that vertex is guaranteed to be in the 3-core of G_{rand}, and therefore we can assume it will be assigned a unique colour. So, we can consider a vertex-colouring of $G[U]$ that is refined with each round of sprinkling: at each round, new random vertices in U are assigned unique colours, and then the 2-dimensional Weisfeiler–Leman algorithm propagates this information to obtain a new stable vertex-colouring. In each round, we show that the colour classes, and the components of the disparity graph, are broken into smaller and smaller pieces.

6.1. Preparatory lemmas.

Before we get into the details of the proof of Theorem 1.3, we start with a few general graph-theoretic lemmas.

First, we need two near-trivial facts about disparity graphs with respect to equitable colourings: in a disparity graph, the density between any pair of colour classes is at most 1/2, and refinement cannot increase degrees in the disparity graph. Recall the definition of equitability from Definition 2.15, and recall the definition of the generalised disparity graph $D_L(G, c)$ from Definition 2.18 (if c_0 is a coarsening of c, then $D(G, c_0)$ can be interpreted as a generalised disparity graph of the form $D_L(G, c)$).

Fact 6.1. *Consider any graph G and any equitable colouring $c : V(G) \to \Omega$, and let A, B be two distinct colour classes of c.*

- *$D(G, c)[A]$ is regular, with degree at most $(|A| - 1)/2$.*
- *$D(G, c)[A, B]$ is biregular. In this bipartite subgraph, the common degree of vertices in A is at most $|B|/2$, and the common degree of vertices in B is at most $|A|/2$.*

Fact 6.2. *Let $c : V(G) \to \Omega$ be an equitable vertex-colouring of a graph G. Then for every vertex v, the degree of v in $D(G, c)$ is at most the degree of v in any generalised disparity graph $D_L(G, c)$ (and therefore at most the degree of v in $D(G, c_0)$, for any possibly non-equitable coarsening c_0 of c).*

Next, the following lemma shows that refining a colouring cannot significantly increase the sizes of connected components of the disparity graph.

Lemma 6.3. *Let $c : V(H) \to \Omega$ be an equitable vertex-colouring of a graph H, and consider a matrix $L \in \{0,1\}^{\Omega\times\Omega}$. Let $H_1 = D_L(H, c)$ and $H_2 = D(H, c)$, and for a vertex v, let $X_1(v), X_2(v)$ be the (vertex sets of) the connected components containing v in the graphs H_1 and H_2, respectively. Then*

$$|X_2(v)| \le 2|X_1(v)|.$$

(The same inequality therefore holds when $H_1 = D(H, c_0)$ for some coarsening c_0 of c).

The proof of Lemma 6.3 uses the equitability of c together with the following simple structural fact (see [19, Exercise 1.1.12]). The "special" graphs in the second bullet point of Fact 6.4 may already give some hint for why the factor of two in Lemma 6.3 is required.

Fact 6.4. *Let H be a graph.*

- *If H has at least $\binom{|V(H)|}{2}/2$ edges, then it is connected.*
- *If H is a biregular bipartite graph with bipartition A, B and at least $|A||B|/2$ edges, then it is connected, unless H is a disjoint union of two isomorphic complete bipartite graphs (in which case H has exactly $|A||B|/2$ edges). In this latter case we say H is "special".*

Proof of Lemma 6.3. Fix a vertex v, and let $C_1, C_2, ..., C_k \subseteq V(G)$ be the colour classes which contain at least one vertex of the component $X_2(v)$. Let $X_1^*(v)$ be the (vertex set of the) component of v in $H_1[C_1 \cup \dots \cup C_k]$, so $X_1^*(v) \subseteq X_1(v)$ and it suffices to show that

$$|X_1^*(v)| \ge \frac{|X_2(v)|}{2}. \tag{6.1}$$

First, we show that the paths in H_2 between $C_1, \dots, C_k$ imply the existence of corresponding paths in H_1.

Claim 6.5. *Consider any $i \le k$ and any vertex $u \in C_j$. In the graph $H_1[C_1 \cup \dots \cup C_k]$ there is a path between u and C_i.*

Proof of claim. By the choice of $C_1, \dots, C_k$, note that for each i, j there is a path in $H_2[C_1 \cup \dots \cup C_k]$ between some vertex of C_i and some vertex of C_j. But if there is any edge in $H_2 = D(H, c)$ between two colour classes C_i and C_j, then $D(H, c)[C_i, C_j]$ is neither complete nor empty[18] (recalling Fact 6.1), and the same is true for $H_1[C_i, C_j] = D_L(H, c)[C_i, C_j]$. So, for each i, j there is a path in $H_1[C_1 \cup \dots \cup C_k]$ between some vertex of C_i and some vertex of C_j, and the desired result follows from Fact 2.19. ■

Now, we break into cases depending on the structure of the graphs $H_1[C_i, C_j]$. Recall the structural description in Fact 6.4, and the definition of "special".

Case 1: $H_1[C_i, C_j]$ is not special for any $i \ne j$. Note that if $H_1[C_i, C_j] = H_2[C_i, C_j]$ for all i, j, we trivially have $X_2(v) = X_1^*(v)$, and the desired result (6.1) follows. So, we assume that $H_1[C_i, C_j] \ne H_2[C_i, C_j]$ for some i, j. Since H_1, H_2 can both be interpreted as generalised disparity graphs of H with respect to c, this means that $H_1[C_i, C_j]$ is the bipartite complement of $H_2[C_i, C_j]$.

By Fact 6.1, $H_2[C_i, C_j]$ is (bi)regular with at most half the total possible number of edges, so $H_1[C_i, C_j]$ is biregular with at *least* half the total possible number of edges. Since in this case we are assuming that $H_1[C_i, C_j]$ is not special, by Fact 6.4 $H_1[C_i, C_j]$ is connected.

By Claim 6.5, in $H_1[C_1 \cup \dots \cup C_k]$ there is some path from every vertex to C_i. Since $H_1[C_i, C_j]$ is connected, this means that in fact $H_1[C_1 \cup \dots \cup C_k]$ is connected, meaning that $X_1^*(v) = C_1 \cup \dots \cup C_k \supseteq X_2(v)$, and the desired result (6.1) follows.

Case 2: There is a special subgraph. It remains to consider the case where $H_1[C_i, C_j]$ is special for some $i \ne j$. This means that $H_1[C_i, C_j]$ is a disjoint union of two complete bipartite graphs with bipartitions $(C_i^{\mathrm{A}}, C_j^{\mathrm{A}})$ and $(C_i^{\mathrm{B}}, C_j^{\mathrm{B}})$, for some partition of C_i into equal-sized subsets $C_i^{\mathrm{A}}, C_i^{\mathrm{B}}$ and some partition of C_j into equal-sized subsets $C_j^{\mathrm{A}}, C_j^{\mathrm{B}}$. Note that $H_1[C_i^{\mathrm{A}}, C_j^{\mathrm{A}}]$ and $H_1[C_i^{\mathrm{B}}, C_j^{\mathrm{B}}]$ are both connected.

[18] Here we are abusing notation slightly and writing $F[A, A]$ to mean $F[A]$. "Complete" should be understood in context to mean either a complete graph or a complete bipartite graph.

Now, recall that we are assuming that c is equitable for H, and by the definition of a generalised disparity graph it immediately follows that c is equitable for H_1. By Claim 6.5, in $H_1[C_1 \cup \cdots \cup C_k]$ there is some path from every vertex to C_i. So, $H_1[C_1 \cup \cdots \cup C_k]$ has at most two connected components (one including $C_i^{\mathrm{A}}, C_j^{\mathrm{A}}$ and one including $C_i^{\mathrm{B}}, C_j^{\mathrm{B}}$). If there is only one component, we are done as in the previous case. If there are two components, note that both components have the same degree statistics between each pair of parts C_i, C_j (by equitability); since C_i^{A} and C_i^{B} have the same size, this means that both components have exactly half the vertices of $H_1[C_1 \cup \cdots \cup C_k]$. So, $|X_1^*(v)| = |C_1 \cup \cdots \cup C_k|/2 \geq |X_2(v)|/2$; this is the desired inequality (6.1). □

Next, we need a graph-theoretic lemma about distances in sparse graphs (i.e., graphs of bounded degree). Roughly speaking, this lemma says that for any set of vertices C in a sparse connected graph, we can find a vertex y and a set of distances $\mathcal{I}$, such that the number of vertices in C which see y at those distances is not too large and not too small. This means that if C is large, then there are many vertices which see y at distances in $\mathcal{I}$ and many vertices which do not see y at distances in $\mathcal{I}$ (i.e., it is not the case that y has the same distance to almost all vertices).

Lemma 6.6. *Let F be a connected graph of maximum degree at most $\Delta \geq 2$, and fix a vertex subset $C \subseteq V(F)$. Then, for any $1 \leq a \leq |C|$, there is a vertex $y \in V(G)$ and a subset $\mathcal{I} \subseteq \mathbb{N}$, such that the number of vertices $v \in C$ with $\operatorname{dist}_G(y, v) \in \mathcal{I}$ lies in the interval $[a, a\Delta)$.*

Proof of Lemma 6.6. For any vertex y and set $\mathcal{I} \subseteq \mathbb{N}$, let $Q_{\mathcal{I}}(y)$ be the number of vertices in C whose distance to y lies in $\mathcal{I}$. Fix a vertex z.

Case 1: there are at least a vertices in C with the same distance d to z. In this case, let $z_d = z$ and recursively define a sequence of vertices $z_{d-1}, \ldots, z_0$ by choosing z_i to be a neighbour of z such that $Q_{\{i\}}(z_i) \geq Q_{\{i+1\}}(z_{i+1})/\Delta$ (this is possible, because the $Q_{\{i+1\}}(z_{i+1})$ vertices in C which have distance $i+1$ to z_{i+1} are all at distance i to at least one of the at most Δ neighbours of z_{i+1}).

Note that $Q_{\{d\}}(z_d) \geq a$ and $Q_{\{0\}}(z_0) = 1$, so as we gradually decrease i from d to 0, we must find some i for which $Q_{\{i\}}(z_i) \in [a, a\Delta)$ (and then we can take $y = z_i$ and $\mathcal{I} = \{i\}$).

Case 2: for every $d \in \mathbb{N}$, there are at most a vertices in C with distance d to z. In this case, let $\mathcal{I}(i) = \{0, 1, \ldots, i-1\}$. Note that $Q_{\mathcal{I}(0)}(z) = 0$ and $Q_{\mathcal{I}(n)}(z) = |C|$ for large enough n, and note that $Q_{\mathcal{I}(i+1)}(z) - Q_{\mathcal{I}(i)}(z) \leq a$ for each i. So, as we gradually increase i from 0 to n, we must find some i for which $Q_{\mathcal{I}(i)}(z) \in [a, a\Delta)$ (and then we can take $z = y$ and $\mathcal{I} = \mathcal{I}(i)$). □

Finally, the following very simple lemma will be used in the “fingerprinting” argument briefly outlined in Section 1.3.9. For a set of vertices A in a graph G, write $N(A) = \bigcup_{v \in A} N(v)$ for the set of all vertices adjacent to a vertex in A.

Lemma 6.7. *Let F be a non-empty biregular bipartite graph with bipartition $C \cup B$, where the vertices in B have degree d_B. Then there is a subset $S \subseteq B$ of size at most $|B|/2d_B + 1$ with the property that $|N(S)| \geq |C|/4$.*

Proof. Let $S_0 = \emptyset$, and for $i = 1, \ldots, \lceil |C|/(2d_B) \rceil$, recursively define $S_i = S_{i-1} \cup \{z_i\}$, where $z_i \in B$ is a vertex which has the maximum possible number of neighbours in $C \setminus N(S_{i-1})$. For each i, note that $N(S_{i-1}) \leq (i-1)d_B < (|C|/(2d_B))d_B \leq |C|/2$, so $C \setminus N(S_{i-1}) \geq |C|/2$. The degree of every vertex in C is $d_B|B|/|C|$, so the vertices in $C \setminus N(S_{i-1})$ are incident to at least $d_C \cdot |C|/2 = d_B|B|/2$ edges. Thus, z_i is incident to at least $d_B/2$ vertices in $C \setminus N(S_{i-1})$. It follows that $|N(S)| \geq (|C|/2d_B)(d_B/2) = |C|/4$. □

6.2. **Bounding the degrees.** Now we begin to get to the heart of the proof of Theorem 1.3. Recall from the discussion at the start of this section that we are going to define a partition $V(G) = V_{\text{core}} \cup U$, and with each round of sprinkling, each vertex of U will be assigned a unique colour if it has at least three neighbours in V_{core}. Provided that $p \geq 100/n$, we will calculate that in each round of sprinkling, each vertex of U is assigned a unique colour with probability at least 0.7 independently.

Our first key lemma is the simple observation that this random assignment of unique colours typically causes the disparity graph to have maximum degree $O(\log n)$. Indeed, if a vertex v in the disparity graph has large degree, it must have quite a different neighbourhood to some other vertex u in its colour class. But then it is very likely we assign a unique colour to some vertex which is in the neighbourhood of v but not u, or vice versa, so colour refinement will distinguish u and v from each other.

Lemma 6.8. *Consider an n-vertex graph H and an equitable colouring $c : V(H) \to \Omega$. Let c' be a random refinement of c, obtained by assigning each vertex a new unique colour with probability at least 0.7, independently. Then, whp $D(H, \mathcal{R}^* c')$ has maximum degree at most $4 \log n$.*

We remark that in our proof of Theorem 1.3, we will apply Lemma 6.8 with $H = G[U]$.

Proof of Lemma 6.8. For a pair of vertices u, v, let $N_H(u) \triangle N_H(v)$ be the set of vertices in H which are a neighbour of exactly one of u, v.

Claim 6.9. *Whp the following holds. For every pair of vertices u, v with $|N_H(u) \triangle N_H(v)| \geq 2\log n$, we have $\mathcal{R}^* c'(u) \neq \mathcal{R}^* c'(v)$.*

Proof of claim. For any particular pair of vertices u, v, if any of the vertices in $N_H(u) \triangle N_H(v)$ are assigned a unique colour by c' then $\mathcal{R}^* c'(u) \neq \mathcal{R}^* c'(v)$. So, if $|N_H(u) \triangle N_H(v)| \geq 2\log n$ then $\mathbb{P}[\mathcal{R}^* c'(u) = \mathcal{R}^* c'(v)] \leq 0.3^{2\log n} = o(n^{-2})$. The desired result follows from a union bound over pairs u, v. ■

Now, suppose that the conclusion of Claim 6.9 holds (from now on we no longer view c' as random), and suppose for the purpose of contradiction that some vertex u has at least $4\log n$ neighbours in $D(H, \mathcal{R}^* c')$. Let $N_{D(H,\mathcal{R}^* c')}(u)$ be this set of neighbours, and let C' be the colour class of u in $D(H, \mathcal{R}^* c')$. Observe that we must have $|C'| \geq 2$, because colour classes of size 1 correspond to isolated vertices in $D(H, \mathcal{R}^* c')$. Now, by Fact 6.1, every vertex is adjacent to at most half the vertices in C', so if v is a uniformly *random* vertex in C' (this is now the only source of randomness, as we are viewing c' as fixed), then for every vertex $x \in V(H)$, we have $\mathbb{P}[vx \text{ is an edge of } D(H, \mathcal{R}^* c')] \leq 1/2$. This means that $\mathbb{E}[|N_{D(H,\mathcal{R}^* c')}(u) \cap N_{D(H,\mathcal{R}^* c')}(v)|] \leq |N_{D(H,\mathcal{R}^* c')}(u)|/2$, so there must be some specific vertex $v \in C'$ with $|N_{D(H,\mathcal{R}^* c')}(u) \cap N_{D(H,\mathcal{R}^* c')}(v)| \leq |N_{D(H,\mathcal{R}^* c')}(u)|/2$. That is to say, with respect to $D(H, \mathcal{R}^* c')$, the symmetric difference of neighbourhoods of u and v is at least $|N_{D(H,\mathcal{R}^* c')}(u)|/2 \geq 2\log n$, and since u, v lie in the same colour class C', the same is true with respect to H. This contradicts the conclusion of Claim 6.9. □

6.3. **Splitting connected components via colour classes.** Our next key lemma is that in order to control the sizes of the connected components of the disparity graph (after sprinkling), it suffices to control the number of vertices of a single colour in a single component. We will prove this with an adaptive algorithm to explore a connected component, revealing randomness as we go. Along the way we also prove that sprinkling typically ensures that large components can be partitioned into full colour classes.

Lemma 6.10. *Consider an n-vertex graph H and an equitable colouring $c : V(H) \to \Omega$. Let c' be a random refinement of c, obtained by assigning each vertex a new unique colour with probability at least* 0.7*, independently. Then, whp c' satisfies the following properties.*

1. *Every connected component of $D(H, c)$ with at least $\log n$ vertices can be partitioned into colour classes of $\mathcal{R}^* c'$.*
2. *For every colour class C of c, and every connected component[19] X of $D(H, c)$, every connected component of $D(H, \mathcal{R}^* c')$ that intersects $C \cap X$ has at most $5|C \cap X| + 5\log n$ vertices.*

Proof. Fix a colour class C (of c) and a connected component X (of $D(H, c)$), such that $C \cap X$ is nonempty (there are at most n such choices). Suppose that $|X| \geq \log n$ (noting that (2) follows deterministically by Lemma 6.3 when $|X| < \log n$). We will show that with probability $1 - o(1/n)$,

(1) $C \cap X$ can be partitioned into colour classes of $\mathcal{R}^* c'$, and
(2) every connected component of $D(H, \mathcal{R}^* c')$ that intersects $C \cap X$ has at most $5|C \cap X| + 5\log n$ vertices.

The desired result will follow by a union bound over all nonempty choices of $C \cap X$.

Say that a vertex $v \in U$ is *special* if it is assigned a new unique colour by c'. We reveal which vertices are special in an adaptive fashion, according to the following 2-phase algorithm.

- **Phase 1:** Consider an arbitrary subset $Y \subseteq X$ of size $\log n$, and reveal which of the vertices in $(C \cap X) \cup Y$ are special.
  - If none of these vertices of X are special, abort the whole algorithm.
- **Phase 2:** Let $t = 0$, and repeatedly do the following.
  (a) Let $E_t \supseteq C \cap X$ be the set of vertices whose specialness has been revealed so far, let $S_t \subseteq E_t$ be the set of vertices revealed to be special so far, and let c_t be the colouring obtained from c by assigning each vertex in S_t a new unique colour (i.e., c_t represents the information about c' that we know so far).

[19] Here, and in the rest of the section, we conflate components with their vertex sets (i.e., here X is really a vertex set on which a connected component lies).

(b) For each vertex $v \notin E_t$, let c_t^v be the colouring obtained from c_t by assigning v a new unique colour (i.e., c_t^v is the hypothetical colouring we would have after this step, if we revealed v to be special).

(c) If there is some $v \notin E_t$ such that the number of colour classes of $\mathcal{R}^* c_t^v$ intersecting E_t exceeds the number of colour classes of $\mathcal{R}^* c_t$ intersecting E_t, then choose such a v arbitrarily, and reveal whether it is special.
   - If there is no such v, then set $\tau = t$ and abort the algorithm.

(d) Increment t (i.e., set $t = t + 1$).

To briefly summarise: in Phase 1, we ensure that there is a vertex in X with a unique colour (this will imply (1), and is also an ingredient in the proof of (2)), then in Phase 2 we keep revealing whether vertices are special, as long as their specialness would create new colour classes among the previously revealed vertices.

First, note that

$$\mathbb{P}[\text{the algorithm aborts in Phase 1}] \leq 0.3^{\log n} = o(1/n).$$

For the rest of the proof, we assume that Phase 1 has completed without aborting (all probability calculations should be interpreted as being conditional on the information revealed by this phase).

For all $t \leq \tau$ let N_t be the number of vertices in E_t, minus the number of colour classes of $\mathcal{R}^* c_t$ intersecting E_t. Note that $N_0 \leq |C \cap X| + \log n$. Each time when step (c) of Phase 2 occurs (for a vertex v), either v is special, in which case N_t decreases by at least 1, or v is not special, in which case N_t increases by 1. Since vertices are more likely than not to be special (each vertex is special with probability 0.7), this process typically does not continue for long, as follows.

Claim 6.11. *We have $|E_\tau| \leq 5|C \cap X| + 5 \log n$ with probability $1 - o(1/n)$.*

Proof of claim. Artificially extend the definition of N_t for $t > \tau$, by taking $N_{t+1} = N_{t+1} + 1$ with probability 0.3 and $N_{t+1} = N_{t+1} - 1$ with probability 0.7. For $s = |C \cap X| + \log n$ we have $\mathbb{E}[N_{4s}] \leq s + (0.3 - 0.7)4s = -0.6s$. Also note that $E_t = t + s$. Thus $|E_\tau| > 5s$ implies that $\tau > 4s$. In particular, not every vertex in E_{4s} lies in a distinct colour class of $\mathcal{R}^* c_{4s}$ (this event would had cause Phase 2 to terminate- see step (c)), meaning that $N_{4s} > 0$. So

$$\mathbb{P}\big[|E_\tau| > 5|C \cap X| + 5 \log n\big] \leq \mathbb{P}[N_{4s} > 0] \leq \mathbb{P}[N_{4s} - \mathbb{E}N_s > 0.6s] = o(1/n)$$

by a Chernoff bound (Lemma 2.26). ■

Now, let W be the union of all the colour classes of $\mathcal{R}^* c_\tau$ which are entirely contained in E_τ. The following two claims essentially complete the proof of Lemma 6.10.

Claim 6.12. *$C \cap X$ can be partitioned into colour classes of $\mathcal{R}^* c_\tau$, so $C \cap X \subseteq W$.*

Proof of claim. This claim is where we use the fact that Phase 1 of the algorithm did not abort: let v be a special vertex in X, so v has a unique colour with respect to $\mathcal{R}^* c_\tau$.

Since $\mathcal{R}^* c_\tau$ is a refinement of c, we can interpret $D(G, c)$ as a generalised disparity graph of the form $D_L(G, \mathcal{R}^* c_\tau)$. Note that there is a path in $D(G, c)$ between a vertex of colour $\mathcal{R}^* c_\tau(v)$ and any vertex in $C \cap X$, but there is no such path between a vertex of colour $\mathcal{R}^* c_\tau(v)$ and any vertex in $C \setminus X$ (recalling that v is the unique vertex with colour $\mathcal{R}^* c_\tau(v)$). So, by Fact 2.19, the vertices in $C \cap X$ and the vertices in $C \setminus X$ are assigned disjoint sets of colours by $\mathcal{R}^* c_\tau$. ■

Claim 6.13. *W can be partitioned into connected components of $D(H, \mathcal{R}^* c')$.*

Proof of claim. We need to prove that there is no edge between W and $V(H) \setminus W$ in $D(H, \mathcal{R}^* c')$. Since $\mathcal{R}^* c'$ is a refinement of $\mathcal{R}^* c_\tau$, and W can be partitioned into colour classes of c_τ, it suffices to prove that there is no edge between W and $V(H) \setminus W$ in $D(H, \mathcal{R}^* c_\tau)$. For the rest of this proof, all graph-theoretic language is with respect to $D(H, \mathcal{R}^* c_\tau)$, and all colour classes are with respect to c_τ.

Suppose for the purpose of contradiction that there is an edge between a colour class $Z \subseteq E_\tau$ and a colour class $Y \not\subseteq E_\tau$, and consider any $v \in Y \setminus E_\tau$. By Fact 6.1, the number of neighbours of v in Z lies in the range $[1, |Z|/2]$; in particular, v has both neighbours and non-neighbours in Z. Recall that c_τ^v is the colouring obtained from c_τ by assigning v a new unique colour, and note that $\mathcal{R}^* c_\tau^v$ assigns disjoint sets of colours to the neighbours of v and the non-neighbours of v. So, the number of colour classes of $\mathcal{R}^* c_\tau^v$ intersecting E_τ exceeds the number of colour classes of $\mathcal{R}^* c_\tau$ intersecting E_τ, contradicting the definition of τ (see Phase 2(c)). ■

Claim 6.12 immediately implies (1). Then, Claims 6.12 and 6.13 imply that every connected component of $D(H, \mathcal{R}^*c')$ intersecting $C \cap X$ is contained in E_τ. By Claim 6.11, with probability $1 - o(1/n)$ every such component X has at most $5|X \cap C| + 5\log n$ vertices, proving (2). □

6.4. **Splitting colour classes via distance information.** Given Lemma 6.10, our goal is now to show that sprinkling tends to break up intersections of the form $C \cap X$ (where C is a colour class and X is a connected component of the disparity graph) into much smaller parts. We prove a general-purpose lemma that accomplishes this, which will be applied several times (each time with different parameters) in the proof of Theorem 1.3.

Definition 6.14. For a graph H and a colouring $c : V(H) \to \Omega$, say that $D(H, c)$ is *s-bounded* if for every colour class C of c, and every connected component X of $D(H, c)$, we have $|C \cap X| \le s$.

Lemma 6.15. *Consider integers s, s', Δ such that $s \ge s' \ge 100\log n$ and $s'/(8\Delta) \ge 10\log s$. Consider an n-vertex graph H and an equitable colouring $c : V(H) \to \Omega$, such that $D(H, c)$ is s-bounded and has maximum degree at most Δ. Let c' be a random refinement of c, obtained by assigning each vertex a new unique colour with probability at least 0.7, independently. Then whp $D(H, \mathcal{V}^*c')$ is s'-bounded.*

Recalling Lemma 6.8 (which shows that the maximum degree of the disparity graph is at most $O(\log n)$), we can apply Lemma 6.15 with $s = n$, $s' = O((\log n)^2)$ and $\Delta = O(\log n)$ to establish $O((\log n)^2)$-boundedness. Actually, this would already be enough to prove Theorem 1.3 if we were willing to use group-theoretic canonical labelling schemes on small components of the disparity graph, but for a combinatorial algorithm we need to go down to $O(\log n)$-boundedness. For this we will need some additional control over degrees in the disparity graph, via the following somewhat delicate lemma.

Definition 6.16. For a graph H and a colouring $c : V(H) \to \Omega$, say that $D(H, c)$ is *(r, d)-degree-bounded* if for every colour class C of c which intersects some component of $D(H, c)$ in at least r vertices, every vertex $y \in V(H)$ has at most d neighbours in C, with respect to $D(H, c)$.

Lemma 6.17. *Consider integers s, s' such that $s \ge s' \ge 100\log n$. Consider an n-vertex graph H and a colouring $c : V(H) \to \Omega$, such that $D(H, c)$ is s-bounded and $D(H, c)$ has maximum degree less than $s'/8$. Let c' be a random refinement of c, obtained by assigning each vertex a new unique colour with probability at least 0.7, independently. Then whp $D(H, \mathcal{V}^*c')$ is $(s', 10\log s)$-degree-bounded.*

Both Lemmas 6.15 and 6.17 are proved with the same "fingerprint" idea (briefly sketched in Section 1.3.9). In fact, we will be able to deduce both of Lemmas 6.15 and 6.17 from the following technical lemma.

Lemma 6.18. *Consider integers s, s' such that $s \ge s' \ge 100\log n$. Consider an n-vertex graph H and a colouring $c : V(H) \to \Omega$, such that $D(H, c)$ is s-bounded. Let c' be a random refinement of c, obtained by assigning each vertex a new unique colour with probability at least 0.7, independently. Then, the following property holds whp.*

For a set of vertices S, a vertex y and a subset $\mathcal{I} \subseteq \mathcal{N} \cup \{\infty\}$, let $Q_{\mathcal{I}}(S, y)$ be the number of vertices in S whose distance to y, with respect to $D(H, c)$, lies in $\mathcal{I}$. Then, for every vertex $y \in V(G)$, every $\mathcal{I} \subseteq \mathbb{N} \cup \{\infty\}$ and every colour class C' of $\mathcal{V}^*c'$ which intersects some component of $D(H, \mathcal{V}^*c')$ in at least s' vertices, we have $Q_{\mathcal{I}}(C', y) \notin [10\log s, s'/8)$.

Before proving Lemma 6.18, we deduce Lemmas 6.15 and 6.17.

Proof of Lemma 6.15. Suppose that the conclusions of Lemma 6.10(1) and Lemma 6.18 both hold, and suppose for the purpose of contradiction that there is a colour class C' of $\mathcal{V}^*c'$ and connected component X' of $D(H, \mathcal{V}^*c')$ such that $|C' \cap X'| \ge s'$.

By Lemma 6.3, there is a connected component X of $D(H, c)$ which intersects C' in at least $s'/2 \ge 50\log n$ vertices. By the conclusion of Lemma 6.10(1), we must have $X \supseteq C'$. By Lemma 6.6, with $F = D(H, c)[X]$, there is a vertex $y \in X$ and a set $\mathcal{I}$ such that $Q_{\mathcal{I}}(C', y) \in [s'/(8\Delta), s'/8)$. Since we are assuming $s'/(8\Delta) \ge 10\log s$, this contradicts the statement of Lemma 6.18. □

Proof of Lemma 6.17. Suppose that the conclusion of Lemma 6.18 holds, and suppose for the purpose of contradiction that there were some vertex y and colour class C' of $\mathcal{V}^*c'$ which intersects some component of $D(H, \mathcal{V}^*c')$ in at least s' vertices, such that y has more than $10\log s$ neighbours in C' with respect to $D(H, \mathcal{V}^*c')$. Note that this number of neighbours is less than $s'/8$, by our assumption on the maximum degree of $D(H, c)$ and Fact 6.2.

But taking $\mathcal{I} = \{1\}$ or $\mathcal{I} = (\mathbb{N} \cup \{\infty\}) \setminus \{1\}$, and recalling the definitions of the disparity graphs $D(H, c)$ and $D(H, \mathcal{V}^*c')$, we can write the above number of neighbours as $Q_{\mathcal{I}}(C', y)$, contradicting Lemma 6.18. □

Now we prove Lemma 6.18. The idea is that if there were C', y satisfying $Q_{\mathcal{I}}(C', y) \in [10 \log s, s'/4)$, then every other vertex v in the colour class B' of y (with respect to $\mathcal{V}^*c'$) would also have $Q_{\mathcal{I}}(C', v) \in [10 \log s, s'/4)$, due to the fact that the 2-dimensional Weisfeiler–Leman algorithm can detect distances in $D(H, c)$ (which can be interpreted as a generalised disparity graph $D_L(H, \mathcal{V}^*c)$). We would then be able to define an auxiliary bipartite graph and use Lemma 6.7 to find a relatively small "fingerprint" set of vertices in B' which have a wide variety of distances to the vertices in C' (with respect to $D(H, c)$). But this situation can be ruled out whp with a union bound: if, in $D(H, c)$, a set of vertices S has a wide variety of distances to many other vertices, then it is very unlikely that that the vertices in S are all given the same colour by $\mathcal{V}^*c'$.

Proof of Lemma 6.18. First, we can use the union bound to show that whp certain sets of vertices (with a rich variety of distances to other vertices) are not monochromatic with respect to $\mathcal{V}^*c'$. For a set of vertices S we denote by $\operatorname{Diff}(S)$ the set of all vertices in $D(G, c)$ which do not see all vertices of S at the same distance.

Claim 6.19. *Whp the following holds. For every colour class B of c and connected component X of $D(H, c)$, and every subset $S \subseteq B \cap X$ satisfying*

$$\binom{|B \cap X|}{|S|} 0.3^{|\operatorname{Diff}(S)|} \leq \frac{1}{n^2}, \tag{6.2}$$

S is not contained inside a single colour class of $\mathcal{V}^*c'$.

Proof of claim. For a particular set S, if any of the vertices in $\operatorname{Diff}(S)$ are assigned a unique colour by c' then S does not lie inside a single colour class of $\mathcal{V}^*c$. This follows from Fact 2.23, noting that we can interpret $D(H, c)$ as a generalised disparity graph $D_L(H, \mathcal{V}^*c')$. So, the probability that a particular set S is contained inside a single colour class of $\mathcal{V}^*c'$ is at most $0.3^{|\operatorname{Diff}(S)|}$. The desired result then follows from a union bound, noting that there are at most n nonempty subsets of the form $|B \cap X|$. ■

Suppose that the conclusions of Claim 6.19 and Lemma 6.10(1) both hold, and suppose for the purpose of contradiction that there is a colour class C' of $\mathcal{V}^*c'$, connected component X' of $D(H, \mathcal{V}^*c')$, vertex y and set $\mathcal{I}$ such that $|C' \cap X'| \geq s'$ and $Q_{\mathcal{I}}(C', y) \in [10 \log s, s'/8)$.

Let B' be the colour class containing y with respect to $\mathcal{V}^*c'$ (we may or may not have $B' = C'$), and let F be the auxiliary bipartite graph with parts C' and B', with an edge between $u \in C'$ and $v \in B'$ if the distance between u and v in $D(H, c)$ lies in $\mathcal{I}$. We can interpret $D(H, c)$ as a generalised disparity graph $D_L(H, \mathcal{V}^*c')$, so by Fact 2.23, F is a biregular bipartite graph. In this graph the degree of each vertex in B' is exactly $Q_{\mathcal{I}}(C', y)$. By Lemma 6.7, there is a subset $S \subseteq B'$ of size at most $|C'|/(2Q_{\mathcal{I}}(C', y)) + 1 \leq |C'|/(19 \log s)$ such that $|N_F(S)| \geq |C'|/4 \geq s'/4 \geq 2Q_{\mathcal{I}}(C', y)$ (for both these inequalities we used our assumption $Q_{\mathcal{I}}(C', y) \in [10 \log s, s'/8)$).

Now, fix a vertex $v \in S$ and note that $\operatorname{Diff}(S) \supseteq N_F(S) \setminus N_F(v)$ (because every vertex in $N_F(S) \setminus N_F(v)$ sees some vertex in S at a distance in $\mathcal{I}$, but does not see v at a distance in $\mathcal{I}$). So, $|\operatorname{Diff}(S)| \geq |N_F(S)| - Q_{\mathcal{I}}(C', y) \geq |C'|/8$.

Then (as in the deduction of Lemma 6.15), Lemma 6.3 tells us that there is a connected component X of $D(H, c)$ intersecting C' in at least $s/2$ vertices, and by the conclusion of Lemma 6.10(1), we have $X \supseteq C'$. We must also have $B' \subseteq X$ (if there were a vertex in B' lying in a different component to the vertices of C', we would have $Q_{\mathcal{I}}(C', y) \in \{0, |C'|\}$, which contradicts our assumption $Q_{\mathcal{I}}(C', y) \in [10 \log s, s'/8)$). Letting B be the colour class of c containing B', recalling that $|C'| \geq s' \geq 100 \log n$, and recalling that $D(H, c)$ is s-bounded, we compute

$$\binom{|B \cap X|}{|S|} 0.3^{|\operatorname{Diff}(S)|} \leq |B \cap X|^{|S|} 0.3^{|C'|/8} \leq s^{|C'|/(19 \log s)} 0.3^{|C'|/8} \leq \frac{1}{n^2}.$$

But S is a subset of the colour class B' of $\mathcal{V}^*c'$, contradicting the conclusion of Claim 6.19. □

6.5. Putting everything together.

We now prove Theorem 1.3, combining all the ingredients collected so far. By Corollary 2.7 and Fact 2.22, it suffices to prove the following theorem.

Theorem 6.20. *Consider any $p \in [0, 1/2]$ satisfying $p \geq 100/n$, consider any graph G_0 on the vertex set $\{1, \ldots, n\}$, and let $G_{\text{rand}} \sim \mathbb{G}(n, p)$. Then whp every connected component of $D(G_0 \triangle G_{\text{rand}}, \mathcal{V}^*\phi_G)$ has $O(\log n)$ vertices.*

Remark 6.21. Our proof approach shows that the implicit constant in "$O(\log n)$" can be made as small as desired, by increasing the "100" in the assumption on p.

Proof. Let $G = G_0 \triangle G_{\mathrm{rand}}$. First note that we can assume $p \le 2\log n/n$, as otherwise the desired result follows from Theorem 1.2 (note that if $\mathcal{R}^*\sigma$ assigns all vertices distinct colours, then so does $\mathcal{V}^*\phi_G$, by Fact 2.24, and $D(G, \mathcal{V}^*\sigma)$ consists only of isolated vertices. As outlined, we view the random perturbation G_{rand} as the union of eight slightly sparser random perturbations $G^i_{\mathrm{rand}} \sim \mathbb{G}(n, p')$, for $i = 1, ..., 8$, where p' is chosen such that $1 - p = (1 - p')^8$. Since we are assuming $p \ge 100/n$, we have $p' \ge 10/n$.

Since $p' \ge 10/n$, standard results show that $|V_3(G^1_{\mathrm{rand}})| \ge n/2$ whp (see e.g. [70]). So, whp we can fix a subset $V_{\mathrm{core}} \subseteq V_3(G^1_{\mathrm{rand}})$ with size $n/2$. Let $U = V(G) \setminus V_{\mathrm{core}}$, and for each $i \in \{2, \dots, 8\}$, let S_i be the set of vertices in U which have at least three neighbours in V_{core} with respect to G^i_{rand}. We use the sets S_i, together with the 2-dimensional Weisfeiler–Leman algorithm, to recursively define a sequence of vertex-colourings $c_1, \dots, c_8$ of U, as follows.

- Let $c_1 = \mathcal{V}^*\phi_{G[U]}$ (this is a colouring of vertices in U)
- For $i \in \{2, \dots, 8\}$:
  - let b_i be the vertex-colouring obtained from c_{i-1} by giving all vertices in S_i a unique colour, and
  - let $c_i = \mathcal{V}^*\phi_{G[U],b_i}$ (recalling from Definition 2.20 that $\phi_{G[U],b_i}$ is the colouring of pairs of vertices of $G[U]$, obtained by augmenting the "trivial" colouring $\phi_{G[U]}$ with the vertex-colouring b_i).

The plan is to first reveal $G[U]$ and c_1, and then reveal $c_2, \dots, c_8$ one-by-one, studying how the components, colour classes and degrees change along the way (using Lemmas 6.8, 6.10, 6.15, and 6.17). The following claim justifies this plan.

Claim 6.22. *Whp the largest connected component in $D(G, \mathcal{V}^*\phi_G)$ has at most twice as many vertices as the largest connected component in $D(G[U], c_8)$.*

Sketch proof of claim. By Proposition 5.5, whp $\mathcal{R}^*\sigma$ assigns each vertex in the 3-core $V_3(G_{\mathrm{rand}}) \supseteq V_{\mathrm{core}} \cup S_2 \cup \dots \cup S_8$ a unique colour, and by Fact 2.24 the same is true for $\mathcal{V}^*\phi_G$. We will see that the desired conclusion holds whenever this is the case.

Extend each c_i to a colouring c_i^{ext} of $V(G)$, by assigning each vertex in V_{core} a unique colour. Note that $D(G, c_i^{\mathrm{ext}})$ has the same components as $D(G[U], c_i)$ (plus singleton components for each vertex of V_{core}), so by Lemma 6.3 it suffices to show that $\mathcal{V}^*\phi_G$ is a refinement of each c_i^{ext}. This can be proved inductively, with eight applications of Fact 2.25 (note that we are assuming $\mathcal{V}^*\phi_G$ assigns each vertex a unique colour, so $G[U]$ can be partitioned into full colour classes). ■

For the rest of the proof, we reveal all the edges in G^1_{rand} (which determines U), and reveal all the edges in $G[U]$. We assume that $|V_3(G^1_{\mathrm{rand}})| \ge n/2$ (so V_{core} and U are well-defined).

Given Claim 6.22, our objective is to use the remaining randomness of $G^2_{\mathrm{rand}}, \dots, G^8_{\mathrm{rand}}$ (via the colourings $c_2, \dots, c_8$) to prove that whp every component of $D(G[U], c_8)$ has at most $O(\log n)$ vertices.

First, note that the random sets S_i are independent. For each $i \in \{2, \dots, 8\}$, each vertex $v \in U$ is independently present in S_i with probability

$$\mathbb{P}[\mathrm{Binomial}(n/2, 10/n) \ge 3] = 1 - \frac{37}{2e^5} + o(1) \ge 0.7.$$

We now track how $D(G[U], c_i)$ evolves with i.

Round 1. Applying Lemma 6.8 with $H = G[U]$, $c = c_1$ and $c' = c_2$, we see that whp $D(G[U], c_2)$ has maximum degree at most $4\log n$. Reveal an outcome of c_2 such that this is the case (i.e., for the rest of the proof, all probabilities should be interpreted as being conditional on an outcome of c_2 with this property).

Round 2. Applying Lemma 6.15 with $H = G[U]$, $c = c_2$, $c' = c_3$, $\Delta = 4\log n$, $s = n$ and $s' = 10^4(\log n)^2$, we see that whp $D(G[U], c_3)$ is $10^4(\log n)^2$-bounded. Reveal an outcome of c_3 such that this is the case.

Round 3. By Fact 6.2, the maximum degree of $\Delta(D(H, c_3))$ is at most the maximum degree of $\Delta(D(H, c_2))$, which is at most $4\log n$. Applying Lemma 6.15 again, with $H = G[U]$, $c = c_3$, $c' = c_4$, $\Delta = 4\log n$, $s = 10^4(\log n)^2$ and $s' = 10^4 \log n \log\log n$, we see that whp $D(G[U], c_4)$ is $10^4 \log n \log\log n$-bounded. Reveal an outcome of c_4 such that this is the case.

Round 4. By Lemma 6.10(2) with $H = G[U]$, $c = c_4$ and $c' = c_5$, using the $10^4 \log n \log\log n$-boundedness we have just established in the previous round, whp every connected component of $D(G[U], c_5)$

has at most $10^5 \log n \log\log n$ vertices. Also, applying Lemma 6.17 with $H = G[U]$, $c = c_4$, $c' = c_5$, $s = 10^4 \log n \log\log n$ and $s' = 100 \log n$ (noting that s' is at least 8 times the maximum degree of $D(H, c_4)$, by Fact 6.2), we see that whp $D(G[U], c_5)$ is $(100 \log n, 11 \log\log n)$-degree-bounded. Reveal an outcome of c_5 such that both of these properties hold.

Let $U^{\mathrm{big}} \subseteq U$ be the union of all sets of the form $C \cap X$ with size at least $100 \log n$, where C is a colour class of c_5 and X is a component of $D(G[U], c_5)$ (say such a set $C \cap X$ is a "big intersection set"). Every connected component of $D(G[U], c_5)$ has at most $10^5 \log n \log\log n$ vertices, so it contains at most $10^3 \log\log n$ big intersection sets. By degree-boundedness, every vertex has at most $11 \log\log n$ neighbours (with respect to $D(G[U], c_5)$) in each big intersection set, so the maximum degree of $D(G[U], c_5)[U^{\mathrm{big}}]$ is at most $11 \log\log n \cdot 10^3 \log\log n \le 10^5 (\log\log n)^2$.

Round 5. By Lemma 6.10(1) with $H = G[U]$, $c = c_5$ and $c' = c_6$, whp U^{big} can be partitioned into colour classes of c_6. Also, by Lemma 6.10(2) (with the same H, c, c'), whp $U \setminus U^{\mathrm{big}}$ can be covered by connected components of $D(H, c_6)$, each with at most $10^6 \log n$ vertices. Reveal an outcome of c_6 such that both these properties hold.

For $i \in \{5, 6, 7\}$, let c_i^{big} be the restriction of c_i to U^{big}, so $D(G[U], c_5)[U^{\mathrm{big}}]$ can be interpreted as a generalised disparity graph $D_L(G[U], c_5^{\mathrm{big}})$. Then $D(G[U^{\mathrm{big}}], c_6^{\mathrm{big}})$ has maximum degree at most $10^5 (\log\log n)^2$ by Fact 6.2, and its components have at most $2 \cdot 10^5 \log n \log\log n$ vertices, by Lemma 6.3.

Round 6. We apply Lemma 6.15 yet again, this time with $H = G[U^{\mathrm{big}}]$, $c = c_6^{\mathrm{big}}$, $c' = c_7^{\mathrm{big}}$, $\Delta = 10^5 (\log\log n)^2$, $s = 2 \cdot 10^5 \log n \log\log n$ and $s' = 100 \log n$, to see that whp $D(G[U^{\mathrm{big}}], c_7^{\mathrm{big}})$ is $100 \log n$-bounded. Reveal an outcome of c_7 such that this is the case.

Recalling that $U \setminus U^{\mathrm{big}}$ can be covered by connected components of $D(H, c_6)$ with at most $100 \log n$ vertices, Lemma 6.3 tells us that in fact the whole of $D(G[U], c_7)$ is $2 \cdot 10^5 \log n$-bounded.

Round 7. Finally, apply Lemma 6.10 with H, $c = c_7$ and $c' = c_8$, to see that whp all connected components of $D(H, c_8)$ have at most $O(\log n)$ vertices, as desired. □

7. Near-critical random graphs

In this section we prove that for random graphs of any density, colour refinement whp assigns distinct colours to each vertex of degree at least 3 in the 2-core. By Corollary 2.14 and Remark 2.12, this implies Theorems 1.4 and 1.8.

Theorem 7.1. *For any sequence $(p_n)_{n \in \mathbb{N}} \in [0,1]^{\mathbb{N}}$, and $G \sim \mathbb{G}(n, p_n)$, whp every two distinct vertices $u, v \in V_{2,3}(G)$ are assigned different colours in the stable colouring $\mathcal{R}^*\sigma$.*

Theorem 7.1 follows directly from Proposition 5.2 (with assumption **A2**) in the case $p \ge (1 + (\log n)^{-40})/n$, and it is vacuous for $p \le (1 - \varepsilon)/n$, for any constant $\varepsilon > 0$, as in this regime $V_{2,3}(G)$ is empty whp (see for example [36, Lemma 2.10]). So, we focus on the near-critical regime $0.9/n \le p \le (1 + (\log n)^{-2})/n$.

In this near-critical regime we cannot apply the machinery developed in Propositions 4.1 and 5.2 Indeed, this machinery fundamentally relies on expansion properties only available for supercritical random graphs, and critical random graphs typically have poor expansion properties (in particular, most pairs of vertices are quite far from each other). But this gives us another way to proceed: if two vertices u, v are far from each other (or, more generally, if for reasonably large i we can find vertices which have distance i from u while being at distance greater than i from v), then if we consider exploration processes starting from u and v, we can run these processes for quite a long time while continuing to maintain some independent randomness between the two processes. Although this a rather small amount of randomness per step (compared to Proposition 4.1), we can accumulate this for many steps, to see that there is very likely to be some deviation between the degree statistics of these processes. This will allow us to conclude with Lemma 3.5.

The following lemma can be viewed as an analogue of Proposition 4.1, which will be used to execute the above plan. (It is much easier to prove than Proposition 4.1).

Lemma 7.2. *Let $G \sim \mathbb{G}(n, p)$ for some p satisfying $0.9/n \le p \le (1 + (\log n)^{-40})/n$. Then whp G satisfies the following property. For every pair of vertices u, v such that $\mathcal{R}^*\sigma(u) = \mathcal{R}^*\sigma(v)$, we have $\mathcal{S}^i(\{u, v\}) = \emptyset$ for some $i \le 10 \log n$.*

(Recall that the sets $\mathcal{S}^i(\{u, v\})$, defined in Definition 3.4, describe the "vertices that appear differently in the i-th step of the view exploration process").

For our proof of Lemma 7.2 we will need a simple lemma showing that in very sparse random graphs, every vertex has distance greater than $10\log n$ from most other vertices (therefore throughout the first $10\log n$ steps of an exploration process, there are always many vertices that have not yet been explored).

Lemma 7.3. *Let $G \sim \mathbb{G}(n,p)$ for some p satisfying $p \le (1+1/\log n)/n$. Whp, G does not have any vertex v which is within distance $10\log n$ of at least $n/4$ different vertices.*

Proof. For a vertex v, let $N^i(v)$ be the set of vertices at distance i from v. We claim that, whp, for all vertices v, and all $i \le 10\log n$, we have

$$|N^i(v)| \le \max\Big\{n^{0.1},\ (1+5/\log n)|N^{i-1}(v)|\Big\}. \tag{7.1}$$

Indeed, for any fixed v, imagine an exploration process where we iteratively reveal the sets $N^i(v)$ (at step i we reveal all edges incident to vertices in $N^{i-1}(v)$). For fixed i, if we condition on any outcome of $N^{i-1}(v)$, then $|N^i(v)|$ is at most the number of edges between $N^{i-1}(v)$ and previously unrevealed vertices, which is stochastically dominated by the binomial distribution $\operatorname{Binomial}\big(n|N^{i-1}(v)|,p\big)$. By a Chernoff bound (Lemma 2.26), (7.1) holds with probability at least $1-o(1/n^2)$, so the result holds by a union bound over all vertices v, and positive integers i.

Then, note that if (7.1) holds for each $i \le 10\log n$, the total number of vertices within distance $10\log n$ of v is at most $\sum_{i=0}^{10\log n}(1+5/\log n)^i n^{0.1} \le (10\log n)(1+5/\log n)^{10\log n}n^{0.1} \le n/4$. □

Now we prove Lemma 7.2.

Proof of Lemma 7.2. Fix $v,u \in [n]$. We consider the exploration process defined in the proof of Proposition 4.1 (also used in the proofs of Proposition 5.2 and Lemma 4.2), to iteratively reveal the sets $\mathcal{S}^{i-1}(\{u,v\})$.

Fix vertices u,v and let $i \le 10\log n$. Suppose we have so far revealed $\mathcal{S}^1(\{u,v\}),\dots,\mathcal{S}^{i-1}(\{u,v\})$, suppose that $\mathcal{S}^{i-1}(\{u,v\}) \ne \emptyset$ and suppose that $|\mathcal{S}^{\le i-1}(\{u,v\})| \le n/2$. For vertices w,x, let ξ_{wx} be the indicator random variable for the event that wx is an edge of G, and let a_x be the number of times x appears in $\mathcal{L}^{i-1}(u,v)$, minus the number of times it appears in $\mathcal{L}^{i-1}(v,u)$. So,

$$|\mathcal{L}^i(u,v)| - |\mathcal{L}^i(v,u)| = \sum_{w,x\in V(G)} a_x\xi_{wx}.$$

Some of the ξ_{wx} may have already been revealed. In particular, there are

$$\big(n - |\mathcal{S}^{\le i-1}(\{u,v\})|\big)\cdot|\mathcal{S}^{i-1}(\{u,v\})| \ge n/2$$

pairs of vertices $w \notin \mathcal{S}^{\le i-1}(\{u,v\})$ and $x \in \mathcal{S}^{i-1}(\{u,v\})$ for which $a_x \ne 0$ and ξ_{wx} has not yet been revealed. Reveal ξ_{wx} for all but $n/2$ of these pairs, putting us in a position to apply Theorem 2.28. Recall the definition of $M(\lfloor n/4\rfloor,p)$ from Definition 2.27; conditional on information revealed so far, we see that the probability of the event $|\mathcal{L}^i(u,v)| - |\mathcal{L}^i(v,u)| = 0$ is at most

$$M(\lfloor n/4\rfloor,p) = (1-p)^{\lfloor n/4\rfloor} \le e^{-0.9/4} + o(1) \le 0.8$$

(recalling that $p \ge 0.9/n$). Recall from Lemma 3.5 that if $|\mathcal{L}^i(u,v)| \ne |\mathcal{L}^i(v,u)|$ then $\mathcal{R}^*\sigma(u) \ne \mathcal{R}^*\sigma(v)$. So, for our fixed u,v, the joint event that

$$\mathcal{S}^{i-1}(\{u,v\}) \ne \emptyset \text{ and } |\mathcal{S}^{\le i-1}(\{u,v\})| \le n/2 \text{ for all } i \le 10\log n, \quad \text{and } \mathcal{R}^*\sigma(u) = \mathcal{R}^*\sigma(v)$$

holds with probability at most $0.8^{10\log n} = o(n^{-2})$. Taking a union bound over choices of u,v, we see that whp the above event does not hold for any u,v. To finish the proof, observe that the property in Lemma 7.3 implies that $|\mathcal{S}^{\le i-1}(\{u,v\})| \le n/2$ for all u,v and all $i \le 10\log n$. □

We need two final ingredients before completing the proof of Theorem 7.1: in very sparse random graphs, whp there are no short cycles in close proximity to each other, and this implies a certain structural fact about vertices in $V_{2,3}(G)$.

Lemma 7.4. *Let $G \sim G(n,p)$ for some p satisfying $p \le (1+(\log n)^{-2})/n$. Then whp G does not have any connected subgraph with at most $200\log n$ vertices and at least two cycles.*

Proof. An edge-minimal connected subgraph containing two cycles is always either:

- a pair of vertex-disjoint cycles joined by a (possibly trivial) path, or
- two vertices with three internally disjoint paths between them.

The probability that G contains two vertex-disjoint cycles of lengths k_1, k_2, joined by a path of length $k_3 \geq 0$ (i.e., a path with k_3 edges) is at most

$$n^{k_1+k_2+k_3-1}p^{k_1+k_2+k_3} = n^{-1}(np)^{k_1+k_2+k_3} \leq \frac{1}{n}\left(1+\frac{1}{(\log n)^2}\right)^{k_1+k_2+k_3}.$$

The number of vertices in such a configuration is $k_1 + k_2 + k_3 - 1$, so the probability that such a configuration exists on at most $200 \log n$ vertices is at most

$$\frac{(200 \log n)^3}{n}\left(1+\frac{1}{(\log n)^2}\right)^{200\log n+1} = o(1).$$

Similarly, the probability that G contains two vertices joined by three internally disjoint paths of lengths $k_1, k_2, k_3 \geq 1$ is at most

$$n^{k_1+k_2+k_3-1}p^{k_1+k_2+k_3},$$

so essentially the same calculation shows that whp there is no such configuration on at most $200 \log n$ vertices. □

Lemma 7.5. *For an integer $k \geq 1$, let G be a connected graph containing no connected subgraph on at most $20k$ vertices with at least two cycles. Consider two distinct vertices $u, v \in V_{2,3}(G)$. Then for $i \leq k$ there is a vertex w with $d(u, w) = i$ and $d(v, w) > i$, or with $d(v, w) = i$ and $d(u, w) > i$.*

Proof. Let $w^1 = u$ and $w^2 = v$, and for $i \in \{1, 2\}$ let $w^i_1, w^i_2, \ldots, w^i_{d_i}$ be the neighbours of w^i. For $i \in \{1, 2\}$ and $j \in \{1, \ldots, d_i\}$, let W^i_j be the set of vertices within distance k of w^i_j, with respect to the graph $G - \{u, v\}$ obtained by deleting u and v.

It may be helpful to imagine an exploration process where we start at a vertex w^i and "explore outwards in the direction of w^i_j", at each step discovering a new "layer" of vertices which are adjacent to previously discovered vertices. Then W^i_j can be interpreted as the set of vertices discovered in the first k steps of this process.

Claim 7.6. *If for any i, j we have that $G[W^i_j]$ contains a cycle, then in fact W^i_j contains a cycle C with the property that C, together with a shortest path between w^i_j and C, collectively comprise a subgraph $G^i_j \subseteq G[W^i_j]$ with at most $2k$ vertices.*

Proof of claim. Consider a breadth-first search tree T (rooted at w^i_j) spanning W^i_j; if W^i_j contains a cycle then $G[W^i_j]$ has an edge xy which does not appear in T. Let C be the cycle consisting of the edge xy, and the unique path in T between x and y. ■

Claim 7.7. *For any i, j, i', j', if the sets W^i_j and $W^{i'}_{j'}$ intersect, then there is a connected subgraph $G^{i,i'}_{j,j'} \subseteq G[W^i_j \cup W^{i'}_{j'}]$ containing w^i_j and $w^{i'}_{j'}$, and fewer than $2k$ vertices in total.*

Proof of claim. If there is $x \in W^i_j \cap W^{i'}_{j'}$, we can simply take $G^{i,i'}_{j,j'}$ to be the union of a shortest path between w^i_j and x followed by a shortest path between x and $w^{i'}_{j'}$. ■

Now, consider the auxiliary multigraph G^* constructed as follows.

- The vertex set of G^* consists of the two vertices u, v and all their neighbours (this is at most $2 + d_1 + d_2$ vertices, but it may be fewer if some neighbours of u and v coincide, or if u and v are adjacent). For each i, j, put an edge[20]between w^i, w^i_j in G^* (these edges are all also in G). Call these *basic edges*.
- For each $(i, j) \neq (i', j')$, if W^i_j and $W^{i'}_{j'}$ intersect, then put an edge between w^i_j and $w^{i'}_{j'}$ in G^* if $w^i_j \neq w^{i'}_{j'}$. Call this a *type-1 special edge*.
- For each i, j, if w^i_j is not incident to a type-1 special edge and W^i_j contains a cycle with respect to G, then put a loop on w^i_j in G^*. Call this a *type-2 special edge*.

Claim 7.8. *G^* does not have a connected component with at least two cycles.*

Proof of claim. For the purpose of contradiction, suppose such a connected component were to exist. Considering all possibilities for the structure of this component, we see that there must be a connected subgraph $G' \subseteq G^*$ with at least two cycles, at most three type-1 special edges, and at most two type-2 special edges (specifically, consider an edge-minimal connected subgraph with at least two cycles). Via

[20] If u and v are adjacent then v is one of the w^1_js and u is one of the w^2_js, but we put only one edge between u and v.

Claims 7.6 and 7.7, the special edges in G' correspond to subgraphs of G which each have at most $2k$ vertices, and the union of these subgraphs, together with the basic edges of G', yields a connected subgraph of G with at least two cycles and at most $(2+3)(2k+2) \leq 20k$ vertices. This contradicts our assumption on G. ■

If every vertex of G^* had degree at least 2, then G^* would have a connected component with at least two cycles (since u, v have degree at least 3). So, Claim 7.8 shows that there is a vertex of G^* which has degree 1. Suppose without loss of generality that this vertex is w_1^1, so W_1^1 does not contain any cycles (of G) and does not intersect any other W_j^i.

Recall that $w^1 = u$ lies in the 2-core of G, so if we "explore outwards from u in the direction of w_1^1" for long enough, we will eventually see a cycle or some other w_j^i. This does not happen within k steps, so for every $i \leq k$ there is some vertex $w \in W_1^1$ with $d(u,w) = i$. Since W_1^1 does not intersect any W_j^2, we have $d(v,w) > i$. □

Now we are ready to complete the proof of Theorem 7.1.

Proof of Theorem 7.1 in the regime $0.9/n \leq p \leq (1 + (\log n)^{-2})/n$. Recall the definitions of $\mathcal{L}^i(u,v)$ and $\mathcal{S}^i(\{u,v\})$ (in Definition 3.4) as the multisets of vertices that "appear differently" in the view exploration processes starting from u and v. For any vertices u, v, w with $d(v,w) = i$ and $d(u,w) > i$ (or vice versa), we have $w \in \mathcal{S}^i(\{u,v\})$. So, by Lemma 7.4 and Lemma 7.5, whp G has the property that for every pair of distinct vertices $u, v \in V_{2,3}$, and every $i \leq 10 \log n$, we have $S^i(\{u,v\}) \neq 0$. By Lemma 7.2, it follows that the vertices in $V_{2,3}(G)$ have distinct colours with respect to $\mathcal{R}^*\sigma$. □

References

[1] E. Aigner-Horev, D. Hefetz, and M. Krivelevich, *Cycle lengths in randomly perturbed graphs*, Random Structures Algorithms **63** (2023), no. 4, 867–884.

[2] D. Angluin, *Local and global properties in networks of processors (extended abstract)*, Proceedings of the Twelfth annual ACM symposium on Theory of computing, STOC '80, ACM Press, 1980.

[3] S. Antoniuk, A. Dudek, C. Reiher, A. Ruciński, and M. Schacht, *High powers of Hamiltonian cycles in randomly augmented graphs*, J. Graph Theory **98** (2021), no. 2, 255–284.

[4] V. Arvind, J. Köbler, G. Rattan, and O. Verbitsky, *Graph isomorphism, color refinement, and compactness*, Comput. Complexity **26** (2017), no. 3, 627–685.

[5] L. Babai, *Graph isomorphism in quasipolynomial time [extended abstract]*, STOC'16—Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2016, pp. 684–697.

[6] L. Babai, *Groups, graphs, algorithms: the graph isomorphism problem*, Proceedings of the International Congress of Mathematicians—Rio de Janeiro 2018. Vol. IV. Invited lectures, World Sci. Publ., Hackensack, NJ, 2018, pp. 3319–3336.

[7] L. Babai, *Canonical form for graphs in quasipolynomial time: preliminary report*, STOC'19—Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2019, pp. 1237–1246.

[8] L. Babai, P. Erdős, and S. M. Selkow, *Random graph isomorphism*, SIAM J. Comput. **9** (1980), no. 3, 628–635.

[9] L. Babai and L. Kucera, *Canonical labelling of graphs in linear average time*, 20th Annual Symposium on Foundations of Computer Science (sfcs 1979), IEEE, October 1979.

[10] J. Balogh, R. Morris, and W. Samotij, *Independent sets in hypergraphs*, J. Amer. Math. Soc. **28** (2015), no. 3, 669–709.

[11] J. Balogh, A. Treglown, and A. Z. Wagner, *Tilings in randomly perturbed dense graphs*, Combin. Probab. Comput. **28** (2019), no. 2, 159–176.

[12] N. Bansal, D. Reichman, and S. W. Umboh, *LP-based robust algorithms for noisy minor-free and bounded treewidth graphs*, Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, PA, 2017, pp. 1964–1979.

[13] H. Bennett, D. Reichman, and I. Shinkar, *On percolation and $\mathcal{NP}$-hardness*, Random Structures Algorithms **54** (2019), no. 2, 228–257.

[14] C. Berkholz, P. Bonsma, and M. Grohe, *Tight lower and upper bounds for the complexity of canonical colour refinement*, Theory Comput. Syst. **60** (2017), no. 4, 581–614.

[15] T. Bohman, A. Frieze, M. Krivelevich, and R. Martin, *Adding random edges to dense graphs*, Random Structures Algorithms **24** (2004), no. 2, 105–117.

[16] T. Bohman, A. Frieze, and R. Martin, *How many random edges make a dense graph Hamiltonian?*, Random Structures Algorithms **22** (2003), no. 1, 33–42.

[17] B. Bollobás, *Distinguishing vertices of random graphs*, Annals of Discrete Mathematics, vol. 13, p. 33–49, North-Holland, 1982.

[18] B. Bollobás, *Random graphs*, second ed., Cambridge Studies in Advanced Mathematics, vol. 73, Cambridge University Press, Cambridge, 2001.

[19] A. Bondy and U. Murty, *Graph theory*, Cambridge University Press, Cambridge, 2010.

[20] J. Böttcher, J. Han, Y. Kohayakawa, R. Montgomery, O. Parczyk, and Y. Person, *Universality for bounded degree spanning trees in randomly perturbed graphs*, Random Structures Algorithms **55** (2019), no. 4, 854–864.

[21] J. Böttcher, R. Montgomery, O. Parczyk, and Y. Person, *Embedding spanning bounded degree graphs in randomly perturbed graphs*, Mathematika **66** (2020), no. 2, 422–447.

[22] J. Böttcher, O. Parczyk, A. Sgueglia, and J. Skokan, *Triangles in randomly perturbed graphs*, Combin. Probab. Comput. **32** (2023), no. 1, 91–121.

[23] J.-Y. Cai, M. Fürer, and N. Immerman, *An optimal lower bound on the number of variables for graph identification*, Combinatorica **12** (1992), no. 4, 389–410.

[24] A. Cardon and M. Crochemore, *Partitioning a graph in $O(|A| \log_2 |V|)$*, Theoret. Comput. Sci. **19** (1982), no. 1, 85–98.

[25] G. Chen and I. N. Ponomarenko, *Lectures on coherent configurations (update June 2024)*, http://www.pdmi.ras.ru/%7Einp/ccNOTES.pdf.

[26] D. Corneil and M. Goldberg, *A non-factorial algorithm for canonical numbering of a graph*, Journal of Algorithms **5** (1984), no. 3, 345–362.

[27] K. P. Costello and V. H. Vu, *The rank of random graphs*, Random Structures Algorithms **33** (2008), no. 3, 269–285.

[28] T. Czajka and G. Pandurangan, *Improved random graph isomorphism*, J. Discrete Algorithms **6** (2008), no. 1, 85–92.

[29] S. Das, P. Morris, and A. Treglown, *Vertex Ramsey properties of randomly perturbed graphs*, Random Structures Algorithms **57** (2020), no. 4, 983–1006.

[30] S. Das and A. Treglown, *Ramsey properties of randomly perturbed graphs: cliques and cycles*, Combin. Probab. Comput. **29** (2020), no. 6, 830–867.

[31] J. Ding, J. H. Kim, E. Lubetzky, and Y. Peres, *Anatomy of a young giant component in the random graph*, Random Structures Algorithms **39** (2011), no. 2, 139–178.

[32] J. Ding, E. Lubetzky, and Y. Peres, *Anatomy of the giant component: the strictly supercritical regime*, European J. Combin. **35** (2014), 155–168.

[33] J. Ding, Z. Ma, Y. Wu, and J. Xu, *Efficient random graph matching via degree profiles*, Probab. Theory Related Fields **179** (2021), no. 1-2, 29–115.

[34] Z. Dvořák, *On recognizing graphs by numbers of homomorphisms*, J. Graph Theory **64** (2010), no. 4, 330–342.

[35] P. Erdős and A. Rényi, *Asymmetric graphs*, Acta Math. Acad. Sci. Hungar. **14** (1963), 295–315.

[36] A. Frieze and M. Karoński, *Introduction to random graphs*, Cambridge University Press, Cambridge, 2016.

[37] F. Fuhlbrück, J. Köbler, and O. Verbitsky, *Identifiability of graphs with small color classes by the Weisfeiler-Leman algorithm*, SIAM J. Discrete Math. **35** (2021), no. 3, 1792–1853.

[38] J. Gaudio and E. Mossel, *Shotgun assembly of Erdős–Rényi random graphs*, Electron. Commun. Probab. **27** (2022), Paper No. 5, 14.

[39] J. Gaudio, M. Z. Rácz, and A. Sridhar, *Average-case and smoothed analysis of graph isomorphism*, arXiv preprint arXiv:2211.16454.

[40] M. Grohe, *Descriptive complexity, canonisation, and definable graph structure theory*, Lecture Notes in Logic, vol. 47, Association for Symbolic Logic, Ithaca, NY; Cambridge University Press, Cambridge, 2017.

[41] M. Grohe, K. Kersting, M. Mladenov, and P. Schweitzer, *Color refinement and its applications*, p. 349–372, The MIT Press, August 2021.

[42] M. Grohe, K. Kersting, M. Mladenov, and E. Selman, *Dimension reduction via colour refinement*, Algorithms—ESA 2014, Lecture Notes in Comput. Sci., vol. 8737, Springer, Heidelberg, 2014, pp. 505–516.

[43] M. Hahn-Klimroth, G. S. Maesaka, Y. Mogge, S. Mohr, and O. Parczyk, *Random perturbation of sparse graphs*, Electron. J. Combin. **28** (2021), no. 2, Paper No. 2.26, 12.

[44] J. Han, P. Morris, and A. Treglown, *Tilings in randomly perturbed graphs: bridging the gap between Hajnal–Szemerédi and Johansson–Kahn–Vu*, Random Structures Algorithms **58** (2021), no. 3, 480–516.

[45] J. Hopcroft, *An $n \log n$ algorithm for minimizing states in a finite automaton*, Theory of machines and computations (Proc. Internat. Sympos., Technion, Haifa, 1971), Academic Press, New York-London, 1971, pp. 189–196.

[46] N. Immerman and E. Lander, *Describing graphs: a first-order approach to graph canonization*, Complexity theory retrospective, Springer, New York, 1990, pp. 59–81.

[47] S. Janson, T. Łuczak, and A. Rucinski, *Random graphs*, Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley-Interscience, New York, 2000.

[48] T. Johnston, G. Kronenberg, A. Roberts, and A. Scott, *Shotgun assembly of random graphs*, arXiv preprint arXiv:2211.14218.

[49] F. Joos and J. Kim, *Spanning trees in randomly perturbed graphs*, Random Structures Algorithms **56** (2020), no. 1, 169–219.

[50] T. Juškevičius and V. Kurauskas, *On the Littlewood-Offord problem for arbitrary distributions*, Random Structures Algorithms **58** (2021), no. 2, 370–380.

[51] R. Kaas and J. M. Buhrman, *Mean, median and mode in binomial distributions*, Statist. Neerlandica **34** (1980), no. 1, 13–18.

[52] R. M. Karp, *Probabilistic analysis of a canonical numbering algorithm for graphs*, Relations between combinatorics and other parts of mathematics (Proc. Sympos. Pure Math., Ohio State Univ., Columbus, Ohio, 1978), Proc. Sympos. Pure Math., XXXIV, Amer. Math. Soc., Providence, RI, 1979, pp. 365–378.

[53] S. Kiefer and D. Neuen, *Bounding the Weisfeiler-Leman dimension via a depth analysis of I/R-trees*, Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, ACM, New York, 2024, p. 14.

[54] S. Kiefer, P. Schweitzer, and E. Selman, *Graphs identified by logics with counting*, ACM Trans. Comput. Logic **23** (2021), no. 1.

[55] A. Krebs and O. Verbitsky, *Universal covers, color refinement, and two-variable counting logic: lower bounds for the depth*, 2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2015), IEEE Computer Soc., Los Alamitos, CA, 2015, pp. 689–700.

[56] M. Krivelevich, M. Kwan, and B. Sudakov, *Bounded-degree spanning trees in randomly perturbed graphs*, SIAM J. Discrete Math. **31** (2017), no. 1, 155–171.

[57] M. Krivelevich, D. Reichman, and W. Samotij, *Smoothed analysis on connected graphs*, SIAM J. Discrete Math. **29** (2015), no. 3, 1654–1669.

[58] M. Krivelevich, B. Sudakov, and P. Tetali, *On smoothed analysis in dense graphs and formulas*, Random Structures Algorithms **29** (2006), no. 2, 180–193.

[59] P. Li and J. Leskovec, *The expressive power of graph neural networks*, p. 63–98, Springer Nature Singapore, 2022.

[60] N. Linial and J. Mosheiff, *On the rigidity of sparse random graphs*, J. Graph Theory **85** (2017), no. 2, 466–480.

[61] R. J. Lipton, *The beacon set approach to graph isomorphism*, Technical Report 135, Yale University. Department of Computer Science, 1978.

[62] A. Magen and M. Moharrami, *Robust algorithms for max independent set on minor-free graphs based on the Sherali-Adams hierarchy*, Approximation, randomization, and combinatorial optimization, Lecture Notes in Comput. Sci., vol. 5687, Springer, Berlin, 2009, pp. 258–271.

[63] H. Mehta and D. Reichman, *Local treewidth of random and noisy graphs with applications to stopping contagion in networks*, Approximation, randomization, and combinatorial optimization. Algorithms and techniques, LIPIcs. Leibniz Int. Proc. Inform., vol. 245, Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2022, pp. Art. No. 7, 17.

[64] H. L. Morgan, *The generation of a unique machine description for chemical structures—a technique developed at Chemical Abstracts Service.*, Journal of Chemical Documentation **5** (1965), no. 2, 107–113.

[65] C. Morris, Y. Lipman, H. Maron, B. Rieck, N. M. Kriege, M. Grohe, M. Fey, and K. Borgwardt, *Weisfeiler and Leman go machine learning: the story so far*, J. Mach. Learn. Res. **24** (2023), Paper No. [333], 59.

[66] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, *Weisfeiler and Leman go neural: Higher-order graph neural networks*, Proceedings of the AAAI Conference on Artificial Intelligence **33** (2019), no. 01, 4602–4609.

[67] D. Neuen and P. Schweitzer, *An exponential lower bound for individualization-refinement algorithms for graph isomorphism*, STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2018, pp. 138–150.

[68] R. Paige and R. E. Tarjan, *Three partition refinement algorithms*, SIAM J. Comput. **16** (1987), no. 6, 973–989.

[69] O. Pikhurko and O. Verbitsky, *Logical complexity of graphs: a survey*, Model theoretic methods in finite combinatorics, Contemp. Math., vol. 558, Amer. Math. Soc., Providence, RI, 2011, pp. 129–179.

[70] B. Pittel, J. Spencer, and N. Wormald, *Sudden emergence of a giant k-core in a random graph*, J. Combin. Theory Ser. B **67** (1996), no. 1, 111–151.

[71] D. Saxton and A. Thomason, *Hypergraph containers*, Invent. Math. **201** (2015), no. 3, 925–992.

[72] M. Singhal, *Erdős–Littlewood–Offord problem with arbitrary probabilities*, Discrete Mathematics **345** (2022), no. 11, 113005.

[73] D. A. Spielman and S.-H. Teng, *Smoothed analysis: motivation and discrete models*, Algorithms and data structures, Lecture Notes in Comput. Sci., vol. 2748, Springer, Berlin, 2003, pp. 256–270.

[74] D. A. Spielman and S.-H. Teng, *Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time*, J. ACM **51** (2004), no. 3, 385–463.

[75] M. Sysło, *Linear time algorithm for coding outerplanar graphs*, Contributions to graph theory and its applications (Internat. Colloq., Oberhof, 1977) (German), Tech. Hochschule Ilmenau, Ilmenau, 1977, pp. 259–269.

[76] G. Tinhofer, *Graph isomorphism and theorems of Birkhoff type*, Computing **36** (1986), no. 4, 285–300.

[77] G. Tinhofer, *A note on compact graphs*, Discrete Appl. Math. **30** (1991), no. 2-3, 253–264, ARIDAM III (New Brunswick, NJ, 1988).

[78] O. Verbitsky and M. Zhukovskii, *Canonical labelling of sparse random graphs*, arXiv preprint arXiv:2409.18109 (2024).

[79] E. M. Wright, *Graphs on unlabelled nodes with a given number of edges*, Acta Math. **126** (1970), 1–9.

[80] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, *How powerful are graph neural networks?*, International Conference on Learning Representations (ICLR) 2019 (2018).

[81] M. Yamashita and T. Kameda, *Computing on anonymous networks, part I: Characterizing the solvable cases*, IEEE Transactions on Parallel and Distributed Systems **7** (1996), no. 1, 69–89.

Appendix A. Canonical labelling proofs

In this section we provide some details for the (routine) proofs of Corollaries 2.7 and 2.14 and Proposition 2.11.

First, we make the basic (well-known) observation that for the purposes of efficient canonical labelling, it suffices to label each connected component separately.

Recall that $\mathcal{G}_n$ is the set of n-vertex graphs (on the vertex set $\{1,\dots,n\}$); we write $\mathcal{G}_{\le n} \subseteq \bigcup_{k=0}^{n} \mathcal{G}_k$ for the set of all graphs on *at most* n vertices.

Fact A.1. *Suppose $\Phi_{\mathcal{H}}$ is a canonical labelling scheme for a graph family $\mathcal{H} \subseteq \mathcal{G}_{\le n}$, such that $\Phi_{\mathcal{H}}(G)$ can be computed in time T for each $G \in \mathcal{H}$. Let $\mathcal{F}$ be the family of graphs whose connected components are all isomorphic to a graph in $\mathcal{H}$. Then there is a canonical labelling scheme Φ for $\mathcal{G}_n$, such that for every $G \in \mathcal{F}$, we can compute $\Phi(G)$ in time $O(n^3 \log n + nT + m)$.*

Proof. For a graph $G \in \mathcal{F}$, our canonical labelling $\Phi(G) \in \mathcal{S}_n$ (interpreted as an ordering of the vertices of G) is computed as follows. First, identify the connected components of G (which takes time $O(n+m)$). For each component C (with n_C vertices, say), we interpret C as a graph on the vertex set $\{1,\dots,n_C\}$, and compute the ordering $\Phi_{\mathcal{H}}(C)$. Since there are at most n components, we can do this in time $O(nT)$.

Now that we have a labelling on each component, we simply need to decide how the components are ordered relative to each other. To do this, we consider the adjacency matrix of each connected component. To unambiguously specify these adjacency matrices we need an ordering of the vertices of each component; we simply use the canonical labelling $\Phi_{\mathcal{H}}(C)$ that we computed above. Then, we sort the components lexicographically by their adjacency matrices[21]. This can be done in time $O(n^3 \log n)$ (the string corresponding to each adjacency matrix has length at most n^2, so comparing two sequences takes time $O(n^2)$, and we are sorting at most n components)[22].

Note that we did make some "arbitrary choices" in this description. Indeed, if there were two components which have the same adjacency matrix, then we had to arbitrarily break a tie when sorting these components. These arbitrary choices are not a problem (i.e., we have indeed described a canonical labelling scheme), because any two outcomes of these arbitrary choices correspond to an automorphism of G. □

Taking $\mathcal{H}$ to be the set of all graphs on at most $C \log n$ vertices, for a constant C, Corollary 2.7 follows immediately from Proposition 2.5, Theorem 2.6 and Fact A.1.

Next, we provide some explanation for Proposition 2.11. First, we observe that the vertices in the kernel, the non-kernel vertices in the 2-core, and the vertices outside the 2-core can all be distinguished from each other via the colour information generated by the colour refinement algorithm. Let $V_2(G)$ be the vertex set of the 2-core of G.

Lemma A.2. *Let G be a graph. The colour $\mathcal{R}^*\sigma(v)$ is enough information to distinguish between the three possibilities $v \in V(G) \setminus V_2(G)$, $v \in V_2(G) \setminus V_{2,3}(G)$ and $v \in V_{2,3}(G)$.*

Proof. Consider the process of "peeling off" vertices of degree less than 2, to generate the 2-core: at each step, we identify all vertices of degree less than 2, and remove them. It is easy to prove by induction that the colour of a vertex with respect to $\mathcal{R}^t\sigma$ is enough information to determine whether that vertex will be peeled off by the t-th peeling step (indeed, recall that $\mathcal{R}^t\sigma(v)$ tells us the number of neighbours that v has in each colour, with respect to $\mathcal{R}^{t-1}\sigma(v)$).

Recall that $\mathcal{R}^*\sigma(v)$ contains all the information about $\mathcal{R}^t\sigma$ for each t until stabilisation is reached. So, $\mathcal{R}^*\sigma(v)$ tells us whether $v \in V_2(G)$. Since $\mathcal{R}^*\sigma(v)$ has the property that every vertex of a given colour has the same number of neighbours in every other colour (Fact 2.16), it follows that, in the 2-core, the vertices of degree at least 3 have different colours than the vertices of degree 2. □

Now we explain how to reconstruct a connected graph G given that the vertices in $V_{2,3}(G) \neq \emptyset$ are assigned distinct colours by $\mathcal{R}^*\sigma$.

Proof of Proposition 2.11. By Lemma A.2, if we know the multiset of colours in $\mathcal{R}^*\sigma$ then we know the multiset of colours assigned to vertices in $V_{2,3}(G)$. Start with a generic set of vertices with these colours. We want to show that the colours specify a unique way to put bare paths between these vertices to reconstruct the 2-core, and they also specify a unique way to attach trees to vertices of the 2-core to reconstruct G.

It would be possible to show this by induction (in a similar fashion to Lemma A.2), but it is perhaps most convenient/intuitive to proceed via universal covers (cf. Remark 3.3).

Indeed, to reconstruct the bare paths between vertices in $V_{2,3}(G)$, consider a vertex $u \in V_{2,3}(G)$, and consider the universal cover $\mathcal{T}_G(u)$ (recall from Lemma 3.2 that this is equivalent information to $\mathcal{R}^*\sigma(u)$). The vertices in this universal cover can be viewed as copies of vertices in G; we do not necessarily know the identities of these vertices, but we do know their colours (with respect to $\mathcal{R}^*\sigma$). Since the vertices in $V_{2,3}(G)$ receive unique, distinguishable colours (by the assumption of this proposition, and Lemma A.2), we can directly read off the lengths of the bare paths between u and other vertices in $V_{2,3}(G)$, and we can read off what the other endpoints of these bare paths are.

To reconstruct the trees attached to vertices of $V_2(G)$, similarly consider a vertex $u \in V_2(G)$ and consider the universal cover $\mathcal{T}_G(u)$. Some of the children of the root are in $V_2(G)$ (we can see this from their colours, by Lemma A.2); after deleting these children (and their subtrees) we see the exact structure of the tree outside the 2-core attached to u. □

Finally, to prove Corollary 2.14, we apply Fact A.1 with $\mathcal{H}$ being the set of graphs on at most n vertices which are either CR-determined or outerplanar (recalling Theorems 2.9 and 2.13 and Proposition 2.11).

[21] Here, we are viewing a $k \times k$ matrix as a string of length k^2, so lexicographic ordering makes sense.

[22] We are being very crude here, it is straightforward to bring the time down to $O(n^2)$.

Appendix B. A simpler proof of a weaker result

Recall the definition of the disparity graph from Definition 1.10. In this appendix we give the details of the argument sketched in Section 1.3.5, proving the following simpler version of Theorem 1.3.

Theorem B.1. *Consider any $p \in [0,1/2]$ satisfying $p \geq 100 \log\log n/n$, consider any graph G_0, and let $G_{\mathrm{rand}} \sim \mathbb{G}(n,p)$. Let σ be the trivial colouring, assigning each vertex the same colour. Then whp the connected components of $D(G_0 \triangle G_{\mathrm{rand}}, \mathcal{R}^*\sigma)$ each have at most $\log n/\log\log n$ vertices. In particular, by Corollary 2.7, whp $G_0 \triangle G_{\mathrm{rand}}$ can be tested for isomorphism with any other graph in polynomial time*[23].

At a very high level the strategy to prove Theorem B.1 is as follows. First, we view the random perturbation G_{rand} as the union of three slightly sparser random perturbations $G^1_{\mathrm{rand}}, G^2_{\mathrm{rand}}, G^3_{\mathrm{rand}} \sim \mathbb{G}(n,p')$. Assuming $p = \Omega(\log\log n/n)$, standard results show that the 3-core of G^1_{rand} (which is a subgraph of the 3-core of G_{rand}) comprises *almost all* the vertices of G, and by Proposition 5.2, whp almost all these vertices are assigned unique colours by $\mathcal{R}^*_G\sigma$.

So, we first reveal the 3-core of G^1_{rand}, and fix a set V_{core} of $n/2$ vertices in this 3-core. We then define the complementary set $U = V(G) \setminus V_{\mathrm{core}}$, and reveal all edges of G_{rand} except those between U and V_{core}. For the rest of the proof we work with the remaining random edges in G^2_{rand} between U and V_{core}.

Specifically, whenever a vertex in U has at least three neighbours in V_{core}, that vertex is guaranteed to be in the 3-core of G_{rand}, and therefore (whp) it is assigned a unique colour. Such vertices are removed from consideration in the disparity graph. To show that such deletions typically shatter the disparity graph into small components, we will use the following lemma.

Lemma B.2. *Let G be an n-vertex graph with maximum degree at most $4 \log n$. Let $G(q)$ be the graph obtained by removing each vertex with probability $q \geq 1 - 1/(\log n)^6$ independently. Then whp $G(q)$ does not have a component of size larger than $\log n/(4 \log\log n)$.*

We prove Lemma B.2 at the end of this section. First, we deduce the statement of Theorem B.1.

Proof of Theorem B.1. Let $G = G_0 \triangle G_{\mathrm{rand}}$. First note that we can assume $p \leq 2 \log n/n$, as otherwise the desired result follows from Theorem 1.2 (note that if $\mathcal{R}^*\sigma$ assigns all vertices distinct colours, then $D(G, \mathcal{R}^*\sigma)$ consists only of isolated vertices.

Write $G_{\mathrm{rand}} = G^1_{\mathrm{rand}} \cup G^2_{\mathrm{rand}} \cup G^3_{\mathrm{rand}}$, where $G^1_{\mathrm{rand}}, G^2_{\mathrm{rand}}, G^3_{\mathrm{rand}} \sim \mathbb{G}(n,p')$ are independent random graphs with edge probability p' satisfying $1 - p = (1-p')^3$. Note that $p' \geq (100/3) \log\log n/n \geq 20 \log\log n/n$.

Since $np' \to \infty$, the 3-core of G^1_{rand} has $n - o(n)$ vertices whp (see e.g. [36, Exercise 2.4.14]). So, whp we can fix a subset $V_{\mathrm{core}} \subseteq V_3(G^1_{\mathrm{rand}})$ with size $n/2$. Let $U = V(G) \setminus V_{\mathrm{core}}$, and for $i \in \{2,3\}$ let S_i be the set of vertices in U which have at least three neighbours in V_{core} with respect to G^i_{rand}. We use the sets S_i, together with the colour refinement algorithm, to recursively define a sequence of vertex-colourings c_1, c_2, c_3 of U, as follows.

- Recall that $\sigma_{G[U]}$ is the trivial vertex-colouring of $G[U]$, and let $c_1 = \mathcal{R}^*_{G[U]}\sigma_{G[U]}$.
- For $i \in \{2,3\}$: let b_i be the vertex-colouring obtained from c_{i-1} by giving all vertices in S_i a unique colour, and let $c_i = \mathcal{R}^* b_i$.

The plan is to first reveal $G[U]$ and c_1 and c_2, using Lemma 6.8 to bound the maximum degree of $D(G[U], c_2)$, and to then reveal c_3, using Lemma B.2 to bound the sizes of the components of $D(G[U], c_3)$. The following claim justifies this plan.

Claim B.3. *Whp the largest connected component in $D(G, \mathcal{R}^*\sigma_G)$ has at most twice as many vertices as the largest connected component in $D(G[U], c_3)$.*

Sketch proof of claim. By Proposition 5.5, whp $\mathcal{R}^*\sigma$ assigns each vertex in the 3-core $V_3(G_{\mathrm{rand}}) \supseteq V_{\mathrm{core}} \cup S_2 \cup S_3$ a unique colour. We will see that the desired conclusion holds whenever this is the case.

Extend each c_i to a colouring c_i^{ext} of $V(G)$, by assigning each vertex in V_{core} a unique colour. Note that $D(G, c_i^{\mathrm{ext}})$ has the same components as $D(G[U], c_i)$ (plus singleton components for each vertex of V_{core}), so by Lemma 6.3 it suffices to show that $\mathcal{R}^*\sigma_G$ is a refinement of each c_i^{ext}. This can be proved inductively, with three applications of Fact 2.17 (note that we are assuming $\mathcal{R}^*\sigma_G$ assigns each vertex a unique colour, so $G[U]$ can be partitioned into full colour classes). ■

[23] Corollary 2.7 uses the Corneil–Goldberg exponential-time canonical labelling scheme on each connected component of the disparity graph, meaning that the components can have up to $O(\log n)$ vertices. However, here we can ensure that the components have at most $\log n/\log\log n$ vertices, so the Corneil–Goldberg labelling scheme is not necessary: we can afford to use a trivial factorial-time canonical labelling scheme on each connected component.

For the rest of the proof, we reveal all the edges in G^1_{rand} (which determines U), and reveal all the edges in $G[U]$. We assume that $|V_3(G^1_{\mathrm{rand}})| \geq n/2$ (so V_{core} and U are well-defined).

Given Claim B.3, our objective is to use the remaining randomness of G^2_{rand} and G^3_{rand} (via the colourings c_2 and c_3) to prove that whp every component of $D(G[U], c_3)$ has at most $\log n/(2\log\log n)$ vertices.

First, note that the random sets S_i are independent. For each $i \in \{2, 3\}$, each vertex $v \in U$ is independently present in S_i with probability

$$\begin{aligned}1 - \mathbb{P}(\mathrm{Binomial}(n/2, p') \leq 2) &\geq 1 - 3\binom{n/2}{2}(p')^2(1-p')^{n/2-2} \\ &\geq 1 - (np')^2 e^{-np'/2} \geq 1 - (20\log\log n)^2 e^{-10\log\log n} \geq 1 - 1/(\log n)^6.\end{aligned}$$

We now track how $D(G[U], c_i)$ evolves with i.

Round 1. Applying Lemma 6.8 with $H = G[U]$, $c = c_1$ and $c' = c_2$, we see that whp $D(G[U], c_2)$ has maximum degree at most $4\log n$. Reveal an outcome of c_2 such that this is the case (i.e., for the rest of the proof, all probabilities should be interpreted as being conditional on an outcome of c_2 with this property).

Round 2. Recall that each vertex is present in S_3 with some probability $q \geq 1 - 1/(\log n)^6$. So, $D(G[U], c_2) - S_3$ is obtained from $D(G[U], c_2)$ by deleting each vertex with probability q, independently. By Lemma B.2, whp the connected components of $D(G[U], c_2) - S_3$ all have size at most $\log n/(4\log\log n)$. Reveal an outcome of c_3 such that this is the case.

Let H' be the graph obtained from $G[U]$ by deleting all edges incident to vertices of S_3 (leaving those vertices isolated). So, apart from isolated vertices, H' has the same components as $G[U] - S_3$, and these components therefore all have size at most $\log n/(4\log\log n)$. But note that H' can be interpreted as a generalised disparity graph $G_L(G[U], c_3)$, so by Lemma 6.3, the connected components of $D(G, c_3)$ all have size at most $\log n/(2\log\log n)$, as desired. □

Now we prove Lemma B.2.

Proof of Lemma B.2. Say that a vertex of G is *dead* if it is not in $G(q)$; otherwise it is *alive.*

Fix $v \in V(G)$. We wish to show that the component of v in $G(q)$ (if v is alive) has at most $\log n/(4\log\log n)$ vertices, with probability at least $1 - o(n^{-1})$ (the desired result will then follow from a union bound over v).

We explore the component of v in $G(q)$ using breadth-first search, revealing whether vertices are dead or alive as we go. Specifically, let $S^0 = \{v\}$ (we can assume v is alive, or there is nothing to prove) and mark v as "explored". For each $t \geq 1$ (in increasing order), let T^t be the set of neighbours of vertices in S^{t-1} which have not yet been marked as "explored", let S^t be the set of alive vertices in T^t, and mark all vertices in T^t as "explored". This process terminates when $S^t = \emptyset$.

Note that $S = \bigcup_{t=0}^{\infty} S^t$ is precisely the connected component of v in $G(q)$. Since each vertex in G has degree at most $4\log n$, we have $|T^t| \leq (4\log n)|S^{t-1}|$ for all $t \geq 1$, so writing $T = \bigcup_{t=0}^{\infty} T^t$ we have $|T| \leq (4\log n)|S|$.

Just like in the proof of Lemma 4.2, we can couple our exploration process with an infinite sequence of Bernoulli$(1-q)$ random variables $(\alpha_i)_{i\in\mathbb{N}}$: whenever we need to know the alive/dead status of a vertex v, we look at the next α_i in our sequence. In particular, $|S| - 1$ is the number of times we see $\alpha_i = 1$ in the first $|T|$ coin flips, so the probability that $|S| \geq \log n/(4\log\log n)$ (i.e., the component of G has at most $\log n/(4\log\log n)$ vertices) is at most the probability that for some $s \geq \log n/(4\log\log n)$, among the first $(4\log n)s + 1$ of the α_i we see $\alpha_i = 1$ at least s times. This probability is at most

$$\begin{aligned}\sum_{s=\frac{\log n}{4\log\log n}} \binom{4s\log n + 1}{s}(1-q)^s \leq \sum_{s=\frac{\log n}{4\log\log n}} (5\log n)^s(1-q)^s = \sum_{s=\frac{\log n}{4\log\log n}} \left(\frac{5}{(\log n)^5}\right)^s \\ = e^{-(5/4+o(1))\log n} = o(n^{-1}).\end{aligned}$$

The desired result follows. □

Institute of Science and Technology Austria (ISTA).
Email address: michael.anastos@ist.ac.at

Institute of Science and Technology Austria (ISTA).
Email address: matthew.kwan@ist.ac.at

University of Manitoba.
Email address: Ben.Moore@umanitoba.ca