Towards Tight Bounds for Testing k-Colorability

Nick Kushnir
School of Mathematics, Tel Aviv University, Israel

Asaf Shapira
School of Mathematics, Tel Aviv University, Israel

Abstract

Determining the sample complexity for testing k-colorability is perhaps the most well studied problem in property testing. It was (implicitly) introduced almost 50 years ago by Bollobás, Erdős, Simonovits and Szemerédi, who used the regularity lemma in order to give a tower-type bound for this problem. This bound has been successively improved in a long list of works, bringing the state-of-the-art bounds to lie between $\Omega(1/\varepsilon)$ and $O((k/\varepsilon)\log^2(1/\varepsilon))$. We obtain the following new results:

(i) Our first main result is an improved $O((k/\varepsilon)\log(1/\varepsilon))$ upper bound, bringing the sample complexity closer to "truly" linear in ε. To prove this result, we improve upon a variant of the container method, introduced recently in the breakthrough paper of Blais and Seth.

(ii) Perhaps the most important gap in our understanding of this problem is that while the best known upper bound increases with k, the lower bound is independent of k. Our second main result fills this gap by providing a new $\Omega(\frac{\log k}{\varepsilon})$ lower bound, improving upon a result of Alon and Krivelevich from 2002. We conjecture that this is the true sample complexity of k-colorability.

2012 ACM Subject Classification Mathematics of computing → Graph theory; Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases k-colorability, property testing, sample complexity, graph property testing, random graphs

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.35

Category RANDOM

Funding *Asaf Shapira*: Supported in part by ERC Consolidator Grant 863438.

1 Introduction

Given a graph $G = (V, E)$, we use $\chi(G)$ to denote its *chromatic number*, that is, the smallest integer k so that V can be colored with the colors $\{1, \ldots, k\}$ so that for each edge $(u, v) \in E$ the vertices u, v are colored differently. Computing $\chi(G)$, or even just determining if $\chi(G) \leq 3$, is of course one of the classical NP-hard problems. One of the best "combinatorial evidence" for the hardness of determining $\chi(G)$ is a classical result of Erdős [12], which states that for arbitrary k and n, there are n-vertex graphs of chromatic number k in which *every* set of $\Omega(n)$ vertices induces a 3-colorable graph. Qualitatively, this means that the chromatic number of a graph lacks any relation between its global and local aspects. Suppose now that we strengthen our global assumption on G, and instead of just assuming that it has large chromatic number, we further assume that it is "far" from having small chromatic number. Can we then show that G has a subgraph on a small number of vertices which witnesses this fact? Interestingly, the answer in this case is dramatically different. We now make this statement more precise.

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2026).
Editors: Mohit Singh and Tom Gur; Article No. 35; pp. 35:1–35:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

For a fixed $\varepsilon > 0$ we say that G is ε-far from being k-colorable, if one should remove from G at least εn^2 edges in order to obtain a graph which is k-colorable[1]. A result of Bollobás, Erdős, Simonovits and Szemerédi [10] from 1978 states that if G is ε-far from being 2-colorable then G contains a set of vertices of size $C(\varepsilon)$ which spans a non 2-colorable graph. This result was later extended by Rödl and Duke to arbitrary fixed k. Both proofs [10, 25] relied on Szemerédi's regularity lemma [29], therefore they only supplied tower-type bounds for $C(\varepsilon)$. Interestingly, both proofs in fact show that a random sample of $C(\varepsilon)$ vertices spans a non-k-colorable subgraph with probability at least 2/3. Let us then define $S_k(\varepsilon)$ to be the smallest q such that if G is ε-far from being k-colorable then a sample of q vertices spans a non-k-colorable subgraph with probability at least 2/3. We call $S_k(\varepsilon)$ the *sample complexity* of k-colorability.

We pause our discussion for a moment and observe that the fact that $S_k(\varepsilon)$ is independent of n has an immediate surprising algorithmic implication. A randomized algorithm is said to be an ε-tester for k-colorability if it accepts k-colorable graphs with probability at least 2/3 and rejects those that are ε-far from being k-colorable with probability at least 2/3. It is now easy to see that we can obtain an ε-tester for k-colorability which simply samples $S_k(\varepsilon)$ vertices, and checks[2] if the subgraph they induce is k-colorable. This observation was made explicit by Goldreich, Goldwasser and Ron [17] in their seminal paper which introduced[3] the area of property testing, and where similar algorithms were designed for more general partition properties. See [5, 16] for more details on property testing. The take-home message here is that besides the purely graph theoretic interest in understanding $S_k(\varepsilon)$, an additional motivation stems from its algorithmic application.

The study of the sample complexity of k-colorability took two dramatic turns in the mid 90's. First, Frieze and Kannan [14, 15] made the crucial observation that in order to study $S_k(\varepsilon)$ one does not need the full power of Szemerédi's regularity lemma [29], and that a weaker regularity lemma suffices. This observation brought down the sample complexity to $2^{(k/\varepsilon)^{O(1)}}$, already making a significant improvement over the regularity-based tower-type bound of Rödl and Duke [25]. Independently of [14, 15], Goldreich, Goldwasser and Ron [17] obtained the first polynomial bound for $S_k(\varepsilon)$. They did so by completely doing away with the regularity lemma method, showing that $S_k(\varepsilon) = \tilde{O}(k^2/\varepsilon^3)$. This bound was later improved by Alon and Krivelevich [1] to $\tilde{O}(k/\varepsilon^2)$. A significant breakthrough was obtained by Sohler [28] who improved the dependence on ε to near linear, giving an upper bound of $\tilde{O}(k^6/\varepsilon)$. In a recent breakthrough, Blais and Seth [7] further improved this, showing that $S_k(\varepsilon) = O((k/\varepsilon)\log^2(1/\varepsilon))$. In terms of lower bounds, Alon and Krivelevich [1] proved that $S_k(\varepsilon) = \Omega(1/\varepsilon)$.

As is evident, determining the correct order of $S_k(\varepsilon)$ has long served as a testbed for developing many of the most important techniques that became ubiquitous in the area of graph property testing. Consequently, resolving this problem was raised as an open problem

[1] Observe that every graph can be made k-colorable by partitioning the vertices into k sets of equal size and removing the edges within these sets. We thus remove at most n^2/k, hence we can always assume that $k \leq 1/\varepsilon$.

[2] Note that this algorithm actually accepts with probability 1 every k-colorable input.

[3] We should point out that [17] used the notion of *query complexity* which is the number of edges we need to sample from G in order to design an ε-tester for k-colorability. Clearly the query complexity is at most quadratic in $S_k(\varepsilon)$, since we can simply query all edges inside the sampled set of vertices. Interestingly, there is a theorem of Goldreich and Trevisan [19] stating that this quadratic gap actually holds for every graph property, and a recent theorem of Goldreich and Wigderson [20] stating that it is sometimes unavoidable. See [21, 18] for further discussion. Since almost all works on k-colorability used the notion of sample complexity, we also use it here.

in several papers[4], for example in 2002 by Alon and Krivelevich [1] and more recently by Blais and Seth [7]. Observe that a striking inconsistency has persisted for the past 20 years: while all upper bounds for $S_k(\varepsilon)$ increase with k, the only known lower bound is independent of k. Our first main result in this paper fills this qualitative gap, improving upon the lower bound obtained in 2002 by Alon and Krivelevich [1].

▶ **Theorem 1.1.** *The sample complexity of k-colorability satisfies $S_k(\varepsilon) = \Omega(\frac{\log k}{\varepsilon})$.*

The main idea of the proof of Theorem 1.1 is the following. As we mentioned above Erdős [12] showed that there is a graph of chromatic number k in which every set of vertices of linear size is 3-colorable. Alon and Krivelevich [1] showed that by taking a blowup of such a graph one can obtain an $\Omega(1/\varepsilon)$ lower bound. The parameters in Erdős's example are tight if one insists that *every* set of vertices of linear size has small chromatic number. Our main innovation here is that in the setting of testing it is enough that most sets satisfy this condition, which allows us to use a theorem of Łuczak [23] on the chromatic number of sparse random graphs.

Our second main result in this paper improves the state of the art upper bound as follows.

▶ **Theorem 1.2.** *The sample complexity of k-colorability satisfies $S_k(\varepsilon) = O(\frac{k}{\varepsilon} \log \frac{1}{\varepsilon})$.*

The proof of Theorem 1.2 builds on the approach introduced in the recent breakthrough of Blais and Seth [7, 6], which itself is a variant of the *Container Method* in extremal combinatorics. This method very roughly states that if the edges of a (hyper)graph are reasonably well distributed then there is a small collection of vertex subsets $C_1, \ldots, C_t$ called *containers*, so that every independent set is a subset of one of these containers. The main advantage is that this lemma allows one to improve upon naive union bounds since instead of having to union bound over all independent sets, one only needs to consider the sets C_i. This method has its origins in the works of Kleitman and Winston [22] and Sapozhenko [26] who considered containers in graphs. A major breakthrough was obtained about 10 years ago by Balogh, Morris, and Samotij [4] and by Saxton and Thomason [27] who extended this method also to hypergraphs, allowing many more applications of this method. Blais and Seth [7, 6] found a clever way of applying this method in the setting of testing graph properties. As in [4, 27] the main challenge is to design an algorithm[5] which receives the graph and constructs the containers. Our main contribution here is in improving the analysis of the algorithm presented in [6]. More concretely, in Lemma 3.10 we remove a $\log 1/\varepsilon$ factor from the bound obtained in [6] for the same lemma (Lemma 6[6] in [6]), which results in the improved upper bound in Theorem 1.2.

The two main differences between our analysis here and the ones in [7, 6] are the following: First, we use the reduction of [6] which reduces testing k-colorability to finding a single independent set in an appropriate graph (the one in [7] requires k independent sets which is more costly), but we run on this graph the container algorithm of [7], which is much simpler to analyse and adapt to our needs. The second idea is to break the analysis of the container

[4] It is worth mentioning that some papers [8, 9] focused on the special case of determining $S_2(\varepsilon)$ (which might be simpler given the simple structure of bipartite graphs) while other papers [2, 6, 11] studied the more general problem of hypergraph colorability.

[5] We stress that by "algorithm" we refer to the procedure that (explicitly) constructs the containers, and not to any algorithmic aspect of testing k-colorability.

[6] The result of [6] actually applies to more general hypergraph coloring properties. Our improved bound can be easily extended to that setting, but we chose to focus on the graph setting in order to simplify the proofs.

algorithm of [7] into Claims 3.3, 3.4 and 3.5, which allows us to make the crucial observation in Claim 3.4. This claim allows us to save a $\log \frac{1}{\varepsilon}$ factor in the edge counts at the end of the proof of Lemma 3.10. To summarize, we combine the ideas presented in [7, 6] with the crucial observation in Claim 3.4.

Our new lower/upper bounds suggest the natural question: what is the true asymptotic behaviour of $S_k(\varepsilon)$? While the k term and the $\log 1/\varepsilon$ term in the upper bound seem like an artifact of the proof, it appears as if the $\log k$ term in the lower bound is much harder to improve upon. We thus suspect that the following holds.

▶ **Conjecture 1.3.** *The sample complexity of k-colorability satisfies $S_k(\varepsilon) = \Theta(\frac{\log k}{\varepsilon})$.*

For what it is worth, our feeling is that it will be hard to remove the final $\log \frac{1}{\varepsilon}$ factor from the upper bound on $S_k(\varepsilon)$ using the method of [7, 6], but that it should be possible to improve the dependence on k (perhaps even all the way to the optimal/conjectured $\log k$) using this method.

Two intermediate goals towards Conjecture 1.3 are to prove an upper bound of $C(k)/\varepsilon$ (for some arbitrary constant $C(k)$) and an upper bound of $\frac{(\log k)^C}{\varepsilon} \log \frac{1}{\varepsilon}$. Observe that a bound of the form $O(\log k/\varepsilon^C)$ (with $C = 7$) is known [3, 13], but is inferior to the known $O(\frac{k}{\varepsilon} \log^{O(1)} \frac{1}{\varepsilon})$ bounds as long as $C \geq 2$ since, as we mentioned earlier, we can assume that $k \leq 1/\varepsilon$. It would also be very interesting to investigate the sample complexity of *tolerantly* testing k-colorability [24], namely, estimating how far is an input from being k-colorable. While this can be done with sample complexity polynomial in $1/\varepsilon$ using the results of [17], determining the sample complexity of this problem seems very challenging.

Paper overview

In Section 2 we prove Theorem 1.1. We first state and prove Lemma 2.3 which is the main lemma in the section, and then use it to prove the lower bound. In Section 3 we prove Theorem 1.2 using the container method.

2 Proof of Theorem 1.1

Our proof of Theorem 1.1 will rely on the following theorem of Łuczak [23] regarding the chromatic number of $G(n, p)$.

▶ **Theorem 2.1.** *There exists a constant d_0 such that if $d = d(n) = n \cdot p(n) > d_0$ and $p(n) = o(1)$, then with probability $1 - o(1)$*

$$\chi(G(n,p)) \leq \frac{np}{\log np}$$

Łuczak [23] actually obtains the much stronger bound $\chi(G(n,p)) < (\frac{1}{2} + o(1))np/\log np$ but the above result suffices for our purposes here. We will also need the following basic Chernoff bound.

▶ **Lemma 2.2.** *Given $p_1, \ldots, p_n \in [0, 1]$ and mutually independent random variables $X_1, \ldots, X_n$, taking values in $\{0, 1\}$, with $\mathbb{P}[X_i = 1] = p_i$, denote $p = (p_1 + \cdots + p_n)/n$ and $X = X_1 + \cdots + X_n$. Then for every $0 < a \leq pn$*

$$\mathbb{P}[X - \mathbb{E}[X] < -a] < e^{-a^2/2pn}$$

In order to prove Theorem 1.1, we will first prove the following lemma.

▶ **Lemma 2.3.** *There is k_0 so that for every $k > k_0$, and every large enough m there is an m-vertex graph H satisfying:*

(i) *For every $1 \le r \le m/80$, a uniformly sampled subset of r vertices of H spans a k-colorable graph with probability at least $\frac{9}{10}$.*

(ii) *One needs to remove from H at least $m \log k$ edges in order to make it k-colorable.*

Proof. Set $p = 40k \log k/m$. To prove the lemma, we will show that for all large enough m, the random graph $H = G(m, p)$ satisfies each of the two desired properties with probability at least $3/4$, and hence there is a graph satisfying both properties.

We first prove that property (i) holds with probability at least $3/4$ with respect to $r = m/80$. Fix a set R of $m/80$ vertices of $H = G(m, p)$. Due to the independence of the $G(m, p)$ model, we can think of the induced subgraph $H[R]$ as generated by $G(m/80, p)$. Thus, by Theorem 2.1, with the parameters $n = m/80, p = 40k \log k/m$, we deduce that for all large enough k[7] and all large enough m, with probability at least 0.99 we have

$$\chi(R) \le \frac{np}{\log np} = \frac{k \log k}{2 \log\left(\frac{k \log k}{2}\right)} < k \; .$$

Let $X = X(H)$ denote the number of subgraphs of $H = G(m, p)$ of size $m/80$ that are not k-colorable. By the above assertion we have $\mathbb{E}[X] \le 0.01 \cdot \binom{m}{m/80}$. Hence, by Markov's inequality

$$\mathbb{P}\left[X > 0.1\binom{m}{m/80}\right] < 1/4 \; .$$

Therefore, with probability at least $3/4$, at most $0.1\binom{m}{m/80}$ of the subsets of H of size $m/80$ span a non-k-colorable subgraph, which is exactly what property (i) states for $r = m/80$. Consider now any $r' < r$. We can sample a random set of size r' by first picking random set of size r and then picking from it a random set of size r'. By the above argument, the random set of size r spans a k-colorable graph with probability at least $9/10$, and since a subgraph of a k-colorable graph is also k-colorable, a random set of size r' is also k-colorable with probability at least $9/10$.

We now prove that property (ii) holds with probability at least $3/4$. Towards this, we first prove that with probability at least $3/4$ every set of vertices R in H of size $|R| = r \ge \frac{m}{2k}$ contains at least $2r \log k$ edges. Let us then fix a subset $R \subset V(H)$ of size r. From Lemma 2.2, with $n = \binom{r}{2}$, all $p_i = p$ and $a = \binom{r}{2}p - 2r \log k$, we have

$$\begin{aligned}
\mathbb{P}[|E(G[R])| < 2r \log k] &< e^{-(\binom{r}{2}p - 2r\log k)^2/2\binom{r}{2}p} \\
&< e^{-(rp - 5\log k)^2/(4p)} \\
&< e^{-pr^2/10} \; .
\end{aligned}$$

[7] We need k to be large enough so that $pm = \frac{1}{2}k \log k$ is larger than d_0 in Theorem 2.1.

Hence, the probability that there is a subset R of size $m/2k \le r \le m$ violating the above property is at most

$$\begin{aligned}\sum_{m/2k\le r\le m}\binom{m}{r}e^{-pr^2/10} &\le \sum_{m/2k\le r\le m}(2ek)^r\cdot e^{-pr^2/10}\\ &\le \sum_{m/2k\le r\le m}e^{r(\ln(2ek)-pr/10)}\\ &\le \sum_{m/2k\le r\le m}e^{r(\ln(2ek)-2\log k)}\\ &\le \sum_{m/2k\le r\le m}e^{r(-\log k)}=o(1)\end{aligned}$$

We now prove that if the condition in the previous paragraph holds, then item (ii) holds as well, and hence it holds with probability at least $3/4$. Indeed, take any partition of $V(H)$ into k sets $C_1,\ldots,C_k$. Since at most half of H's vertices belong to sets C_i of size at most $m/2k$ and since any C_i of size at least $m/2k$ contains at least $2|C_i|\log k$ edges, we conclude that H contains at least $m\log k$ edges within the sets $C_1,\ldots,C_k$. Since this is true for any partition into k sets, at least $m\log k$ edges need to be removed from H to make it k-colorable. ◀

Proof (of Theorem 1.1). Let k_0 be the constant in Lemma 2.3. We may clearly assume that $k>k_0$ since we can either absorb this into the constant in the $\Omega(\frac{\log k}{\varepsilon})$ bound or alternatively just use the $\Omega(\frac{1}{\varepsilon})$ lower bound of [1] which holds for all k. Given $k>k_0$ and all small enough $\varepsilon>0$, we will show that there are infinitely many $n\in\mathbb{N}$ for which there is a graph G which is ε-far from being k-colorable, and yet a randomly chosen set of $\Omega(\frac{\log k}{\varepsilon})$ vertices spans a k-colorable graph with probability at least $9/10$. Set

$$m=\frac{\ln k}{\varepsilon}\ . \tag{1}$$

Since we may assume that ε is small enough, we can assume that m is large enough for Lemma 2.3 to hold. Let H be the graph guaranteed by the lemma. For each integer n, let G be the n/m-blowup of H, that is, the graph obtained by replacing each vertex v of H with an independent set I_v of size n/m, and every edge (u,v) of H with a complete bipartite graph connecting the sets I_u,I_v. We will now show that G satisfies the two properties above, namely, that it is ε-far from being k-colorable, and that a sample of $s=m/80=\Omega(\frac{\log k}{\varepsilon})$ of vertices spans a k-colorable graph with probability at least $9/10$.

Let us number the vertices of H as $1,\ldots,m$ and the vertex sets that replaced them in G as $I_1,\ldots,I_m$. Given a set of vertices S in G, let $S_j=S\cap I_j$ and let S_H be the set of vertices $1\le j\le m$ of H for which $S_j\ne\emptyset$. It is now easy to see that for each S, if the graph $H[S_H]$ is k-colorable then so is $G[S]$. Indeed, we can color all the vertices of I_j in G using the color assigned to vertex j in H. We also observe that by symmetry, if we randomly select a set S in G of size s, then for every $s'<s$, all sets of size s' are equally likely to form the set S_H. We deduce that conditioned on the size of S_H, the set S_H is a uniform subset of H of size at most $m/80$. By Lemma 2.3 a uniform set S_H of any fixed size at most $m/80$ in H is k-colorable with probability at least $9/10$. Hence, if we randomly pick a set S of size $m/80$ with probability at least $9/10$ the graph $H[S_H]$ is k-colorable. But from our observation above, this means that with probability at least $9/10$ the graph $G[S]$ is also k-colorable.

We now prove that G is ε-far from being k-colorable. Take any partition of $V(G)$ into k color sets $C_1, \ldots, C_k$, and let M denote the number of monochromatic edges of this partition, namely, the number of edges within these k sets. Let use this k-coloring in order to (randomly) k-color H as follows: assign color q to a vertex u with probability $|C_q \cap I_u|/|I_u|$. The expected number of monochromatic edges of H in such a coloring is precisely

$$\begin{aligned}\sum_{(u,v)\in E(H)} \sum_{q=1}^{k} \frac{|C_q \cap I_u|}{|I_u|} \frac{|C_q \cap I_v|}{|I_v|} &= \frac{m^2}{n^2} \sum_{q=1}^{k} \sum_{(u,v)\in E(H)} |C_q \cap I_u||C_q \cap I_v| \\ &= \frac{m^2}{n^2} \sum_{q=1}^{k} |(x,y) \in E(G) : x, y \in C_q| \\ &= \frac{M \cdot m^2}{n^2} \,.\end{aligned}$$

By Item (ii) of Lemma 2.3, every k-coloring of H creates at least $m \log k$ monochromatic edges, implying that $\frac{M \cdot m^2}{n^2} \geq m \log k$ and recalling (1) this means that $M \geq \varepsilon n^2$. We conclude that every k-coloring of G creates at least εn^2 monochromatic edges, implying that G is ε-far from k-colorable. ◀

3 Proof of Theorem 1.2

As we mentioned in Section 1, our proof relies on the container method. Algorithm-1 described below receives as input an independent set I in a graph G and produces a sequence $C_1, C_2, \ldots$ of supersets of I, which we call "containers", together with a sequence $F_1, F_2, \ldots$ of subsets of I, which we call "fingerprints". As the name suggests, each fingerprint F_i determines the container C_i. The algorithm constructs these two sequences using a (surprisingly) simple greedy algorithm. Starting with $F_0 = \emptyset$ and $C_0 = V(G)$, in each iteration t the algorithm picks a new vertex $v_t \in I \setminus F_{t-1}$ to be added to the fingerprint. It does so by taking v_t to be the vertex in $I \setminus F_{t-1}$ of the highest degree in $G[C_{t-1}]$ (the graph induced by G on C_{t-1}). It then adds v_t to F_{t-1} to obtain F_t. It also removes all neighbors of v_t from C_{t-1} (none of them belongs to I since it is an independent set) as well as all vertices of C_{t-1} of degree larger than that of v_t (none of them belongs to I by choice of v_t). Note that the algorithm terminates after $|I|$ steps but we further set $C_t = F_t = I$ for $t > |I|$.

Algorithm 1 FINGERPRINT AND CONTAINER GENERATOR.

Input: A graph $H = (V, E)$ and an independent set $I \subseteq V$

Initialize $F_0 \leftarrow \emptyset$ and $C_0 \leftarrow V(H)$;
for $t = 1, 2, \ldots, |I|$ **do**
  $v_t \leftarrow$ the vertex in $I \setminus F_{t-1}$ that has the largest degree in $H[C_{t-1}]$;
  `// remove all vertices with higher degree than` v_t
  $X \leftarrow \{w \in C_{t-1} : \deg_{H[C_{t-1}]}(w) > \deg_{H[C_{t-1}]}(v_t)\}$;
  `// remove all neighbors of` v_t
  $Y \leftarrow \{w \in V : \{w, v_t\} \in E\}$;
  `// Add` v_t `to the fingerprint`
  $F_t \leftarrow F_{t-1} \cup \{v_t\}$;
  `// Update container set by removing` X `and` Y
  $C_t \leftarrow C_{t-1} \setminus (X \cup Y)$;
Return $F_1, \ldots, F_{|I|}$ and $C_1, \ldots, C_{|I|}$;

3.1 Basic Properties

We now prove several useful properties regarding Algorithm 1.

▶ **Definition 3.1.** *For a graph $H = (V, E)$ and an independent set $(I \subseteq V)$, we write $C_t(I)$ for the container obtained after t iterations of the Algorithm when the algorithm is run with input I on the graph H. Similarly, $F_t(I)$ denotes the corresponding fingerprint after t iterations.*

By construction, we have

$$F_1 \subseteq F_2 \subseteq \cdots \subseteq F_{|I|} = I \subseteq C_{|I|} \subseteq C_{|I|-1} \subseteq \cdots \subseteq C_1 \subseteq C_0 \tag{2}$$

Let us state more useful properties of these sequences. Note that the construction of the containers and the fingerprints given the independent set I guarantees that the $C_t(I)$ equals $C_t(F_t(I))$. This is stated in the following claim:

▷ Claim 3.2. For any graph $H = (V, E)$ and any independent set I in H, and any t, the fingerprint $F_t(I)$ and container $C_t(I)$ of I satisfy

$$C_t\big(F_t(I)\big) = C_t(I).$$

Proof. Fix t to be the iteration index. We will prove that $C_t\big(F_t(I)\big) = C_t(I)$. We will prove by induction on the iteration index i that for every $i \le t$ we have $C_i\big(F_t(I)\big) = C_i(I)$. For $i = 0$ the statement is trivial. Let us assume that the statement is true for i and prove it for $i + 1$. From the inductive assumption we know that $C_i\big(F_t(I)\big) = C_i(I)$. Consider the vertex v_{i+1} that was chosen in the $(i + 1) - th$ iteration of the algorithm (running on input I). By construction, the degree of v_{i+1} in $C_i(I)$ is the highest from $I \backslash F_i$. In particular, the degree is also the highest from the following set $F_t \backslash F_i$ in the graph $C_i(I)$ and hence also in $C_i\big(F_t(I)\big)$, and so it is chosen as the $i + 1$ vertex when the algorithm is running on $F_t(I)$ as well. As the current C_i are the same in both algorithms, and so is the chosen vertex in the $(i + 1)$-th iteration, it follows that $C_{i+1}\big(F_t(I)\big) = C_{i+1}(I)$. ◁

▷ Claim 3.3. Let $H = (V, E)$ be a graph, let I be an independent set in H, and let $0 \le t \le |I|$. Denote by Δ_t the maximal degree in C_t. Then, Δ_t is weakly decreasing w.r.t t.

Proof. Follows immediately from the fact that $C_t \subseteq C_{t-1}$ as stated in (2). ◁

▷ Claim 3.4. Let $H = (V, E)$ be a graph, let I be an independent set in H, let $1 \le t \le |I|$ and let Δ_t denote the maximum degree of $H[C_t(I)]$. Then $\Delta_t \le |C_{t-1} \backslash C_t|$.

Proof. Suppose v_t is the vertex in $I \backslash F_{t-1}$ that has the largest degree in $H[C_{t-1}]$. Let $d(v_t)$ denote the degree of vertex v_t in $H[C_{t-1}]$. Then $|C_{t-1} \backslash C_t| \ge d(v_t)$ since Algorithm-1 removes from C_{t-1} the $d(v_t)$ neighbors of v_t in C_{t-1}. On the other hand $\Delta_t \le d(v_t)$ since Algorithm-1 removes from C_{t-1} all vertices of degree larger than $d(v_t)$. ◁

▷ Claim 3.5. Let $H = (V, E)$ be a graph, let I be an independent set in H, and let $1 \le t \le |I|$. Then the maximum degree of $H[C_t]$ is at most $|V|/t$.

Proof. Fix $1 \le t \le |I|$. As stated in (2), we have $C_t \subseteq C_{t-1} \subseteq \cdots \subseteq C_1 \subseteq C_0$. Hence,

$$|V| \ge |C_0| - |C_t| = \sum_{i=1}^{t} \big(|C_{i-1}| - |C_i|\big) = \sum_{i=1}^{t} |C_{i-1} \setminus C_i| \ .$$

Therefore, there is some index $j \in [t]$ such that $|C_{j-1} \backslash C_j| \le |V|/t$. Thus, in the notation of Claim 3.4, we have $\Delta_j \le |V|/t$. By Claims 3.3 and 3.4 we have also $\Delta_t \le |V|/t$. ◁

3.2 Graph Construction

A k-coloring of a graph G is given by k independent sets corresponding to the k color classes. Therefore, if we want to apply the container algorithm to such a description, we have to apply it k times, once for each of the k independent sets. We now describe a simple way to encode the k-colorings of a graph G using another graph H. The advantage is that now every k-coloring of G corresponds to a single independent set in H. This will simplify the analysis in the next two subsections.

▶ **Definition 3.6.** *Given a graph $G = (V, E)$ and an integer k, we define the graph $H_{G,k} = (\bar{V}, \bar{E})$ as follows. If the vertices of G are $V = \{v_1, \ldots, v_n\}$ then $H_{G,k}$ has the following $k \cdot n$ vertices: for each vertex $v_i \in V$, the vertex set $\bar{V}$ contains k vertices $v_{i,1}, \ldots, v_{i,k}$. For what follows, let $S_i = \{v_{i,1}, \ldots, v_{i,k}\}$. The edges of $H_{G,k}$ are the following: for every $v_i \in V$ we put a complete graph on S_v, and for every edge (v_i, v_j) of G and every $1 \leq q \leq k$ we have and edge $(v_{i,q}, v_{j,q})$.*

One should think of vertex $v_{i,q}$ as indicating the fact that v_i is colored q. The following follows immediately from the definition above.

▶ **Fact 3.7.** *Given a graph $G = (V, E)$ and an integer k let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. Suppose $S = \{v_1, \ldots, v_s\} \subseteq V(G)$ is such that $G[S]$ is k-colorable with k-coloring $\phi : S \mapsto [k]$. Then the set $S_\phi = \{v_{1,\phi(v_1)}, \ldots, v_{s,\phi(v_s)}\}$ is an independent set in $H_{G,k}$.*

▶ **Definition 3.8.** *Given a graph $G = (V, E)$ and an integer k, let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. If $U \subset \bar{V}$ then we use $vert(U)$ to denote the set of vertices v for which $S_v \cap U \neq \emptyset$, that is, the set of vertices $v_i \in V(G)$ so that at least one vertex $v_{i,q}$ is in U.*

▶ **Observation 3.9.** *Given a graph $G = (V, E)$ and an integer k, let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. Suppose $U \subset \bar{V}$ satisfies $vert(U) = V(G)$ and $|U| = n$ (i.e. so that $|U \cap S_v| = 1$ for every $v \in V(G)$). Then U encodes a coloring $\phi : V \to [k]$ on the vertices of G. Furthermore, the number of edges in $H[U]$ is exactly the number of monochromatic edges in G w.r.t the coloring ϕ.*

Proof. Let $\phi : V \mapsto [k]$ be such that $\phi(v_i) = q$ if $v_{i,q} \in U$. It is then clear from definition of $H_{G,k}$ that every monochromatic edge created by ϕ corresponds to precisely one edge of $H[U]$. ◀

3.3 A New Container Lemma

The next lemma is the key tool we need for the proof of Theorem 1.2. It improves upon a similar lemma from [6] by a $\log 1/\varepsilon$ factor[8].

▶ **Lemma 3.10.** *Given a graph $G = (V, E)$ and an integer k let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. Suppose that for some $\varepsilon > 0$ the graph G is ε-far from being k-colorable. Then, if I is an independent set in $H_{G,k}$ then there exists $1 \leq t \leq 4k/\varepsilon$ such that*

$$|vert(C_t(I))| \leq \left(1 - \frac{\varepsilon t}{4k}\right) n \; .$$

Here, $C_t(I)$ refers to the container of Algorithm-1.

[8] We remark again that the analogous Lemma 6 in [6] applies also to hypergraph colorability properties, but it is easy to extend our improved bound to this more general setting.

Proof. Assuming by contradiction that for all $1 \le t \le 4k/\varepsilon$ we have

$$|\mathrm{vert}(C_t(I))| > \left(1 - \frac{\varepsilon t}{4k}\right) n \;, \tag{3}$$

we will construct a coloring $\phi : V \to [k]$ of the vertices of G, with less than εn^2 monochromatic edges, contradicting the assumption of the lemma. Observe that by Observation 3.9 we can construct such a coloring by finding a set U of size n in $H_{G,k}$ which satisfies $\mathrm{vert}(U) = V$ and which contains less than εn^2 edges. We will construct U as a union of sets $U_0, \ldots, U_{4k/\varepsilon}$ defined as follows.

Let $C_0(I), \ldots, C_{4k/\varepsilon}(I)$ be the containers produced by Algorithm-1 when applied to I in the graph $H_{G,k}$. Recall that $C_t(I) \subseteq C_{t-1}(I)$ so we will consider these containers in reverse order starting from $C_{4k/\varepsilon}(I)$. Given $C_{4k/\varepsilon}(I)$, let $U_{4k/\varepsilon} \subseteq C_{4k/\varepsilon}(I)$ be the following set: for each vertex $v_i \in G$ for which $S_i \cap C_{4k/\varepsilon}(I) \neq \emptyset$, put in $U_{4k/\varepsilon}$ an arbitrarily chosen vertex from $S_i \cap C_{4k/\varepsilon}(I)$.

We now repeat this for every $t = 4k/\varepsilon - 1, \ldots, 0$, constructing a set $U_t \subseteq C_t(I) \setminus C_{t+1}(I)$ as follows: Let R_t be the vertices $v_i \in V(G)$ so that none of the vertices of S_i belongs to $U_{4k/\varepsilon} \cup \ldots \cup U_{t+1}$. Then for each vertex $v_i \in R_t$ for which $S_i \cap (C_t(I) \setminus C_{t+1}(I)) \neq \emptyset$, put in U_t an arbitrarily chosen vertex from $S_i \cap (C_t(I) \setminus C_{t+1}(I))$.

Let $U = U_{4k/\varepsilon} \cup U_{4k/\varepsilon-1} \cup \ldots \cup U_1 \cup U_0$. Note that our choice or R_t guarantees that U contains at most one vertex from each set S_i. Furthermore, since $C_0 = \bar{V}$ we conclude that U will contain a vertex from each S_i. More precisely, if $S_i \cap C_{4k/\varepsilon} \neq \emptyset$ then we add a vertex of S_i to $U_{4k/\varepsilon}$. Otherwise, if t is the largest[9] integer $1 \le t \le 4k/\varepsilon$ so that C_t contains vertices of S_i, then in step t we will add one of the vertices of S_i to U_t. We thus conclude from Observation 3.9 that U encodes a k-coloring of G. Hence, it remains to prove the U contains less than εn^2 edges.

We claim that $H[U_0 \cup U_1 \cup \cdots \cup U_{4k/\varepsilon}]$ contains less than εn^2 edges. Since $U_{4k/\varepsilon} \subseteq C_{4k/\varepsilon}(I)$, then by Claim 3.5 (with $|\bar{V}| = kn$), the maximum degree in the subgraph $H[U_{4k/\varepsilon}]$ is at most $\frac{kn}{4k/\varepsilon} = \frac{\varepsilon}{4} n$. Hence, $H[U_{4k/\varepsilon}]$ has at most $\frac{\varepsilon}{4} n^2$ edges. Now, consider adding to $U_{4k/\varepsilon}$ the vertices of $U_0 \cup \cdots \cup U_{4k/\varepsilon-1}$ in reverse order. For any $0 \le t \le 4k/\varepsilon - 1$, we choose $U_t \subseteq C_t$, hence when adding U_t, the number of edges in $H[U_{4k/\varepsilon} \cup \ldots \cup U_t]$ increases by at most $|U_t| \cdot \Delta_t$ where Δ_t is the maximum degree of $H[C_t(I)]$. We conclude that the number of edges in U is at most

$$\frac{\varepsilon}{4} n^2 + \sum_{t=0}^{4k/\varepsilon-1} |U_t| \Delta_t \;. \tag{4}$$

Note that our construction guarantees that for any $0 \le t \le 4k/\varepsilon - 1$ we have

$$\mathrm{vert}(U_0 \cup \ldots \cup U_t) = V \setminus \mathrm{vert}(C_{t+1}(I)) \;.$$

From our assumption in (3) we know that $|\mathrm{vert}(C_{t+1}(I))| > \left(1 - \frac{\varepsilon(t+1)}{4k}\right) n$, which means that

$$|\mathrm{vert}(U_0 \cup \ldots \cup U_t)| \le \frac{\varepsilon(t+1)}{4k} n \;.$$

Note also that since (by construction) the sets $\mathrm{vert}(U_0), \ldots, \mathrm{vert}(U_{4k/\varepsilon})$ are disjoint then

$$|\mathrm{vert}(U_0 \cup \ldots \cup U_t)| = \sum_{j=0}^{t} |\mathrm{vert}(U_j)| = \sum_{j=0}^{t} |U_j| \;.$$

[9] Since we consider the containers in reverse order, t is the "first" index satisfying this condition.

We thus conclude that $\sum_{j=0}^{t}|U_j| \le \frac{(t+1)\varepsilon n}{4k}$. By Claim 3.3, the sequence Δ_t is decreasing with respect to t. Hence, in (4), the contribution of $|U_t|$ decreases as t increases, implying that the sum is maximized when $|U_t| = \frac{\varepsilon n}{4k}$ for every $0 \le t \le 4k/\varepsilon - 1$. We can thus bound (4) from above by

$$\frac{\varepsilon}{4}n^2 + \frac{\varepsilon n}{4k}\sum_{t=0}^{4k/\varepsilon-1}\Delta_t = \frac{\varepsilon}{4}n^2 + \frac{\varepsilon n}{4k}\Delta_0 + \frac{\varepsilon n}{4k}\sum_{t=1}^{4k/\varepsilon-1}\Delta_t \le \frac{\varepsilon}{2}n^2 + \frac{\varepsilon n}{4k}\sum_{t=1}^{4k/\varepsilon-1}\Delta_t\ , \tag{5}$$

where in the last inequality we used the fact that $\Delta_0 \le |\bar{V}| = kn$. By Claim 3.4, for every $t \ge 1$ we have $\Delta_t \le |C_{t-1}(I)\backslash C_t(I)|$, implying that the RHS of (5) is at most

$$\frac{\varepsilon}{2}n^2 + \frac{\varepsilon n}{4k}\sum_{t=1}^{4k/\varepsilon-1}|C_{t-1}(I)\backslash C_t(I)| \le \frac{\varepsilon}{2}n^2 + \frac{\varepsilon n}{4k}\cdot|C_0(I)\backslash C_{4k/\varepsilon-1}(I)| < \frac{\varepsilon}{2}n^2 + \frac{\varepsilon n}{4k}\cdot kn < \varepsilon n^2\ ,$$

thus completing the proof. ◀

We now extract the following useful corollary. We remind the reader of the notation S_ϕ in Fact 3.7. If S_ϕ is such a subset of $\bar{V}$ then we will use F_ϕ in order to name subsets of S_ϕ (we mention this since we could call such subsets F without the subscript ϕ).

▶ **Corollary 3.11.** *Given a graph $G = (V, E)$ and an integer k let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. Suppose that for some $\varepsilon > 0$ the graph G is ε-far from being k-colorable. Suppose $S = \{v_1, \ldots, v_s\}$ is such that $G[S]$ is k-colorable with coloring $\phi : S \mapsto [k]$. Then there is $1 \le t \le 4k/\varepsilon$ and an $F_\phi \subseteq S_\phi$ of size $|F_\phi| = t$ so that*
 (i) *$C_t(F_\phi) = C_t(S_\phi)$.*
 (ii) *$|vert(C_t(F_\phi))| \le \left(1 - \frac{\varepsilon t}{4k}\right)n$.*
Here, C_t refers to the containers produced by Algorithm-1.

Proof. By Fact 3.7 we know that S_ϕ is an independent set in $H_{G,k}$. By Lemma 3.10 we know that there is a $1 \le t \le 4k/\varepsilon$ so that $|\mathrm{vert}(C_t(S_\phi))| \le \left(1 - \frac{\varepsilon t}{4k}\right)n$. Let $F_t \subseteq S_\phi$ be the fingerprint of $C_t(S_\phi)$. Then by Claim 3.2 we have $C_t(F_t) = C_t(S_\phi)$ so we also have $|\mathrm{vert}(C_t(F_t))| \le \left(1 - \frac{\varepsilon t}{4k}\right)n$. Finally note that $|F_t| = t$ (by Algorithm-1) and thus we can take F_t as the set F_ϕ in the statement. ◀

3.4 Proof of Theorem 1.2

Suppose $G = (V, E)$ is a graph on n vertices and G is ε-far from being k-colorable. Let $S \subset V$ be a random set of vertices of size $s = C\frac{k}{\varepsilon}\ln(k/\varepsilon)$, drawn uniformly at random and without replacements, where C is a large enough constant to be determined below. We need to show that the probability that $G[S]$ is k-colorable is at most $1/3$. Denote the vertices of S by $\{u_1, \ldots, u_s\}$ so that $[s]$ is the set of indices of these vertices. Let $H_{G,k} = (\bar{V}, \bar{E})$ be the graph from Definition 3.6. If I is an independent set in $H_{G,k}$ then we say that a container $C_t(I) \subset \bar{V}$ is *small* if $|\mathrm{vert}(C_t(I))| \le \left(1 - \frac{\varepsilon t}{4k}\right)n$.

For every subset of indices $R \subseteq [s]$ let $R^* = \{u_i : i \in R\}$ (so $R^* \subseteq S$). If R is as above and $\phi : R \mapsto [k]$ then we let $R^*_\phi = \{u_{i,\phi(i)} : i \in R\}$ (so $R^*_\phi \subseteq \bar{V}$ as in Fact 3.7). For every subset of indices $R \subseteq [s]$ of size $t = |R| \le \frac{4k}{\varepsilon}$ and every coloring function $\phi : R \mapsto [k]$, let $X_{R,\phi}$ be the event that R^*_ϕ is an independent set in $H_{G,k}$, that $C_t(R^*_\phi)$ is a small container, and that ϕ can be extended to coloring of S so that $C_t(R^*_\phi) = C_t(S_\phi)$, where we refer to containers of Algorithm-1 running on $H_{G,k}$. Let $X = \bigcup_{R,\phi} X_{R,\phi}$.

We claim that if $G[S]$ is k-colorable, then event X must hold. Indeed, if ϕ is a k-coloring of $G[S]$ and $F_\phi \subseteq S_\phi$ is the set satisfying Corollary 3.11 then F_ϕ determines the set R and ϕ of one of the events $X_{R,\phi}$ that must hold. This means that if we let C denote the event that $G[S]$ is k-colorable then $\mathbb{P}[C] \le \sum_{X_{R,\phi}} \mathbb{P}[C|X_{R,\phi}]$. We will now complete the proof by showing that this last expression is at most 1/3.

Consider one of the events $X_{R,\phi}$ with $t = |R|$ and assume we first sampled the vertices R^* and then the remaining $s - t$ vertices. If event $X_{R,\phi}$ holds then $|\mathrm{vert}(C_t(R^*_\phi))| \le \left(1 - \frac{\varepsilon t}{4k}\right) n$ and the remaining $s - t$ vertices of S should be such that ϕ can be extended to all of S so that $C_t(R^*_\phi) = C_t(S_\phi)$. But this means in particular that these $s - t$ vertices should all belong to $\mathrm{vert}(C_t(R^*_\phi))$. Indeed, if $s_i \in S$ does not belong to $\mathrm{vert}(C_t(R^*_\phi))$, then for every color q, the vertex $v_{i,q} \notin C_t(R^*_\phi)$ implying that we cannot extend ϕ from R^* to S so that $S_\phi \subseteq C_t(R^*_\phi)$. But by the basic properties of containers $S_\phi \subseteq C_t(S_\phi)$ so we cannot have $C_t(R^*_\phi) = C_t(S_\phi)$. We thus conclude that

$$\mathbb{P}[C|X_{R,\phi}] \le \left(\frac{|\mathrm{vert}(C_t(R^*_\phi))|}{n}\right)^{s-t} \le \left(1 - \frac{\varepsilon t}{4k}\right)^{s-t} .$$

Applying the union bound over all $1 \le t \le 4k/\varepsilon$, all possible choices of subsets $R \subseteq [s]$ of size $|R| = t$, and all the possible colorings[10] ϕ, we have

$$\mathbb{P}[C] \le \sum_{X_{R,\phi}} \mathbb{P}[C|X_{R,\phi}] \le \sum_{t=1}^{4k/\varepsilon} \binom{s}{t} \cdot k^t \cdot \left(1 - \frac{\varepsilon t}{4k}\right)^{s-t} \le \sum_{t=1}^{4k/\varepsilon} \exp\left(t\ln(ks) - \frac{\varepsilon ts}{8k}\right) ,$$

where in the last inequality we use the fact that $s \ge 2t$. It is easy to see that if $s = C\frac{k}{\varepsilon}\log\frac{k}{\varepsilon} = O(\frac{k}{\varepsilon}\log\frac{1}{\varepsilon})$, for[11] a constant C large enough, then the above expression is bounded from above by 1/3. We conclude that the probability that $G[S]$ is colorable, given that G itself is ε-far from being k-colorable, is at most 1/3, thus completing the proof.

References

1. N. Alon and M. Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Mathematics*, 15:211–227, 2002. `doi:10.1137/S0895480199358655`.
2. Noga Alon and Asaf Shapira. Testing satisfiability. *Journal of Algorithms*, 47:87–103, 2003. `doi:10.1016/S0196-6774(03)00019-1`.
3. G. Andersson and L. Engebretsen. Property testers for dense constraint satisfaction programs on finite domains. *Random Structures and Algorithms*, 21:14–32, 2002. `doi:10.1002/rsa.10041`.
4. J. Balogh, R. Morris, and W. Samotij. Independent sets in hypergraphs. *Journal of the American Mathematical Society*, 28:669–709, 2015.
5. A. Bhattacharyya and Y. Yoshida. *Property Testing: Problems and Techniques*. Springer Nature, 2022.
6. E. Blais and C. Seth. New graph and hypergraph container lemmas with applications in property testing. In *Proc. of STOC 2024*, pages 1793–1804, 2024. `doi:10.1145/3618260.3649708`.
7. E. Blais and C. Seth. Testing graph properties with the container method. *SIAM Journal on Computing*, 54(6):1626–1642, 2025. `doi:10.1137/24m1653173`.
8. A. Bogdanov and F. Li. A better tester for bipartiteness? arXiv preprint. `arXiv:1011.0531`.
9. A. Bogdanov and L. Trevisan. Lower bounds for testing bipartiteness in dense graphs. In *Proc. of CCC 2004*, pages 75–81, 2004. `doi:10.1109/CCC.2004.1313803`.

[10] Note (crucially) that we only need to consider coloring of R not of all of S.

[11] The second equality relies on the observation we made on Page 1, that if G is ε-from being k-colorable then $k \le 1/\varepsilon$.

10 B. Bollobás, P. Erdős, M. Simonovits, and E. Szemerédi. Extremal graphs without large forbidden subgraphs. *Annals of Discrete Mathematics*, 3:29–41, 1978.

11 A. Czumaj and C. Sohler. Testing hypergraph coloring. In *Proc. of ICALP 2001*, pages 493–505, 2001. doi:10.1007/3-540-48224-5_41.

12 P. Erdős. On circuits and subgraphs of chromatic graphs. *Mathematika*, 9:170–175, 1962.

13 N. Fiat and D. Ron. On efficient distance approximation for graph properties. In *Proc. of SODA 2021*, pages 1618–1637, 2021.

14 A. Frieze and R. Kannan. The regularity lemma and approximation schemes for dense problems. In *Proc. of FOCS 1996*, pages 12–20, 1996.

15 A. Frieze and R. Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19:175–220, 1999. doi:10.1007/s004930050052.

16 O. Goldreich. *Introduction to property testing*. Cambridge University Press, 2017.

17 O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45:653–750, 1998. doi:10.1145/285055.285060.

18 O. Goldreich and D. Ron. Algorithmic aspects of property testing in the dense graphs model. *SIAM Journal on Computing*, 40:376–445, 2011. doi:10.1137/090749621.

19 O. Goldreich and L. Trevisan. Three theorems regarding testing graph properties. *Random Structures and Algorithms*, 23:23–57, 2003. doi:10.1002/rsa.10078.

20 O. Goldreich and A. Wigderson. Non-adaptive vs adaptive queries in the dense graph testing model. In *Proc. of FOCS 2021*, pages 269–275, 2021.

21 M. Gonen and D. Ron. On the benefits of adaptivity in property testing of dense graphs. *Algorithmica*, 58:811–830, 2010. doi:10.1007/s00453-008-9237-4.

22 D. Kleitman and K. Winston. On the number of graphs without 4-cycles. *Discrete Mathematics*, 41:167–172, 1982. doi:10.1016/0012-365X(82)90204-7.

23 T. Łuczak. The chromatic number of random graphs. *Combinatorica*, 11:45–54, 1991. doi:10.1007/BF01375472.

24 M. Parnas, D. Ron, and R. Rubinfeld. Tolerant property testing and distance approximation. *Journal of Computer and System Sciences*, 72:1012–1042, 2006. doi:10.1016/j.jcss.2006.03.002.

25 V. Rödl and R. Duke. On graphs with small subgraphs of large chromatic number. *Graphs and Combinatorics*, 1:91–96, 1985. doi:10.1007/BF02582932.

26 A. Sapozhenko. Systems of containers and enumeration problems. In *Stochastic Algorithms: Foundations and Applications (SAGA 2005)*, pages 1–13, 2005.

27 D. Saxton and A. Thomason. Hypergraph containers. *Inventiones mathematicae*, 201:925–992, 2015.

28 C. Sohler. Almost optimal canonical property testers for satisfiability. In *Proc. of FOCS 2012*, pages 541–550, 2012.

29 E. Szemerédi. Regular partitions of graphs. In J. C. Bermond, J. C. Fournier, M. Las Vergnas, and D. Sotteau, editors, *Proc. Colloque Inter. CNRS*, pages 399–401, 1978.