2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)

Testing Graph Properties with the Container Method

Eric Blais
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
eric.blais@uwaterloo.ca

Cameron Seth
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
cjmpseth@uwaterloo.ca

2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS) | 979-8-3503-1894-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/FOCS57990.2023.00109

***Abstract*—We establish nearly optimal sample complexity bounds for testing the ρ-clique property in the dense graph model. Specifically, we show that it is possible to distinguish graphs on n vertices that have a ρn-clique from graphs for which at least ϵn^2 edges must be added to form a ρn-clique by sampling and inspecting a random subgraph on only $\tilde{O}(\rho^3/\epsilon^2)$ vertices.**

We also establish new sample complexity bounds for ϵ-testing k-colorability. In this case, we show that a sampled subgraph on $\tilde{O}(k/\epsilon)$ vertices suffices to distinguish k-colorable graphs from those for which any k-coloring of the vertices causes at least ϵn^2 edges to be monochromatic.

The new bounds for testing the ρ-clique and k-colorability properties are both obtained via new extensions of the graph container method. This method has been an effective tool for tackling various problems in graph theory and combinatorics. Our results demonstrate that it is also a powerful tool for the analysis of property testing algorithms.

I. Introduction

Is it possible to test if a graph contains a large clique or if it is k-colorable while examining only a small fraction of that graph? These problems are two of the foundational examples in graph property testing that were originally considered in the groundbreaking work of Goldreich, Goldwasser, and Ron [GGR98]. The problems can be specified formally in the dense graph property testing framework defined as follows.

A (simple undirected) graph G on n vertices is *ϵ-far* from a graph property Π if we need to add or remove at least ϵn^2 edges from G to obtain a graph that does have the property Π. A *canonical ϵ-tester* for Π with sample cost s is a bounded-error[1] randomized algorithm that samples a set S of s vertices of some unknown graph G, examines the induced subgraph $G[S]$, and based only on this local view of the graph can distinguish between the case where G has property Π and the case where it is ϵ-far from Π. The *sample complexity* of Π, denoted $\mathcal{S}_\Pi(n, \epsilon)$, is the minimum value s for which there is a canonical ϵ-tester for Π with sample cost s.

A. Testing Cliques

The ρ-CLIQUE property is the set of all graphs on n vertices that contain a clique on ρn vertices. The problem of testing the ρ-CLIQUE property can be considered in both the *large clique* regime, where ρ is a constant, and in the *small clique* regime, where $\rho := \rho(n)$ is a function of the number of vertices of the graph. In the small clique regime, it is important to note that the problem is non-trivial only when $\epsilon < \rho^2$ is also a function of n, since we can always add at most $\binom{\rho n}{2} < \rho^2 n^2$ edges to any graph to form a ρn-clique.

What is the sample complexity $\mathcal{S}_{\rho\text{-CLIQUE}}(n, \epsilon)$ for ϵ-testing the ρ-CLIQUE property? Goldreich, Goldwasser, and Ron [GGR98] were the first to consider this question and showed that $\mathcal{S}_{\rho\text{-CLIQUE}}(n, \epsilon) = \tilde{O}(\rho/\epsilon^4)$. This result has remarkable qualitative implications in both the large clique and the small clique regimes. In the large clique regime, it shows that ϵ-testing ρ-cliques requires only inspection of a *constant*-sized subgraph when $\epsilon > 0$ is an absolute constant. And in the small clique regime, it shows that the sample complexity for ϵ-testing ρ-clique property is *sublinear* in n for all $\rho = \omega(n^{-1/7})$ when $\epsilon = \Omega(\rho^2)$—i.e., when the tester must distinguish graphs with ρn-cliques from those whose ρn-subgraphs all have constant density bounded away from 1.

Improved bounds on the sample complexity for testing cliques were obtained by Feige, Langberg, and Schechtman [FLS04]. With a new direct analysis of the canonical tester for testing cliques, they showed that $\mathcal{S}_{\rho\text{-CLIQUE}}(n, \epsilon) = \tilde{O}(\rho^4/\epsilon^3)$. And by considering the restricted problem of distinguishing graphs that consist of a single ρn-clique from those that consist of a single $(\rho - \frac{\epsilon}{\rho})n$-clique, they showed that $\mathcal{S}_{\rho\text{-CLIQUE}}(n, \epsilon) = \tilde{\Omega}(\rho^3/\epsilon^2)$.

Our first main result is a new upper bound on the sample complexity of testing the ρ-CLIQUE property that is nearly optimal, since it matches the lower bound of Feige, Langberg, and Schechtman up to polylogarithmic factors.

Theorem 1. *The sample complexity of the ρ-CLIQUE property is $\mathcal{S}_{\rho\text{-CLIQUE}}(n, \epsilon) = \tilde{O}(\frac{\rho^3}{\epsilon^2})$.*[2]

The qualitative implications of Theorem 1 are most striking in the small clique regime. In this regime, the result has a more natural representation when we reformulate it using notation from the Densest k-Subgraph (DkS) problem. (For more details on the DkS problem, see [Man17] and the references therein.)

[1] By *bounded error* we mean there exist absolute constants $\delta_1 > \delta_2$ such that if G has property Π then the algorithm accepts with probability at least δ_1, and if G is ϵ-far from Π then the algorithm accepts with probability at most δ_2.

[2] Here and throughout the article, we use $\tilde{O}(\cdot)$ and $\tilde{\Omega}(\cdot)$ notation to hide terms that are polylogarithmic in the argument. See Sections III and IV for the precise formulation of the main theorems.

979-8-3503-1894-4/23/$31.00 ©2023 IEEE

DOI 10.1109/FOCS57990.2023.00109

Theorem 1' (Alternative formulation). *For every n, $k := k(n) < n$, and $\delta := \delta(n) > 0$, there is a bounded-error randomized algorithm that distinguishes (i) graphs that contain a k-clique from (ii) graphs whose k-subgraphs all contain at most $(1-\delta)k^2$ edges by inspecting the induced subgraph of the input graph G on a random set S of only $|S| = O(\frac{n}{\delta^2 k}\ln^3(\frac{n}{\delta^2 k}))$ vertices.*

This result implies that for every constant $\delta > 0$ and every $k = \omega(\ln^3 n)$, we can distinguish graphs with a k-clique from graphs whose k-subgraphs have density at most $1-\delta$ by examining only a sublinear portion of the graph. The previous bounds in [GGR98] and [FLS04] showed that sublinear sample complexity is achievable for constant δ when $k = \tilde{\omega}(n^{6/7})$ and $k = \tilde{\omega}(n^{1/2})$, respectively.

This result can also be viewed as a generalization of results of Rácz and Schiffer [RS19] and of Huleihel, Mazumdar, and Pal [HMP21] regarding the sample complexity of the Planted k-Clique Detection problem. The latter result shows that sampling an induced subgraph on $O(\frac{n}{k\delta}\ln\frac{n}{k})$ vertices suffices to distinguish (i) random graphs with edge probability $(1-\delta)$ from (ii) random graphs with edge probability $(1-\delta)$ containing a planted k-clique. The former result shows a similar bound for the restricted case of $\delta = 1/2$. When $\delta > 0$ is a constant, the bound in Theorem 1' shows that the same sample complexity (up to logarithmic factors) suffices to solve the more general perfect completeness decision version of the Densest k-Subgraph problem as defined in the theorem statement.

Moreover, Theorem 1' also gives a nearly optimal upper bound on the *query* complexity of algorithms that solve the Densest k-Subgraph problem in the regime where $\delta > 0$ is constant. In this setting, the algorithm is free to query any vertex pair (instead of selecting a set S of vertices and querying all the pairs of vertices in S to determine $G[S]$ exactly), and is free to *adaptively* select which pairs of vertices to query based on the results of its earlier queries. The bound in the theorem implies that there is a bounded-error algorithm that distinguishes graphs with k-cliques from those whose subgraphs have at most $(1-\delta)k^2$ edges and has query complexity at most $O(\frac{n^2}{\delta^4 k^2}\ln^6(\frac{n}{\delta^2 k}))$. However, as shown in [RS19], [HMP21], at least $\Omega(\frac{n^2}{\delta^2 k^2}\ln^2\frac{n}{k})$ queries are required to solve the easier k-Planted Clique problem, even for adaptive algorithms.

B. Testing Colorability

The k-COLORABLE property is the set of all graphs on n vertices that are k-colorable. The case where $k = 2$ corresponds to bipartiteness. For testing bipartiteness, nearly optimal bounds $\mathcal{S}_{2-\text{COLORABLE}}(n,\epsilon) = \tilde{\Theta}(1/\epsilon)$ [GGR98], [AK02] are known. We focus on the case where $k \geq 3$. We again consider two different regimes: the *low chromatic number* regime where k is a constant, and the *polychromatic* regime where $k = k(n)$ is a function of n. In the polychromatic regime, we note that ϵ-testing the k-COLORABLE property is non-trivial only when $\epsilon < 1/k$ (and so also depends on n) since we can always eliminate at most $k\binom{n/k}{2} \leq n^2/k$ edges from a graph to make it k-colorable.

The study of the sample complexity of the k-COLORABLE property has a long history that predates the definition of property testing itself. Rödl and Duke [RD85], building on prior work of Bollobás, Erdős, Simonovits, and Szemerédi [BESS78], showed that $\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon)$ is a constant independent of n when both k and ϵ are constant. However, this result relies on the regularity lemma and so the bounds obtained on the sample complexity grow as a tower of height polynomial in $1/\epsilon$. Notably, this means that these results give only very limited results in the polychromatic regime since they give a sublinear bound on the sample complexity of $k(n)$-colorability only for some $k(n)$ that are iterated logarithm functions of n.

Goldreich, Goldwasser, and Ron [GGR98] obtained much better bounds for k-COLORABLE. They showed that $\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon) = \tilde{O}(k^2/\epsilon^3)$ for all $k \geq 3$. This result provides a significant improvement in the testability of colorability in the polychromatic regime: it shows that k-colorability is testable with a sublinear sample complexity when $\epsilon = \delta/k$ for a constant $\delta > 0$ whenever $k = o(n^{1/5})$.

Alon and Krivelevich [AK02] further improved the bounds for testing k-colorability. They showed that $\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon) = \tilde{O}(k/\epsilon^2)$ for all $k \geq 3$. This result shows that k-colorability can be testable with a sublinear sample complexity for all $k = o(n^{1/3})$. Sohler [Soh12] obtained another improvement in the low chromatic number regime, showing that $\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon) = \tilde{O}(k^6/\epsilon)$ and, in particular, when k is constant then the sample complexity for testing k-colorability is $\tilde{\Theta}(1/\epsilon)$. But in the polychromatic setting, the bound of Alon and Krivelevich remains stronger whenever $k = \omega(1)$ and $\epsilon = \omega(1/k^5)$.

Our next main result unifies and improves on both of the incomparable results of Alon and Krivelevich [AK02] and Sohler [Soh12].

Theorem 2. *The k-COLORABLE property has sample complexity $\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon) = \tilde{O}(\frac{k}{\epsilon})$.*

Theorem 2 implies that sublinear sample complexity suffices to distinguish k-colorable graphs from graphs whose k-colorings all create at least δ/k monochromatic edges for all values of $k = o(\sqrt{n})$ when $\delta > 0$ is constant.

We do not know whether the bound in Theorem 2 is optimal. The best lower bound for testing k-colorability in the polychromatic regime is $\tilde{\Omega}(1/\epsilon)$ [AK02]. Since ϵ-testing k-colorability is non-trivial only when $\epsilon < 1/k$, this means that there is a quadratic gap between our new upper bound and this lower bound for testing k-colorability for large values of k.

C. The Graph Container Method

Theorems 1 and 2 are both established by analyzing the natural tester that checks whether the sampled induced subgraph $G[S]$ has the property that we are testing or not. In the case of large cliques, this tester accepts with probability at least

1/2,[3] and in the case of k-colorability, the tester always accepts when the graph has the property. The challenge in the analysis, as is usually the case in property testing, is in showing that the tester rejects all graphs that are far from having the property with sufficiently large probability. We address this challenge with the use of the graph container method.

We first note that the graph container method is stated in terms of independent sets. In the case of testing ρ-cliques, our application of the graph container method applies more naturally to testing ρ-independent sets, which is equivalent by considering the complement of a graph.

Very briefly, the graph container method builds upon the observation that though a graph may contain a large number of independent sets, for every graph G we can find a much smaller collection of sets of vertices that we call *containers* such that (i) every independent set is a subset of at least one of the containers, (ii) the containers are small, and (iii) for each container C, the induced subgraph $G[C]$ is sparse. The graph container method was originally introduced by Kleitman and Winston [KW82] to bound the number of square-free graphs. The method has since been used to great effect, notably by Sapozhenko (see [Sap05] and the references therein), and was later extended to the hypergraph container method [BMS15], [ST15] that has seen even more wide use throughout combinatorics. The container method has also been recently applied to algorithmic settings for the problem of approximately counting independent sets in bipartite graphs [JPP23] and the problem of obtaining faster exact algorithms for almost-regular graphs [Zam23].

Our use of the graph container method builds on Kleitman and Winston's original approach to it. The key component of the method as they introduced it is a simple greedy algorithm for identifying a small set of vertices of an independent set that form its *fingerprint*. The fingerprint of an independent set I in turn uniquely defines the *container* that contains I. Our main technical lemmas provide tight bounds on the size of the containers as a function of the size of the corresponding fingerprints. We describe the algorithm and present the technical lemmas in Section II.

The tight bounds on the size of containers can then be used to analyze the soundness of testers for ρ-independent sets in the following way. Let S be the set of s vertices sampled by the tester. Any independent set S has a (very small) fingerprint that defines a container with bounded size; we can use this bounded size to show that the probability that more than ρs vertices (including the fingerprint vertices) in S are drawn from that container is extremely small. So small, in fact, that we can apply a union bound over all potential fingerprints in S to bound the overall probability that the sample includes a large independent set. For all the details of the proof, see Section III.

For testing k-colorability, the overall approach is the same, but a different argument is required to show that for any k independent sets in the sample, we can identify k fingerprints whose corresponding containers have a small union. See Section IV for the details.

[3]See the Remark at the end of Section III-C for a discussion related to this choice of acceptance probability.

D. Discussion

Theorems 1 and 2 suggest a number of intriguing open problems. We discuss three of them briefly.

a) Property testing with the container method: Our results demonstrate that the graph container method provides a useful tool to study the problems of testing cliques and colorability. In subsequent work, we show that the graph container method can be used to study all 0–1 graph partition properties, as defined by Nakar and Ron [NR18]. It would be interesting to see if the method is useful for testing other classes of graph properties as well. For example, the class of 0–1 graph partition properties does not include the property of containing one, or multiple, large cliques, and so we suspect there is a larger class of graph partition properties that the container method can be used on.

We would be remiss if we didn't also discuss the hypergraph container method. In the subsequent work mentioned above, we have also been able to show that the hypergraph container method can also be used to provide tight analysis of testers for various partition properties of hypergraphs. However, one of the most exciting aspects of the hypergraph container method in combinatorics is that it has led to a number of results to problems in combinatorics that at first glance do not appear to relate directly to hypergraphs. Could the hypergraph container method similarly yield a new analysis technique for obtaining new property testing results in other domains as well?

b) Query complexity: What exactly is the relation between the query complexity of (adaptive or non-adaptive) testers that are free to query any potential edge in a graph and the query complexity of the canonical tester that samples a set of vertices and queries all potential edges within the corresponding induced subgraph? This question has been studied extensively in the dense graph setting (see, e.g., [AFKS00], [GT03], [BT04], [BL10], [GR10], [GR11], [GW21] and discussions in [Gol17], [BY22]) but remains open for many natural properties of graphs.

As discussed above, Theorem 1 gives a partial answer to this question for the case of testing large cliques, since it shows that the number of edge queries performed by the canonical ϵ-tester for the ρ-CLIQUE property is (up to logarithmic factors) identical to the query complexity of the best adaptive testing algorithm for the same problem when $\epsilon = \Omega(\rho^2)$. But can adaptivity improve query complexity when ϵ is smaller? In particular, is it possible to $\frac{1}{\sqrt{n}}$-test the $\frac{1}{2}$-CLIQUE property with $o(n^2)$ edge queries?

For k-colorability, the analogous question has already been posed even in the setting where $k = 2$: Bogdanov and Li [BL10] conjectured that it is possible to $\frac{1}{n}$-test 2-colorability (that is, bipartiteness) with $o(n^2)$ queries. Theorem 2 does not shed light on this conjecture (or its analogues for larger values of k) but it is possible that Lemma 6 or a similar variant could be helpful in this conjecture and other related ones.

c) Time complexity: Theorem 1' only bounds the sample complexity of the Densest k-Subgraph problem, but the proof of this bound also immediately implies fairly strong results on the time complexity of the problem. Namely, it shows that distinguishing graphs with k-cliques from those whose k-subgraphs all have density at most $(1-\delta)$ for some constant $\delta > 0$ can be done by sampling $s = O(\frac{n}{\delta^2 k} \ln^3(\frac{n}{\delta^2 k}))$ vertices and checking whether the induced subgraph on those vertices contains a clique of size $(k/n)s = O(\delta^{-2} \ln^3(\frac{n}{\delta^2 k}))$. Even by using exhaustive search for the latter task, the resulting time complexity is quasipolynomial in n—or, more precisely, it is $n^{O(\ln^3 n)}$. Stronger time complexity bounds already exist for this problem [FS97], but it would be interesting to see if the graph container method could be used to directly improve the time complexity bounds for various property testing problems and related approximation problems.

II. The Container Method

The notions of *fingerprints* and *containers* of independent sets in graphs are defined according to the procedure defined in Algorithm 1, which is a slight variant of the greedy algorithm that was (implicitly) originally introduced by Kleitman and Winston [KW82]. The algorithm works as follows. The first fingerprint of an independent set I is simply the vertex v contained in I with highest degree in G. Since I is an independent set, any neighbour of v must not be contained in I and, further, by the fact that v has the highest degree in G of all vertices in I, any vertex with higher degree in G cannot be in I. The associated container is constructed by removing all vertices from these two collections. The procedure is repeated until every vertex from I has been selected to the fingerprint.

Algorithm 1: Fingerprint & Container Generator

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

Initialize $F_0 \leftarrow \emptyset$ and $C_0 \leftarrow V$;
for $t = 1, 2, \ldots, |I|$ **do**
  $v_t \leftarrow$ the vertex in $I \setminus F_{t-1}$ with largest degree in $G[C_{t-1}]$;
  // Add v_t to the fingerprint
  $F_t \leftarrow F_{t-1} \cup \{v_t\}$;
  // Remove all the neighbours of v_t from the container
  $C_t \leftarrow C_{t-1} \setminus \{w \in C_{t-1} : (v,w) \in E\}$;
  // And remove all vertices with higher degree than v_t in $G[C_{t-1}]$
  $C_t \leftarrow C_t \setminus \{w \in C_{t-1} : \deg_{G[C_{t-1}]}(w) > \deg_{G[C_{t-1}]}(v_t)\}$;
Return $F_1, \ldots, F_{|I|}$ and $C_1, \ldots, C_{|I|}$;

The greedy algorithm for generating fingerprints and containers was originally defined with an appropriate stopping condition so that each independent set generates a single fingerprint and container. For our proofs, however, it will be more convenient to work directly with the sequence of fingerprints $F_1, F_2, \ldots, F_{|I|}$ and their corresponding containers $C_1, C_2, \ldots, C_{|I|}$ generated by the algorithm for the independent set I. Further, it is also convenient to extend the definition with $C_t = F_t = I$ for all $t > |I|$. We refer to F_t and C_t as the *t-th fingerprint* and *t-th container* of I, respectively.

When we consider the fingerprints or containers of multiple independent sets, we write $F_t(I)$ and $C_t(I)$ to denote the t-th fingerprint and container of the independent set I. (When the current context specifies a single independent set, we will continue to write only F_t and C_t to keep the notation lighter.)

A. Basic Properties of Containers

By construction, the sequence of fingerprints and containers of an independent set satisfy a number of useful properties. For instance, we have

$$F_1 \subseteq F_2 \subseteq \cdots \subseteq F_{|I|} = I \subseteq C_{|I|} \subseteq C_{|I|-1} \subseteq \cdots \subseteq C_2 \subseteq C_1 \tag{1}$$

so that for each $t = 1, 2, \ldots, |I|$, $F_t \subseteq I \subseteq C_t$.

The construction of the algorithm also guarantees that the t-th container of an independent set is the same as the t-th container of its t-th fingerprint.

Proposition 3. *For any graph $G = (V, E)$, any independent set I in G, 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. If $t > |I|$ then $F_t(I) = I = C_t(I)$ and the equality holds. Now consider the case that $t \le |I|$. If $v_1, \ldots, v_t$ are the vertices selected in the first t rounds of the greedy algorithm on input I, then the algorithm selects the same vertices and forms the same container C_t when provided with input $F_t(I)$ instead of I. ☐

Another basic property of the containers of any independent set is that we can bound their maximum degree in the following way.

Proposition 4. *For any graph $G = (V, E)$ on $|V| = n$ vertices, any independent set I in G, and any $t \ge 1$, the t-th container of I has maximum degree bounded by $\Delta(G[C_t]) \le n/t$.*

Proof. If $t > |I|$, then $C_t = I$, and so the maximum degree is 0. Now, consider any $t \le |I|$. Since the containers satisfy the containment $V = C_0 \supseteq C_1 \supseteq \cdots \supseteq C_t$,

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

and there must be an index $j \in [t]$ for which $|C_{j-1} \setminus C_j| \le n/t$. This means that the vertex v_j chosen in the jth round of the greedy algorithm must have degree at most n/t in the induced subgraph $G[C_{j-1}]$. All vertices that had higher degree in the same graph were excluded from C_j, so all vertices have degree at most n/t in $G[C_j]$. Since $C_t \subseteq C_j$, the maximum degree of $G[C_t]$ is bounded above by n/t as well. □

B. Graph Container Lemmas

In order to use containers in our proofs, we need stronger structural results about containers when the graph is ϵ-far from the properties of interest.

In the case of testing the ρ-CLIQUE property, we first change our focus so that we instead look at graphs that are ϵ-far from having the ρ-INDEPSET property. We show that for each such graph G, every independent set in G is a subset of a container of size strictly less than ρn. In fact, our result shows more: there is a container for I of size $(\rho-\alpha)n$ where α is at least as large as ϵ/ρ times the length of the corresponding fingerprint (up to log factors).

Lemma 5 (Graph Container Lemma I)**.** *Let $G = (V, E)$ be a graph on n vertices that is ϵ-far from the ρ-INDEPSET property. For any independent set I in G there exists an index $t \le \frac{8\rho^2}{\epsilon}\ln(\frac{2\rho}{\epsilon})$ such that the size of the t-th container of I is bounded by*

$$|C_t| \le \left(\rho - t\cdot\frac{\epsilon}{8\rho\ln(\frac{2\rho}{\epsilon})}\right)n \tag{2}$$

and $G[C_t]$ contains at most ϵn^2 edges.

Graph container lemmas like the one above are often used to bound the size of a collection of containers that cover all independent sets. Indeed, Lemma 5 also immediately yields such a bound, showing that for every graph G that is ϵ-far from the ρ-INDEPSET property, there is a collection of $n^{\tilde{O}(\rho^2/\epsilon)}$ containers such that every independent set of G is a subset of at least one container. However, for our intended application in the proof of Theorem 1, the size of the collection of containers is not relevant; what we need is the bound on the size of the containers in (2) and, in particular, the trade-off in that bound between the size of the container and its fingerprint.

For testing the k-COLORABLE property, we need a different type of Graph Container lemma which bounds the size of the union of k containers for different independent sets in a graph.

Lemma 6 (Graph Container Lemma II)**.** *Let G be ϵ-far from k-colorable, and let $I_1, \ldots, I_k$ be independent sets in G. Then, there exists $t \le \frac{4}{\epsilon}$ such that*

$$\left|\cup_{i=1}^{k} C_t(I_i)\right| \le \left(1 - t\cdot\frac{\epsilon}{4\ln(\frac{1}{\epsilon})}\right)n. \tag{3}$$

Once again, the upper bound on the size of the union of containers is what we need for the proof of Theorem 2, and this result is possible only because the bound in (3) decreases linearly with the size of the corresponding fingerprints.

We present the proof of Lemma 5 in Section III and the proof of Lemma 6 in Section IV.

III. TESTING CLIQUES AND INDEPENDENT SETS

We complete the proof of Theorem 1 in this section. The proof we describe applies to both the ρ-CLIQUE and the ρ-INDEPSET properties (by considering the complement of the tested graph) but the latter fits in more naturally with the graph container method, so in the rest of the section we discuss the result entirely in terms of independent sets.

A. Container Shrinking Lemma

The main tool in the proof of the Graph Container Lemma for testing independent sets, Lemma 5, is a shrinking lemma that, informally speaking, says that many vertices are excluded in each round of Algorithm 1 whenever the containers contain a large subset C for which the induced subgraph $G[C]$ is sparse.

Lemma 7 (Container Shrinking Lemma)**.** *Let $G = (V, E)$ be a graph on n vertices that is ϵ-far from the ρ-INDEPSET property, let I be an independent set in G, and let $t < |I|$ be an index for which the t-th container C_t of I has cardinality $|C_t| \ge \rho n$. For any $\alpha > 0$, if there exists a subset $C \subseteq C_{t+1}$ of $(\rho-\alpha)n$ vertices where $G[C]$ contains at most $\frac{\epsilon}{4}n^2$ edges, then*

$$|C_{t+1} \setminus C| \le \left(1 - \frac{\epsilon}{4\rho\alpha}\right)|C_t \setminus C|\,.$$

Proof. If C contains a vertex v with degree at least $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|$ in the graph $G[C_t|$ then at least this number of vertices are eliminated since the vertex v_{t+1} selected by the greedy algorithm has degree at least as large as that of v, otherwise v would not have been included in C_{t+1} (and so therefore would not be in C).

Consider now the setting where all the vertices in C have degree at most $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|$. We claim that $G[C_t \setminus C]$ contains at least $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|^2$ edges. This claim suffices to complete the proof of the lemma, since it implies that at least $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|$ vertices in $C_t \setminus C$ have degree at least $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|$ in $G[C_t \setminus C]$, and thus that at least this many vertices from $C_t \setminus C$ are excluded from C_{t+1} by Algorithm 1.

We now complete the proof of the claim that $G[C_t \setminus C]$ contains at least $\frac{\epsilon}{4\rho\alpha}|C_t \setminus C|^2$ edges.

Let R be chosen uniformly at random from all subsets of $C_t \setminus C$ of size αn. Since G is ϵ-far from having independent sets of size ρn, the induced subgraph $G[R \cup C]$ contains at least ϵn^2 edges. By the lemma hypothesis, $G[C]$ contains at most $\frac{\epsilon}{4}n^2$ edges. Finally, each edge connecting a vertex in C to one in $C_t \setminus C$ is included in $G[R \cup C]$ with probability $|R|/|C_t \setminus C|$. So by the maximum degree of the vertices in C in the graph $G[C_t]$, the expected number of edges between C and R is at most $\frac{\epsilon}{4\alpha}|R|n = \frac{\epsilon}{4}n^2$. Therefore, the expected number of edges in $G[R]$ is at least

$$\frac{\epsilon}{2}n^2 = \frac{\epsilon}{2\alpha^2}|R|^2 \ge \frac{\epsilon}{\alpha^2}\binom{|R|}{2}.$$

In other words, the random graph R chosen uniformly at random from the subgraphs of $C_t \setminus C$ of cardinality αn has expected density at least $\frac{\epsilon}{\alpha^2}$. This implies that $C_t \setminus C$ also has density at least $\frac{\epsilon}{\alpha^2}$. So it contains at least $\frac{\epsilon}{\alpha^2}\binom{|C_t \setminus C|}{2} \geq \frac{\epsilon}{\rho\alpha}\binom{|C_t \setminus C|}{2} \geq \frac{\epsilon}{4\rho\alpha}|C_t \setminus C|^2$ edges, as we wanted to show. $\square$

B. Proof of Graph Container Lemma I

We are now ready to complete the proof of the Graph Container Lemma for testing independent sets, restated below.

Lemma 5 (Graph Container Lemma I)**.** *Let $G = (V, E)$ be a graph on n vertices that is ϵ-far from the ρ-*IndepSet *property. For any independent set I in G there exists an index $t \leq \frac{8\rho^2}{\epsilon}\ln(\frac{2\rho}{\epsilon})$ such that the size of the t-th container of I is bounded by*

$$|C_t| \leq \left(\rho - t \cdot \frac{\epsilon}{8\rho\ln(\frac{2\rho}{\epsilon})}\right) n \tag{2}$$

and $G[C_t]$ contains at most ϵn^2 edges.

Proof. Let C denote the first container in the sequence $C_1, C_2, \ldots$ that contains at most $\frac{\epsilon}{4}n^2$ edges. The existence of C is guaranteed because $C_{|I|+1}$ is an independent set and has no edges. Define α such that $|C| = (\rho - \alpha)n$. The value of α is bounded by $\frac{\epsilon}{2\rho} \leq \alpha \leq \rho$, where the lower bound on α is due to the fact that G is ϵ-far from the ρ-IndepSet property.

Define t^* to be the largest index for which $|C_{t^*}| \geq \rho n$. First observe that $t^* \leq |I|$ because otherwise C_{t^*} is an independent set, which contradicts the fact that G is ϵ-far from the ρ-IndepSet property. Hence, by Lemma 7 applied to all values of $t = 0, 1, 2, \ldots, t^* - 1$,

$$|C_{t^*} \setminus C| \leq \left(1 - \frac{\epsilon}{4\rho\alpha}\right)^{t^*} n$$

and since $|C_{t^*} \setminus C| \geq \alpha n$, we conclude that $t^* \leq \frac{4\rho\alpha}{\epsilon}\ln(1/\alpha) \leq \frac{4\rho\alpha}{\epsilon}\ln(2\rho/\epsilon)$.

Consider now values $t > t^*$. When $G[C_t]$ contains more than $\frac{\epsilon}{4}n^2$ edges, then the vertices in C_t have average degree at least $\frac{\epsilon}{4\rho}n$ in this graph. This means that at least $\frac{\epsilon}{4\rho}n$ vertices in C_t have degree at least $\frac{\epsilon}{4\rho}n$ in $G[C_t]$, so at least $\frac{\epsilon}{4\rho}n$ vertices are removed in the t-th iteration of Algorithm 1. Since $|C_{t^*+1} \setminus C| \leq \alpha n$, there can be at most $\frac{4\rho\alpha}{\epsilon}$ such iterations before we reach C. Therefore, the container C satisfies the conclusion of the lemma. $\square$

C. Proof of Theorem 1

We are now ready to use Lemma 5 to complete the proof of Theorem 1. The proof of the theorem also makes use of the following form of Chernoff's bound for hypergeometric distributions.

Lemma 8 (Chernoff's Bound)**.** *Let X be drawn from the hypergeometric distribution $H(N, K, n)$ where n elements are drawn without replacement from a population of size N where K elements are marked and X represents the number of marked elements that were drawn. Then for any $\vartheta \geq E[X]$,*

$$\Pr\left[X \geq \vartheta\right] \leq \exp\left(-\frac{(\vartheta - E[X])^2}{\vartheta + E[X]}\right).$$

Proof. A standard multiplicative form of Chernoff's bound states that for all $\delta > 0$, the random variable X satisfies $\Pr\left[X \geq (1+\delta)E[X]\right] \leq \exp\left(-\frac{\delta^2 E[X]}{2+\delta}\right)$. (This standard bound is usually stated for sums of independent variables, but the same bound also applies to hypergeometric random variables as well. See, e.g., [Mul18]). Using the identity $\vartheta = \left(1 + \frac{\vartheta - E[X]}{E[X]}\right)E[X]$, we can apply this inequality with the parameter $\delta = \frac{\vartheta - E[X]}{E[X]}$ and simplify. $\square$

We are now ready to prove Theorem 1, restated below in its precise form (with the polylogarithmic term) and for the ρ-IndepSet property.

Theorem 1 (Precise formulation)**.** *The sample complexity of the ρ-*IndepSet *property is*

$$\mathcal{S}_{\rho\text{-IndepSet}}(n, \epsilon) = O\left(\frac{\rho^3}{\epsilon^2}\ln^3\left(\frac{1}{\epsilon}\right)\right).$$

Proof. Let S be a random set of $s = c \cdot \frac{\rho^3}{\epsilon^2}\ln^3\left(\frac{1}{\epsilon}\right)$ vertices drawn uniformly at random from V without replacement, where c is a large enough constant. Note that for clarity of presentation, in the rest of the proof we ignore all integer rounding issues as they do not affect the asymptotics of the final result.

If G contains a ρn independent set, then S contains at least ρs vertices from this independent set with probability at least $\frac{1}{2}$, since the number of such vertices follows a hypergeometric distribution, and the median of this distribution is at least ρs [Neu70], [KB80].

In the remainder of the proof we upper bound the probability that S contains a ρs-independent set when G is ϵ-far from containing a ρn-independent set. For the rest of the argument, let us call a container C_t *small* when its cardinality is bounded by $|C_t| \leq \left(\rho - t \cdot \frac{\epsilon}{8\rho\ln(2\rho/\epsilon)}\right)n$, the expression (2) in the conclusion of Lemma 5.

Let us denote the vertices of the sampled set S as $u_1, u_2, \ldots, u_s$. First observe that, by Lemma 5, for *every* independent set I of size ρs in S, there is a $t \leq \frac{8\rho^2\ln(\frac{2\rho}{\epsilon})}{\epsilon}$ and a fingerprint $F_t \subseteq I$ of size t that defines a small container $C_t(F_t)$ such that $I \subseteq C_t(F_t)$. Hence, the probability that S contains an independent set of size at least ρs is at most the probability that there exists some $t \leq \frac{8\rho^2\ln(\frac{2\rho}{\epsilon})}{\epsilon}$ such that S contains a set of t vertices that form the fingerprint F_t of a small container C_t, and S contains at least $\rho s - t$ other vertices from C_t. We now upper bound this probability.

For any t distinct indices $i_1, i_2, \ldots, i_t \in [s]$, consider the event where the vertices $u_{i_1}, \ldots, u_{i_t}$ form the fingerprint F_t of a small container C_t. Let X denote the number of vertices among the other $s - t$ sampled vertices that belong to C_t. By the bound on the size of small containers, the expected value of X is

$$\begin{aligned}\mathrm{E}[X] &\leq \left(\rho - \frac{t\epsilon}{8\rho\ln(\frac{2\rho}{\epsilon})}\right)(s-t)\\ &< \rho s - \frac{t\epsilon s}{8\rho\ln(\frac{2\rho}{\epsilon})}\\ &\leq \rho s - t - \frac{t\epsilon s}{16\rho\ln(\frac{2\rho}{\epsilon})},\end{aligned}$$

where the last inequality uses the fact that $s = c\cdot\frac{\rho^3}{\epsilon^2}\ln^3(\frac{1}{\epsilon})$ for a large enough constant c and that the problem is non-trivial only when $\epsilon < \rho^2$.

By the Chernoff bound in Lemma 8, the probability that we draw at least $\rho s - t$ vertices from C_t in the final $s-t$ vertices drawn to form S is

$$\begin{aligned}\Pr[X \geq \rho s - t] &\leq \exp\left(\frac{-(\rho s - t - E[X])^2}{\rho s - t + E[X]}\right)\\ &\leq \exp\left(-\frac{t^2\epsilon^2 s}{512\rho^3\ln^2(\frac{2\rho}{\epsilon})}\right).\end{aligned}$$

Therefore, by applying a union bound over all values $t \leq \frac{8\rho^2\ln(\frac{2\rho}{\epsilon})}{\epsilon}$ and all possible choices of t indices from $[s]$, the probability that *any* set of at most $\frac{8\rho^2\ln(\frac{2\rho}{\epsilon})}{\epsilon}$ vertices in S form the fingerprint of a small container from which we sample at least ρs vertices in S is at most

$$\begin{aligned}&\sum_t \binom{s}{t}\exp\left(-\frac{t^2\epsilon^2 s}{512\rho^3\ln^2(\frac{2\rho}{\epsilon})}\right)\\ &\leq \sum_t \exp\left(t\ln s - \frac{t^2\epsilon^2 s}{512\rho^3\ln^2(\frac{2\rho}{\epsilon})}\right) < \frac{1}{3}\end{aligned}$$

where the first inequality uses the upper bound $\binom{s}{t} \leq s^t$, and the last inequality again uses the fact that $s = c\cdot\frac{\rho^3}{\epsilon^2}\ln^3(\frac{1}{\epsilon})$ for a large enough constant c. Hence, the probability that S contains an independent set of size at least ρs is less than $1/3$. □

Remark. The completeness guarantee in the proof of Theorem 1 states that the algorithm correctly accepts graphs that have a ρn-independent set with probability at least $\frac{1}{2}$, instead of the usual $\frac{2}{3}$ bound typically used in property testing. Since the soundness guarantee is that the algorithm accepts graphs that are ϵ-far from the ρ-INDEPSET property with probability at most $\frac{1}{3}$, these guarantees still enable us to apply standard error reduction techniques to obtain algorithms that err with probability at most γ with a $\ln(1/\gamma)$ multiplicative factor. In particular, it shows that there is also a standard tester with completeness guarantee $\frac{2}{3}$ with the same asymptotic sample complexity.

Alternatively, it is also possible to obtain the $\frac{2}{3}$ completeness guarantee directly by changing the algorithm slightly to check if the induced subgraph has the $(\rho-\tau)$-INDEPSET property for some appropriate gap parameter $\tau := \tau(\rho,\epsilon)$.

IV. TESTING K-COLORABILITY

In this section we prove Theorem 2. We start by proving a graph container lemma in Section IV-A, and complete the proof in Section IV-B.

A. Proof of Graph Container Lemma II

In this section we prove the main lemma that balances the trade off between the size of the union and the sum of the fingerprints sizes of a set of containers, restated from Section II-B.

Lemma 6 (Graph Container Lemma II)**.** *Let G be ϵ-far from k-colorable, and let $I_1,\ldots,I_k$ be independent sets in G. Then, there exists $t \leq \frac{4}{\epsilon}$ such that*

$$\left|\cup_{i=1}^k C_t(I_i)\right| \leq \left(1 - t\cdot\frac{\epsilon}{4\ln(\frac{1}{\epsilon})}\right)n. \tag{3}$$

Proof. Assume on the contrary that for all $t \leq 4/\epsilon$,

$$\left|\cup_{i=1}^k C_t(I_i)\right| > \left(1 - t\cdot\frac{\epsilon}{4\ln(1/\epsilon)}\right)n. \tag{4}$$

We will obtain a contradiction by constructing a k-partition $V_1,\ldots,V_k$ of the vertices which shows that G is not $\frac{3\epsilon}{4}$-far from k-colorable.

Initialize the sets $V_1 = C_{4/\epsilon}(I_1),\ldots,V_k = C_{4/\epsilon}(I_k)$. Remove vertices from $V_1,\ldots,V_k$ until every vertex appears in one set so that $V_1,\ldots,V_k$ forms a partition of $\cup_{i=1}^k C_{4/\epsilon}(I_i)$. By Proposition 4, each vertex has degree at most $\epsilon n/4$ in it's set, and so the sum of edges contained in the sets $V_1,\ldots,V_k$ is at most $\frac{\epsilon n^2}{4}$.

In the remainder of the proof, we allocate the remaining vertices in $V\setminus\cup_{i\in[k]}V_i = V\setminus\cup_{i\in[k]}C_{4/\epsilon}(I_i)$ to one of the parts and argue that the number of additional edges is small.

For every $t = 1,\ldots,\frac{4}{\epsilon}$, let A_t be the set of vertices that are contained in at least one container at the $(t-1)$-th step, but not contained in any container at step t step of the container procedure. In other words, let

$$A_t = \{v\in V : v\in\cup_{i\in[k]}C_{t-1}(I_i) \text{ and } v\notin\cup_{i\in[k]}C_t(I_i).\}$$

Observe that $A_1,\ldots,A_{4/\epsilon}$ partitions the vertices in $V\setminus\cup_{i\in[k]}V_i$, and so to complete the partition of G we add the vertices of $A_1,\ldots,A_{4/\epsilon}$ to $V_1,\ldots,V_k$ in the following way: for each $t = 1,\ldots,\frac{4}{\epsilon}$ and $v\in A_t$, select an i such that $v\in C_{t-1}(I_i)$, and add v to set V_i.

In order to count the new edges, consider adding the vertices from $A_1,\ldots,A_{4/\epsilon}$ in reverse order (starting at vertices in $A_{4/\epsilon}$ and going to A_1). In this way, when $v\in A_t$ is added to V_i, it holds that $V_i\subseteq C_{t-1}(I_i)$. Hence, by Proposition 4, each $v\in A_t$ contributes at most $\frac{n}{t-1}$ new edges to V_i (or n edges if $t=1$) because v is contained in $C_{t-1}(I_i)$.

Then, the sum of the edges in each of the sets $V_1,\ldots,V_k$ can be upper bounded by

$$\frac{\epsilon n^2}{4} + |A_1|\cdot n + \sum_{t=2}^{4/\epsilon}|A_t|\frac{n}{(t-1)}. \tag{5}$$

For any t, $\cup_{\ell=1}^{t} A_t = V \setminus \cup_{i\in[k]} C_t(I_i)$, and so (4) implies that $\sum_{\ell=1}^{t} |A_\ell| \le \frac{t\epsilon n}{4\ln(1/\epsilon)}$. In the sum (5), the contribution from $|A_t|$ goes down as t increases, and so the sum is maximized when $|A_t| = \frac{\epsilon n}{4\ln(1/\epsilon)}$ for all t. Hence, the sum of the edges in each of the sets $V_1, \ldots, V_k$ can be upper bounded by

$$\frac{\epsilon n^2}{4} + \frac{\epsilon n^2}{4\ln(1/\epsilon)} + \frac{\epsilon n^2}{4\ln(1/\epsilon)} \sum_{t=2}^{4/\epsilon} \frac{1}{t-1} < \epsilon n^2,$$

where the last inequality uses the upper bound $H(m) \le \ln(m)+1$ on the harmonic series and the bound $\epsilon < e^{-2}$ that we can apply without loss of generality. This is a contradiction with the fact that G is ϵ-far from k-colorable. □

B. Proof of Theorem 2

We can now complete the proof of Theorem 2, restated below.

Theorem 2 (Precise formulation)**.** *The sample complexity of the k-COLORABLE property is*

$$\mathcal{S}_{k\text{-COLORABLE}}(n,\epsilon) = O\left(\frac{k}{\epsilon}\ln^2\left(\frac{1}{\epsilon}\right)\right).$$

Proof. Let S be a random set of $s = c\frac{k}{\epsilon}\ln^2\left(\frac{1}{\epsilon}\right)$ vertices drawn uniformly at random from V without replacement, where c is a large enough constant. The tester accepts a graph G if and only if $G[S]$ is k-colorable. When G is k-colorable, then $G[S]$ is also k-colorable so the tester always accepts.

In the remainder of the proof, we upper bound the probability that $G[S]$ is k-colorable when G is ϵ-far from k-colorable. In the following, let us say that k containers $C_t(I_1), \ldots, C_t(I_k)$ have a *small union* when $\left|\cup_{i=1}^{k} C_t(I_i)\right| \le \left(1 - \frac{t\epsilon}{4\ln(1/\epsilon)}\right) n$.

Let us denote the vertices of S as $u_1, u_2, \ldots, u_s$. For any $t \le \frac{4}{\epsilon}$, any values $t_1, \ldots, t_k \le t$, and any set of $t_1+t_2+\cdots+t_k$ distinct indices $\{i_{j,1}, i_{j,2}, \ldots, i_{j,t_j}\}_{j=1,\ldots,k}$ in $[s]$, let us consider the event where the fingerprints $F_t^{(j)} = \{i_{j,1}, \ldots, i_{j,t_j}\}$ define a collection of containers $C_t(F_t^{(1)}), \ldots, C_t(F_t^{(k)})$ with a small union. When this event occurs, the probability that the remaining $s - (t_1 + \cdots + t_k) \ge s - tk$ vertices in S are all drawn from $\cup_{j=1}^{k} C_t(F_t^{(j)})$ is at most $\left(1 - \frac{t\epsilon}{4\ln(1/\epsilon)}\right)^{s-tk}$.

For any t, there are at most $(\binom{s}{\le t})^k \le s^{tk}$ ways to choose k sets of at most t distinct indices as above. So by applying a union bound argument over all $t \le \frac{4}{\epsilon}$ and over all possible choices of indices, the probability that any set of at most $\frac{4k}{\epsilon}$ vertices in S form the k fingerprints of a family of containers with a small union from which we sample all the remaining vertices is at most

$$\sum_{t=1}^{4/\epsilon} s^{tk}\left(1 - \frac{t\epsilon}{4\ln(1/\epsilon)}\right)^{s-tk}$$
$$\le \sum_{t=1}^{4/\epsilon} \exp\left(tk\ln(s) - \frac{t\epsilon s}{8\ln(1/\epsilon)}\right) < 1/3$$

where the first inequality uses the fact that $s > 2tk$ and the second is obtained by using the fact that $s = c\frac{k}{\epsilon}\ln^2\left(\frac{1}{\epsilon}\right)$ for a large enough constant c (and that the problem is non-trivial only when $\epsilon < 1/k$ and $\ln(\frac{k}{\epsilon}) \le \ln\frac{1}{\epsilon^2} = 2\ln\frac{1}{\epsilon}$ in this regime).

By Lemma 6, for every collection of k independent sets $I_1, \ldots, I_k$ in S, there is a value $t \le \frac{4}{\epsilon}$ for which the fingerprints $F_t(I_1), \ldots, F_t(I_k)$ define containers $C_t(I_1), \ldots, C_t(I_k)$ with a small union. And by Proposition 3, each of these containers satisfy $C_t(I_j) = C_t(F_t(I_j))$. Therefore, the argument above implies that the probability that the union of any k independent sets contains all of S, or equivalently that S is k-colorable, is at most $1/3$. □

REFERENCES

[AFKS00] Noga Alon, Eldar Fischer, Michael Krivelevich, and Mario Szegedy. Efficient testing of large graphs. *Combinatorica*, 20(4):451–476, 2000.

[AK02] Noga Alon and Michael Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Mathematics*, 15(2):211–227, 2002.

[BESS78] Béla Bollobás, Paul Erdős, Miklós Simonovits, and Endre Szemerédi. Extremal graphs without large forbidden subgraphs. *Annals of Discrete Mathematics*, 3:29–41, 1978.

[BL10] Andrej Bogdanov and Fan Li. A better tester for bipartiteness? arXiv preprint 1011.0531, November 2010.

[BMS15] József Balogh, Robert Morris, and Wojciech Samotij. Independent sets in hypergraphs. *Journal of the American Mathematical Society*, 28(3):669–709, 2015.

[BT04] Andrej Bogdanov and Luca Trevisan. Lower bounds for testing bipartiteness in dense graphs. In *Proceedings. of the 19th IEEE Annual Conference on Computational Complexity*, pages 75–81, 2004.

[BY22] Arnab Bhattacharyya and Yuichi Yoshida. *Property Testing: Problems and Techniques*. Springer Nature, 2022.

[FLS04] Uriel Feige, Michael Langberg, and Gideon Schechtman. Graphs with tiny vector chromatic numbers and huge chromatic numbers. *SIAM Journal on Computing*, 33(6):1338–1368, 2004.

[FS97] Uriel Feige and Michael Seltser. On the densest k-subgraph problem. Technical Report No. CS97-16, Weizmann Institute of Science, 1997.

[GGR98] Oded Goldreich, Shari Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998.

[Gol17] Oded Goldreich. *Introduction to property testing*. Cambridge University Press, 2017.

[GR10] Mira Gonen and Dana Ron. On the benefits of adaptivity in property testing of dense graphs. *Algorithmica*, 58(4):811–830, 2010.

[GR11] Oded Goldreich and Dana Ron. Algorithmic aspects of property testing in the dense graphs model. *SIAM Journal on Computing*, 40(2):376–445, 2011.

[GT03] Oded Goldreich and Luca Trevisan. Three theorems regarding testing graph properties. *Random Structures & Algorithms*, 23(1):23–57, 2003.

[GW21] Oded Goldreich and Avi Wigderson. Non-adaptive vs adaptive queries in the dense graph testing model. In *Proceedings of the 62nd IEEE Annual Symposium on Foundations of Computer Science*, pages 269–275, 2021.

[HMP21] Wasim Huleihel, Arya Mazumdar, and Soumyabrata Pal. Random subgraph detection using queries. *arXiv preprint arXiv:2110.00744*, 2021.

[JPP23] Matthew Jenssen, Will Perkins, and Aditya Potukuchi. Approximately counting independent sets in bipartite graphs via graph containers. *Random Structures & Algorithms*, 2023.

[KB80] Rob Kaas and Jan M Buhrman. Mean, median and mode in binomial distributions. *Statistica Neerlandica*, 34(1):13–18, 1980.

[KW82] Daniel J. Kleitman and Kenneth J. Winston. On the number of graphs without 4-cycles. *Discrete Mathematics*, 41(2):167–172, 1982.

[Man17] Pasin Manurangsi. Almost-polynomial ratio ETH-hardness of approximating densest k-subgraph. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 954–961, June 2017.

[Mul18] Wolfgang Mulzer. Five proofs of Chernoff's bound with applications. *Bulletin of EATCS*, 1(124), 2018.

[Neu70] Peter Neumann. Über den median einiger dikreter verteilungen und eine damit zusammenhängende monotone konvergenz. *Wissenschaftliche Zeitschrift der Technischen Universität Dresden*, 19:29–33, 1970.

[NR18] Yonatan Nakar and Dana Ron. On the testability of graph partition properties. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2018)*, page 13, 2018.

[RD85] Vojtěch Rödl and Richard A. Duke. On graphs with small subgraphs of large chromatic number. *Graphs and Combinatorics*, 1(1):91–96, 1985.

[RS19] Miklós Z. Rácz and Benjamin Schiffer. Finding a planted clique by adaptive probing. arXiv preprint 1903.12050, 2019.

[Sap05] Alexander Sapozhenko. Systems of containers and enumeration problems. In *Stochastic Algorithms: Foundations and Applications (SAGA 2005)*, pages 1–13. Springer, 2005.

[Soh12] Christian Sohler. Almost optimal canonical property testers for satisfiability. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, pages 541–550, 2012.

[ST15] David Saxton and Andrew Thomason. Hypergraph containers. *Inventiones mathematicae*, 201(3):925–992, 2015.

[Zam23] Or Zamir. Algorithmic applications of hypergraph and partition containers. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, 2023.