Three Theorems regarding Testing Graph Properties*

Oded Goldreich†
Department of Computer Science
Weizmann Institute of Science
Rehovot, ISRAEL.
oded@wisdom.weizmann.ac.il

Luca Trevisan
EECS – Computer Science Div.
UC-Berkeley
Berkeley, CA 94720, USA.
luca@eecs.berkeley.edu

December 9, 2001

Abstract

Property testing is a relaxation of decision problems in which it is required to distinguish YES-instances (i.e., objects having a predetermined property) from instances that are far from any YES-instance. We presents three theorems regarding testing graph properties in the adjacency matrix representation. More specifically, these theorems relate to the project of characterizing graph properties according to the complexity of testing them (in the adjacency matrix representation).

The first theorem is that there exist *monotone* graph properties in $\mathcal{NP}$ for which testing is very hard (i.e., requires to examine a constant fraction of the entries in the matrix). The second theorem is that every graph property that can be tested making a number of queries that is independent of the size of the graph, can be so tested by uniformly selecting a set of vertices and accepting iff the induced subgraph has some fixed graph property (which is not necessarily the same as the one being tested). The third theorem refers to the framework of *graph partition problems*, and is a characterization of the subclass of properties that can be tested using a one-sided error tester making a number of queries that is independent of the size of the graph.

Keywords: Property Testing, Graph Properties, Monotone Properties, NP,

*An extended abstract has appeared in the proceedings of the *42nd FOCS*, Las Vegas, USA, October 2001.
†Supported by the MINERVA Foundation, GERMANY.

Contents

1 Introduction

Property testing (cf., [10, 7]) is a natural notion of approximation for decision problems: For a predetermined property (or decision), the task is to distinguish whether a given instance has this property (i.e., is a YES-instance) or is "far" from any instance having the property.

This work is concerned with testing *graph properties* in the *adjacency matrix representation* (cf. [7]). A tester for a predetermined graph property Π is a (randomized) algorithm that is given a size parameter N and a distance parameter ϵ as well as oracle access to the adjacency matrix of an N-vertex graph G; that is, query $(u,v) \in [N] \times [N]$ is answered by a bit indicating whether or not the edge (u,v) is present in the graph. The algorithm is required to accept (with probability at least $2/3$) any graph having property Π, and reject (with probability at least $2/3$) any graph that is ϵ-far from having property Π, where distance between (N-vertex) graphs is defined as the fraction of edges (over $\binom{N}{2}$) on which the graphs differ.

Our focus is on the *query complexity* of testing some graph properties. We consider two extreme cases. In one case, the query complexity of testing depends only on the distance parameter ϵ (and is independent of the size of the graph N). In this case, we say that testing is very easy. In the other extreme case, for some constant $\epsilon > 0$, any tester must make $\Omega(N^2)$ queries (and is thus not significantly better than a trivial tester that inspect the entire graph). In this case, we say that testing is very hard.

1.1 Our main results

Our first main result (cf. Theorem 1) is that there exist *monotone* graph properties in $\mathcal{NP}$ for which testing is very hard (i.e., requires $\Omega(N^2)$ queries). This improves over an analogous result of Goldreich, Goldwasser and Ron [7, Prop. 10.2.3.2] that established the same lower bound for non-monotone graph properties (in $\mathcal{NP}$). In fact, our result is obtained by a simple extension of their technique. This resolves a natural open problem (raised by several researchers, and most recently by Y. Dodis).

Our second main result (cf. Theorem 2) refers to graph properties that can be tested very easily (i.e., using a number of queries taht only depends on the distance parameter ϵ). We show that such graph properties can be so tested by uniformly selecting a set of vertices of size depending only on ϵ and accepting if and only if the induced subgraph has some graph property (which is not necessarily the same as the one being tested). This improves over a previous result of Alon *et. al.* [2], who only assert that a tester may just inspect a random induced subgraph but do not assert that the decision may depend only on a property of that subgraph (rather than also on the tester's coins). Our result extend to any query complexity so that if the original tester had query complexity q then the new tester has query complexity that is polynomial in q. Furthermore, the transformation preserves one-sided error. It follows that the query complexity of testing graph properties is polynomially related to the query complexity of performing such testing via non-adaptive testers (and while preserving one-sided error). This improves over the naive transformation of adaptive testers to non-adaptive ones (which incurs an exponential blow-up in the query complexity).

Our third main result (cf. Theorem 3) refers to the framework of *graph partition problems* introduced by Goldreich, Goldwasser and Ron [7]. They showed that every problem Π in this framework can be tested very easily (i.e., by making $\text{poly}(1/\epsilon)$ queries). Within this framework, we characterize the subclass of properties that can be tested very easily using a *one-sided error* tester (i.e., the tester must accept any graph having the property with probability 1). Details follow.

A graph partition testing problem is parameterized by a sequence of corresponding pairs of lower and upper bounds. Specifically, for some (implicit) parameter k, the sequence contains k pairs of vertex-sets densities and $k + \binom{k}{2}$ pairs of "edge-densities". A graph has the specified (by the sequence) property if there exists a k-partition of its vertices such that the number of vertices in each component of the partition as well as the number of edges within each component and between each pair of components falls between the corresponding lower and upper bounds (in the sequence of parameters). Goldreich, Goldwasser and Ron [7] showed that every graph partition property (i.e., problem in the above framework) can be tested by making poly$(1/\epsilon)$ queries, but in general their tester has two-sided error probability. They also gave one-sided error testers for k-colorability (which operate by checking whether a random induced poly$(1/\epsilon)$-vertex subgraph is k-colorable). We show that the class of graph partition properties that admit a one-sided error tester of query complexity that is independent of N consists of two subclasses:

1. *The main subclass*: Each property in the subclass corresponds to a k-vertex graph H. An N-vertex graph has the property if its vertices can be k-partitioned such that there are no edges among vertices residing in the same part and so that there are edges between vertices of the i^{th} part and j^{th} part only if (i, j) is an edge of H. (For example, k-colorability is expressed by letting H be the k-vertex clique.)

2. *The non-interesting subclass* consists of two graph properties: the *clique property* and the *trivial property*. The only N-vertex graph satisfying the *clique property* is the N-vertex clique, whereas all (but finitely many) graphs satisfy the *trivial property*.

We note that each property in the above class can be tested with one-sided error by uniformly selecting a set of poly$(1/\epsilon)$ vertices and accepting if and only if the induced subgraph satisfies the very same graph property being tested.

1.2 Perspective

For a wider perspective on property testing see [6, 9]. Our results are related to the project of characterizing graph properties according to the complexity of testing them. This is a natural research project, alas a seemingly very difficult one (cf., [7, 2]). Our results carry good and bad news for this project.

Theorem 1 refutes the conjecture that all *monotone* graph properties can be tested very easily, a conjecture which could have been justified by the fact that (monotone) NP-hard problems such as k-colorability and ρ-clique can be tested very easily (cf. [7, Sec. 6.2&7]). Theorem 1 can be viewed as bad news for the "characterization project" (because it means that yet another natural class of properties has both easily-testable and hard-to-test properties).

Theorem 2 provides a tool for the study of graph properties that can be tested very easily. It asserts that when conducting such a study, one may focus on canonical testers that operate in a relatively simple way. The usefulness of Theorem 2 is demonstrated in our proof of Theorem 3, a proof that repeatedly refers to the fact that the canonical tester accepts if and only if a random induced subgraph has a certain graph property.

Theorem 3 is of the type of results sought after by the "characterization project", alas it refers only to a special class of graph properties (see above). Combined with previous results of [7, Sec. 9] and [2], this suggests that progress can be made with respect to subclasses of graph properties that can be expressed in some uniform structural formalism (rather than merely placed in a uniform complexity class such as $\mathcal{NP}$).

Organization

The abovementioned results are stated formally in Section 2, and their proofs appear in the subsequent sections (i.e., in Sections 3, 4 and 5, respectively).

2 Formal Setting

For any natural number n, we let $[n] \stackrel{\rm def}{=} \{1, ..., n\}$. Without loss of generality, all N-vertex graphs have $[N]$ as their vertex set, and their edges are unordered pairs over $[N]$.

2.1 Graph properties and distance to them

A graph property Π is a predicate defined over graphs that is preserved under graph isomorphism (i.e., if G has property Π and G' is isomorphic to G then G' has property Π).

We say that a graph $G = ([N], E)$ is ϵ-close to having property Π if there exists a graph $G' = ([N], E')$ having property Π such that the symmetric difference between E and E' is at most $\epsilon \cdot \binom{N}{2}$. We say that a graph G is ϵ-far from having property Π if it is NOT ϵ-close to having property Π. A useful observation follows:

Claim 2.1 *If G is ϵ-close to* (resp., ϵ-far from) *having property* Π *then so is any graph that is isomorphic to G.*

Proof: Suppose that $G = ([N], E)$ is ϵ-close to Π and $G' = \pi(G) \stackrel{\rm def}{=} ([N], \{(\pi(u), \pi(v)) : (u, v) \in E\})$ for some permutation $\pi : [N] \to [N]$. Let H be a graph having property Π such that the graphs G and H differ on at most $\epsilon \cdot \binom{N}{2}$ edges. Then, $H' = \pi(H)$ also has property Π and the graphs G' and H' differ on at most $\epsilon \cdot \binom{N}{2}$ edges. ■

2.2 Testers for graph properties

Testers are oracle machines that are given as input a pair (N, ϵ), where N is a size parameter and $\epsilon > 0$ is a distance parameter, as well as oracle access to (the adjacency matrix) of an N-vertex graph. An oracle machine T is called a tester for property Π if for every $G = ([N], E)$ and every ϵ, the following two conditions hold:

1. If G has property Π then $\Pr[T^G(N, \epsilon) = 1] \geq \frac{2}{3}$.
2. If G is ϵ-far from having property Π then $\Pr[T^G(N, \epsilon) = 1] \leq \frac{1}{3}$.

In both items, the probability space is that of the internal coin tosses of machine T, and a typical query (u, v) to oracle G is answered by 1 iff the edge (u, v) is in the graph G. The tester T (for Π) is said to have one-sided error if it always accepts graphs having the property Π; that is, for every $G = ([N], E)$ having the property Π and every ϵ, it holds that $\Pr[T^G(N, \epsilon) = 1] = 1$.

The query complexity a tester T is a function $q : \mathbb{N} \times [0, 1] \to \mathbb{N}$ such that $q(N, \epsilon)$ is an upper bound on the number of queries made by T on input (N, ϵ) and oracle access to the adjacency predicate of any N-vertex graph. The query complexity of a property Π is the minimum query complexity of testers for Π.

2.3 Statement of the Main Results

Existence of hard-to-test monotone graph properties (in $\mathcal{NP}$). A graph property Π is called monotone if adding any edge to any graph that has property Π results in a graph that has property Π. By saying that a graph property Π is in $\mathcal{NP}$, we mean the natural thing; that is, that the problem of deciding whether a given graph has property Π is in $\mathcal{NP}$ (i.e., the set Π is in $\mathcal{NP}$).

Theorem 1 *There exists a* monotone *graph property Π in $\mathcal{NP}$ such that the query complexity of Π, denoted q_Π, satisfies $q_\Pi(N, 0.1) = \Omega(N^2)$.*

Recall that $q_\Pi(N,\epsilon)$ is a lower bound on the number of queries made by any tester for Π on input (N,ϵ) and oracle access to (the adjacency predicate of) any N-vertex graph. Theorem 1 is proven in Section 3.

Canonical forms of testers for graph properties. Let Π be any graph property and T be a tester for Π. We say that T is canonical if, for some function $s : \mathsf{N} \times [0,1] \to \mathsf{N}$ and graph property Π', the tester operates as follows: on input (N,ϵ) and oracle access to any N-vertex graph G, the tester T selects uniformly a set of $s(N,\epsilon)$ vertices (in G), and accepts if and only if the corresponding induced subgraph (of G) has property Π'. Clearly, the query complexity of such a tester is $q(N,\epsilon) = \binom{s(N,\epsilon)}{2}$.

Theorem 2 *Let Π be any graph property having query complexity q_Π. Then Π has a canonical tester of query complexity $q'_\Pi(N,\epsilon) = O(q_\Pi(N,\epsilon)^4)$. Furthermore, if Π has a* one-sided error *tester of query complexity q, then Π has a canonical tester that has* one-sided error *and query complexity $\binom{2q}{2}$.*

In particular, it follows that in the context of testing graph properties, the query complexity of non-adaptive[1] algorithms is polynomially related to the query complexity of adaptive algorithms. Theorem 2 is proven in Section 4.

A characterization of graph partition properties that are easily testable with one-sided error. We refer to *graph partition properties* as described in the introduction and further discussed in Appendix B. We confine ourselves to "non-trivial" graph properties. That is, a graph property Π is non-trivial if for all infinitely many N's there exist an N-vertex graph satisfying property Π as well as an N-vertex graph not satisfying property Π. For a k-vertex graph H and a graph G, we say that G is H-embeddable if the vertices of G can be k-partitioned such that there are no edges (of G) among vertices residing in the same part and so that there are edges between vertices of the i^{th} part and j^{th} part only if (i,j) is an edge of H. (For example, saying that G is C_k-embeddable, where C_k denotes the k-vertex clique, is equivalent to saying that G is k-colorable.)

Theorem 3 *Let Π be an non-trivial graph partition property that is testable with one-sided error and query-complexity independent of N. Then exactly one of the following two cases holds:*

1. *There exists a k-vertex graph H such that, for every sufficiently large graph G, the graph G satisfies Π if and only if G is H-embeddable.*
2. *For all sufficiently large N, an N-vertex graph has property Π if and only if it is an N-vertex clique.*

Theorem 3 is proven in Section 5.

[1] An oracle machine is called non-adaptive if it determines its queries based merely on its input and random-coins, independently of the answers to prior queries.

3 Monotone Graph Properties may be Very Hard to Test

Throughout this section we consider the query complexity of testing, when setting the distance parameter to equal a constant (e.g., $\epsilon = 0.1$). In contrast, the size of the graph (denoted N) is treated as a parameter. Thus, when we describe a set of N-vertex graphs, it is to be understood that we consider the union of all these sets (i.e., over all possible N's).

3.1 Motivation

Goldreich, Goldwasser and Ron showed that there exist graph properties in $\mathcal{NP}$ for which any tester must inspect at least a constant fraction of the vertex-pairs [7, Prop. 10.2.3.2]. Their construction proceeds in two stages:

1. First, it is shown that certain sample spaces yield a collection of Boolean functions (i.e., a property of Boolean functions) that is hard to test (i.e., any tester must inspect at least a constant fraction of the function's values).

   On one hand, the sample space is relatively sparse (and thus a random function is far from any function in the resulting collection), but on the other hand it enjoys a strong pseudorandom feature (and so its projection on any constant fraction of the coordinates looks random). Thus, the functions in the class (which must be accepted with high probability) look random to any tester that inspect only a small constant fraction of the function's values, whereas random functions are far from the class (and should be rejected with high probability). This yields a contradiction to the existence of a tester that inspect only a small constant fraction of the function's values.

2. Next, the domain of the functions is associated with the set of unordered pairs of elements in $[N]$, and the collection of functions is "closed" under graph isomorphism (i.e., if a certain function on $\binom{N}{2}$ is in the collection then so is any function obtained from it by a relabeling of the elements of $[N]$).

   The closure operation makes this collection correspond to a graph property (since it is now preserved under isomorphism). The parameters are such that the resulting collection (although likely to be $N!$ times bigger than the original one) is still sparse enough (and so a random graph is far from it). On the other hand, the indistinguishability feature is maintained.

Unfortunately, the above construction does not yield a monotone property (since the second stage inherits the possible non-monotonicity of the collection constructed in the first stage). We redeem the situation by adding a third stage in which the collection is "closed" under edge-additions. Clearly, this guarantees that the resulting (graph) property is monotone (and it also inherits the feature of being in $\mathcal{NP}$). However, the resulting collection is no longer sparse, and so the fact that a random graph is far from it should be argued differently.

3.2 The actual construction

The actual construction follows the three stages sketched in the motivation above. For every N, we start by considering a sample space of $\binom{N}{2}$-bit long strings taken from an $0.1 \cdot 2^{-t}$-biased sample space (cf., Naor and Naor [8] or [3]), where $t \stackrel{\text{def}}{=} \frac{1}{1000}N^2$. Efficiently constructible sample spaces of size $O(\binom{N}{2}/0.1 \cdot 2^{-t})^2 = 2^{2t+o(t)}$ having the above feature can be found in [3]. We actually omit from the sample space any sample that has less than one third of one-entries. (The importance of this modification and its effect will be analyzed below.) For each sample s in the (residual) space, we define a graph $G_s = ([N], E_s)$ by letting $(i,j) \in E_s$ if and only if the $(i,j)^{\text{th}}$ bit of s equals 1, where we consider any fixed (efficiently computable) order of the elements in $\{(i,j) : 1 \le i < j \le N\}$. We call these $2^{2t+o(t)}$ graphs, the basic graphs. Note that the set of basic graphs is not likely to be closed under isomorphism, and thus this collection does not constitute a graph property. On the other hand, the set of basic graphs is in $\mathcal{NP}$, because elements in the sample space can be generated in time polynomial in N (i.e., there exists a poly(N)-time algorithm that given an $(2t + o(t))$-bit long string produces an $\binom{N}{2}$-bit long string in the sample space).[2]

Next, we consider the set of secondary graphs obtained by "closing" the set of basic graphs under isomorphism. That is, for every s in the sample space (equiv., a basic graph G_s) and every permutation π over $[N]$, we consider the secondary graph $G_{s,\pi} = ([N], E_{s,\pi})$ that is defined so that $(\pi(u), \pi(v)) \in E_{s,\pi}$ iff $(u,v) \in E_s$. By construction, the set of secondary graphs is closed under isomorphism, and so this collection does constitute a graph property. The latter set is also in $\mathcal{NP}$ (by incorporating the isomorphism π in the NP-witness).

Finally, we "close" the set of secondary graphs under edge-addition, obtaining our final set of graphs (which is, by construction, monotone): That is, for every secondary graph $G' = ([N], E')$, and every $E'' \supseteq E'$, we introduce the final graph $G'' = ([N], E'')$.

Comment. At the point, the reason for the modification in the initial sample space may be clear: If, for example, the sample space had contained the all-zero string then the set of final graphs would have contained all graphs, and testing membership in it would have been trivial.

Analysis. Our aim is to show that, although a random graph is far from the set of final graphs, no algorithm that makes $o(N^2)$ queries can distinguish a random graph from a graph selected among the final graphs (with some distribution that is not necessarily uniform). Since a tester for the set of final graphs must accept any final graph (with high probability) and reject a random graph (with high probability), we conclude that such tester must make $\Omega(N^2)$ queries. Specifically, throughout the rest of the analysis, we refer to testers of N-vertex graphs that should accept with probability at least 2/3 every graph that has the property, and reject with probability at least 2/3 every graph that is 0.1-far from having the property. Thus, we omit the distance parameter ϵ, which always equals 0.1, from all notations.

Claim 3.1 *The probability that a random graph is 0.1-close to some final graph is at most 0.1.*

Proof: The idea is to consider the "asymmetric distance" of a random graph from the set of secondary graphs, where the asymmetric distance of $G_1 = ([N], E_1)$ from $G_2 = ([N], E_2)$ is defined as $E_2 \setminus E_1$ (rather than the symmetric distance between the graphs, which equals $(E_2 \setminus E_1) \cup (E_1 \setminus E_2)$). We first bound the "asymmetric distance" of a random graph from the set of secondary graphs (using the fact that each secondary graph has many edges), and then observe that this asymmetric distance does not decrease when we replace the secondary graphs by final graphs. That is, unlike the symmetric distance, the asymmetric distance of G_1 from G_2 can only increase when adding edges to G_2.

Fixing any *secondary* graph, we first upper bound the probability that a random graph misses less than $0.1 \cdot \binom{N}{2}$ of the edges of the (fixed) secondary graph. Using the fact that the secondary

[2]In fact, using known constructions, membership in the (original) sample space can be decided in polynomial-time. Observing that the omitting condition is also decidable in polynomial-time, we conclude that the set of basic graphs is actually in $\mathcal{P}$.

graph contains at least $\frac{1}{3} \cdot \binom{N}{2}$ edges (see above), the expected number of edges missed by a random graph is at least $\frac{1}{6} \cdot \binom{N}{2}$. Since the "missing events" (for edges) are independent, we can employ Chernoff's bound and infer that the probability that less than $0.1 \cdot \binom{N}{2}$ edges are missed (by a random graph) is upper bounded by

$$\exp\left[-2 \cdot \left(\frac{1}{6} - 0.1\right)^2 \cdot \binom{N}{2}\right] \;<\; 2^{-0.006N^2} \tag{1}$$

Recall that the number of secondary graphs is $2^{2t+o(t)} \cdot (N!)$. Using $t = 0.001N^2$, this number is $O(2^{0.002N^2+o(N^2)})$. Employing the union bound, we conclude that the probability that a random graph misses less than $0.1 \cdot \binom{N}{2}$ of the edges of some secondary graph is at most

$$O(2^{0.002N^2+o(N^2)}) \cdot 2^{-0.006N^2} \;<\; 0.1 \tag{2}$$

Since each final graph contains some secondary graph as a subgraph, the probability that a random graph misses less than $0.1 \cdot \binom{N}{2}$ of the edges of some final graph is at most 0.1. The claim follows. ■

Claim 3.2 *Let M be a probabilistic oracle machine that makes at most t queries. Let R_N denote a random graph, and B_N denote a graph uniformly selected among the basic graphs. Then,*

$$|\mathsf{Pr}[M^{R_N}(N) = 1] - \mathsf{Pr}[M^{B_N}(N) = 1]| < 0.2$$

Proof: We identify $\binom{N}{2}$-bit long strings with N-vertex graphs (obtained as in the first stage of the construction). Let G_N denote a graph uniformly selected among all graphs in the sample space; that is, without discarding from the space those samples having less than one third of one-entries (equiv., less than $\frac{1}{3}\binom{N}{2}$ edges). Thus, B_N is obtained from G_N by conditioning that G_N has at least $\frac{1}{3}\binom{N}{2}$ edges. Using the fact that a small bias sample space (as above) is almost pairwise independent, the probability that an element in it has less than one third of one-entries is very small (e.g., tends to zero when $N \to \infty$). Thus, the probability that G_N has at least $\frac{1}{3}\binom{N}{2}$ edges is overwhelmingly high, and so the statistical difference between G_N and B_N is very small (e.g., smaller than 0.1). It follows that

$$|\mathsf{Pr}[M^{G_N}(N) = 1] - \mathsf{Pr}[M^{B_N}(N) = 1]| < 0.1 \tag{3}$$

On the other hand, since the sample space underlying the construction of G_N has bias at most $0.1 \cdot 2^{-t}$, it follows that, for any fixed sequence of coins for M, any fixed sequence of t answers occurs with probability $2^{-t} \pm 0.1 \cdot 2^{-t}$ under G_N (rather than with probability 2^{-t} under R_N).[3] Thus, for any fixed sequence of coins for M, the observed deviation of the t answers of G_N from the t answers of R_N is at most 0.1. It follows that

$$|\mathsf{Pr}[M^{G_N}(N) = 1] - \mathsf{Pr}[M^{R_N}(N) = 1]| < 0.1 \tag{4}$$

Combining Eq. (3) and (4), the claim follows. ■

Combining Claims 3.1 and 3.2, we obtain:

[3]We use the fact that fixing the internal coins of M and the oracle answers to M determines the oracle queries that M makes.

Theorem 3.3 (Theorem 1, restated): *There exists a monotone graph property in $\mathcal{NP}$ for which every tester requires $\Omega(N^2)$ queries* (even when invoked with constant distance parameter).

Proof: Consider the graph property, denoted Π, corresponding to the set of final graphs defined above. Recall that this set indeed corresponds to a monotone graph property in $\mathcal{NP}$. Now, suppose that M is a tester for this property and that M makes less than $N^2/1000$ queries (when invoked with distance parameter 0.1). Then, by Claim 3.2,

$$|\mathsf{Pr}[M^{R_N}(N)=1] - \mathsf{Pr}[M^{B_N}(N)=1]| \ < \ 0.2 \tag{5}$$

Now, since each graph in the support of B_N (i.e., each basic graph) has property Π, the tester must accept (i.e., output 1 on input) such graph with probability at least 2/3. It follows that

$$\mathsf{Pr}[M^{B_N}(N)=1] \geq \frac{2}{3} \ > \ 0.6 \tag{6}$$

On the other hand, the tester may accept with probability at most 1/3 each graph that is 0.1-far from having property Π. By Claim 3.1, the probability that R_N is 0.1-far from having property Π is at least 0.9. It follows that

$$\mathsf{Pr}[M^{R_N}(N)=1] \ \leq \ 0.9 \cdot \frac{1}{3} + 0.1 \cdot 1 \ = \ 0.4 \tag{7}$$

Combining Eq. (5)–(7), we reach a contradiction. The theorem follows. ■

4 Canonical Forms of Graph-Property Testers

We present two "canonization" transformations that can be applied to algorithms that test graph properties. The two "canonization" transformations are:

1. Transformation to testers that inspect a random induced subgraph.
2. Transformation to testers that decide according to whether the induced subgraph has some (possibly other) graph property.

Both transformations incur only a polynomial increase in the query complexity, and preserve one-sided error.

The first transformation improves over a similar transformation that appears in the work of Alon *et. al.* [2]: their transformation incurrs an exponential increase in the query complexity. We provide a description of their transformation in Appendix A. This is done both for sake of self-containment (and because the description in [2] is quite laconic) and as a warm-up towards the our own transformation (presented in Section 4.1).

4.1 Moving to testers that inspect a random induced subgraph

The following transformation improves over the one presented in Appendix A. Rather than incurring an exponential incease in the query complexity, it only incurs a polynomial increase. The transformation proceeds in a reverse order to the one utilized in Appendix A: First we move to testers that inspect an induced subgraph, and only next do we get rid of the potential adaptivity of the tester. In the latter argument, we capitalized on the fact that (by definition) graph properties are preserved under isomorphism (i.e., renaming of vertex names)

Lemma 4.1 *Let Π be any graph property, and T be an arbitrary tester for Π. Suppose that T has query complexity $q(N,\epsilon)$. Then there exists a tester for Π that selects a random subset of $2q(N,\epsilon)$ vertices, denoted R, makes the queries $\{(u,v) : u,v \in R\}$, and decides based on the oracle answers (and its internal coin tosses). Thus, the new tester is non-adaptive and its query complexity is less than $2q(N,\epsilon)^2$. Furthermore, if T has one-sided error then so does the new tester.*

Proof: First we transform T into an algorithm T' that belongs to the class of *vertex-uncovering* algorithms, defined as follows: A vertex-uncovering algorithm operates in iterations such that, in each iteration, depending on its coins and the answers obtained in previous iterations, the algorithm selects a new vertex, denoted v, and makes queries to all pairs (v,u), where u is a vertex selected in some prior iteration. Clearly, T can be emulated by a vertex-uncovering algorithm, denoted T', that makes at most $2q(N,\epsilon)$ iterations, and thus at most $\binom{2q(N,\epsilon)}{2} < 2q(N,\epsilon)^2$ queries: each query of T is emulated by two iterations of T', while these iterations are not necessarily new ones.

We next consider an algorithm T'' obtained from T' as follows. When given oracle access to a graph $G = ([N],E)$, algorithm T'' first selects uniformly a permutation π over $[N]$, and next invokes T' providing it with oracle access to the graph $\pi(G) \stackrel{\rm def}{=} ([N],\pi(E))$, where $\pi(E) \stackrel{\rm def}{=} \{(\pi(u),\pi(v)) : (u,v) \in E\}$. That is, when T' makes query (u,v), algorithm T'' makes query $(\pi(u),\pi(v))$ and feeds the answer to T'. Clearly, algorithm T'' also operates in a vertex-uncovering manner. Below, we will show that the set of vertices selected by T'' is uniformly distributed, and thus that its choice of vertices is actually oblivious of previous answers. But before doing so, we show that T'' maintains the testing features of T' (and thus of T):

- Let $G = ([N],E)$ be a graph having property Π. Then, for any permutation π, it is the case that the graph $\pi(G) \stackrel{\rm def}{=} ([N],\{(\pi(u),\pi(v)) : (u,v) \in E)$ has property Π. Thus, T' must accept the graph $\pi(G)$ with probability at least $2/3$. This means that (for every permutation π) conditioned on π being chosen in the onset (of T''), algorithm of T'' accepts the graph G with probability at least $2/3$ (because, in this case, T'' just emulates for T' an oracle access to the graph $\pi(G)$). Since this holds for every π, it follows that T'' accepts G with probability at least $2/3$.

- Suppose that a graph $G = ([N],E)$ is ϵ-far from having property Π. Then, for any permutation π, the graph $\pi(G)$ is ϵ-far from having property Π. (See Claim 2.1.) The rest of the argument follows analogously to the above (where here we refer to an upper bound on the accepting probability). It follows that T'' accepts G with probability at most $1/3$.

As stated above, the tester T'' operates in a vertex-uncovering manner. We now show that the next vertex selected in each of its operations is uniformly distributed (among all possible choices). In fact, we prove that for any choice r of a random-tape for T', for any possible sequence of queries and answers corresponding to the first i iterations of T'', the next vertex selected by T'' is uniformly distributed among all vertices not selected so far. (Recall that a generic random-tape of T'' is a pair (r,π), where r is a possible random-tape of T' and π is a permutation over $[N]$.) Let T'_r denote the deterministic oracle machine derived from T' by fixing the random-tape of T' to equal r. Similarly, let us denote by T''_r the machine derived from T'' by fixing the first part of the random-tape of T'' to equal r (i.e., given oracle access to G, the machine T''_r uniformly selects a permutation π, and invokes T'_r providing it with oracle access to $\pi(G)$).

Claim 1: For any integer i, random-tape r and any possible sequence of $\binom{i}{2}$ answers corresponding to the first i iterations of T''_r, the next vertex selected by T''_r is uniformly distributed among all vertices not selected so far. Furthermore, this holds even when given values $v_1, ..., v_i$ and conditioning on v_j being the j^{th} vertex uncovered by T''_r.

Proof: Recall that in these i iterations T''_r only inspects the subgraph induces by the i vertices it has uncovered. Let us fix a possible sequence of answers, denoted α. That is, α is a binary sequence of $\binom{i}{2}$ answers given to T''_r (and passed to T'_r) during these i iterations. (Indeed, α encodes a binary symmetric relation over $[i]$.) Recall that the (first $i+1$) vertices selected by T'_r, denoted $u_1^{r,\alpha}, ..., u_i^{r,\alpha}, u_{i+1}^{r,\alpha}$, are determined by T'_r depending on the corresponding prefix of α. The corresponding vertices that are actually uncovered by T''_r are $\pi(u_1^{r,\alpha}), ..., \pi(u_i^{r,\alpha}), \pi(u_{i+1}^{r,\alpha})$, where π is a random permutation selected at the onset of T''_r. Thus, $u_{i+1}^{r,\alpha}$ is determined by T'_r depending only on (r and) α, and at this point π is conditioned only by α being the subgraph induced by $\pi(u_1^{r,\alpha}), ..., \pi(u_i^{r,\alpha})$. Furthermore, if we condition on $\pi(u_j^{r,\alpha}) = v_j$ for $j = 1, ..., i$, then the condition on α becomes irrelevant (because it relates to the subgraph induced by the fixed $v_1, ..., v_i$). But conditioning on $\pi(u_j^{r,\alpha}) = v_j$ for $j = 1, ..., i$, the value of $\pi(u_{i+1}^{r,\alpha})$ is uniformly distributed in $[N] \setminus \{v_j\}_{j=1}^{i}$, and the claim follows. □

Loosely speaking, Claim 1 asserts that the vertices selected by T'' are selected with distribution that is independent of previous answers obtained by T''. It follows that we may select the sequence of vertices beforehand, which means that T'' is essentially non-adaptive. Formally, we consider an algorithm T''' that, given oracle access to a graph $G = ([N], E)$, uniformly selects a random-tape r for T'_r and invokes T'_r answering its queries as follows. In the $i+1^{\text{st}}$ iteration, when T'_r selects a new vertex (denoted u_{i+1}), algorithm T''' uniformly selects $v_{i+1} \in [N] \setminus \{v_j\}_{j=1}^{i}$, and answers the queries of the i^{th} iteration accordingly (i.e., according to the edges between v_{i+1} and the previous v_j's).

Claim 2: For any graph G, when given oracle access G, the output distribution of T''' is indentical to that of T''.

Proof: Fixing G and r, we show that the distribution of answers seen by T'''_r is indentical to that of T''_r, where T'''_r is defined analogously to T''_r. We prove, by induction on i, that the i^{th} vertex selected by each of these machines (i.e., T''_r and T'''_r) is uniformly distributed among all vertices not selected so far, and the claim follows. The induction step is obvious in case of T'''_r. Using Claim 1, the induction step holds with respect to T''_r: fixing any selection of vertices for the first i iterations of T''_r, the induced subgraph seen by T''_r is fixed too, and so the next vertex selected by T''_r is uniformly distributed among all vertices not selected so far. □

Combining Claim 2 with the fact that T'' is a tester for Π, we conclude that T''' is a tester for Π. Observing that T''' operates as claimed in the lemma, and has one-sided error in case T has one-sided error, we are done. ■

Perspective: The main part of the proof of Lemma 4.1 is similar to an analogous statement proven by Bar-Yossef *et. al.* [5] in the context of "sampling algorithms". Consider, for example, the problem of approximating the average value of a function f defined over a huge space, say $f : \{0,1\}^n \to [0,1]$, when given only oracle access to the function. We seek algorithms that make relatively few oracle calls, and call them samplers. The proof of Lemma 4.1 can be easily modified (and in fact simplified) to prove that *the query complexity of non-adaptive samplers equals the query complexity of adaptive ones.* The key observation is that also here the relevant property (i.e., the average value of a function) is invariant under renaming (of the function's arguments). In fact, this is exactly the way this statement (regarding query complexity of "sampling algorithms") is proven in Lemma 9 of [5].

4.2 Moving to a decision determined by the induced subgraph

The current transformation is less generic than the previous one, since it applies only to non-adaptive (graph property) testers. For simplicity, our starting point is actually testers as resulting from Lemma 4.1 (i.e., that query the oracle on the edges of a random induced subgraph). The current transformation consists of three steps:

1. Transformation to testers that decide based on the subgraph they see, possiblly by tossing new coins, but independently of the coins used to select the sample set of vertices. In particular, if the tester selects t vertices, then it decides based only on the induced t-vertex subgraph in which the vertices are labelled by the elements of $[t]$ according to some canonical order.
2. Transformation to testers that decide as above, but do so in a way that is independent of the labelling of the induced subgraph. That is, the decision depends only on an unlabelled version of the induced subgraph.
3. Transformation to testers that decide according to whether or not the induced subgraph has some fixed graph property. That is, this transformation gets rid of the coins that were possibly used in the previous decision process.

That is, the first transformation gets rid of the possible dependence of the decision on the identity of the sampled vertices; that is, the resulting decision depends only on the subgraph seen by the tester (but possibly depends on the ordering of the vertices in this subgraph). The second transformation makes this decision identical for all isomorphic subgraphs (i.e., independent of this ordering), and the third transformation makes this decision deterministic. The (proof of the) last transformation seems to be the most interesting one.

4.2.1 Moving to a sample-independent decision

We first move to testers in which the sample of vertices only determines the queries, but plays no direct role in the final decision (which depends only on the answers to these queries).

Claim 4.2 *Let Π be any graph property, and T be a non-adaptive tester for Π that selects a random subset of $t = t(N, \epsilon)$ vertices, makes queries to determine the induced subgraph, and decides based on the oracle answers (and its internal coin tosses). Then, without loss of generality, the tester T can be decomposed into two parts.*

- *The first part uniformly selects a set of vertices and queries the oracle for the induced subgraph, which it passes to the second part.*
- *The second part makes a decision based on the subgraph obtained from the first part and possibly depending on its own coins, but independent of the coins used by the first part.*

Furthermore, if T has one-sided error then no coins are tossed in the second part.

Proof: Without loss of generality, we may decompose the random-tape of T into the form (S, r), where S is the vertex set selected by T (with probability $p_S = 1/\binom{N}{t}$) and r is the residual randomness. (This step is quite generic: the original random-tape of T induces uniform distribution over the possible vertex-sets, and each possible vertex-set induces uniform distribution over all random-tapes that cause T to select this set.)

For each set of vertices S and each sequence of possible answers α (which is a symmetric relation over S), we denote by $q_{S,\alpha}$ the probability that T accepts, having selected the vertex set S and seeing the answer sequence α. Indeed, $q_{S,\alpha}$ is merely the fraction of r's that make T accept when its sees answers α (after selecting S). For any fixed α, let $q_\alpha \stackrel{\rm def}{=} \sum_S p_S q_{S,\alpha}$ be the expected value of $q_{S,\alpha}$ (for varying S). We now derive a tester that selects S as T does, but decides only acccording to the answers it gets (i.e., independently of S). Specifically, we consider an algorithm T' that selects S with probability p_S, and (for every α) upon obtaining the answer sequence α accepts with probability q_α.

Claim: For any graph G, when given oracle access G, the probability that T' accepts G equals the probability that T accepts a random isomorphic copy of G.

Proof: Let π be a uniformly distributed permutation over $[N]$, and consider the operation of T when given oracle access to $\pi(G)$. Note that T selects S uniformly, inspects the subgraph of $\pi(G)$ induced by S (which equals the subgraph of G induced by $\pi^{-1}(S)$), and accepts with probability $q_{S,\alpha}$, where α is the relation representing this subgraph. In terms of G, the situation is the same as if T would have selected uniformly and independently two sets S and S' (since S' corresponds to $\pi^{-1}(S)$ where π is a uniformly distributed permutation) and accepts with probability $q_{S,\alpha}$, where α is the relation representing the subgraph of G induced by S'. Taking another step, it is as if T would have selected uniformly a set S' and, upon obtaining the answer sequence α, accepts with probability $q_\alpha = \mathsf{E}_S(q_{S,\alpha})$ (becuase S is selected independently of S' and is only used to determine $q_{S,\alpha}$). But this exactly what T' actually does when given oracle access to G. $\Box$

As in the proof of Lemma 4.1, it follows that T' is a tester for Π. The main part of the claim follows.

Finaly we observe that if T has one-sided error then, without loss of generality, all the q_α's may be in $\{0,1\}$ (and so no coins are needed for the final decision). The reason is that if any q_α is strictly smaller than 1 then the answer sequence α cannot occur when accessing a graph that has property Π (or else this graph is rejected with non-zero probability), and so setting this q_α to zero does not effect the performance on graphs having property Π and may only improve the performance on all other graphs. $\blacksquare$

4.2.2 Moving to a isomorphism-oblivious decision

Claim 4.2 asserts that, without loss of generality, the final decision of a tester is obtained by a randomized computation that depends only on the oracle answers (which are obtained by querying all pairs in a random set). We now show that, without loss of generality, this decision is closed under isomorphism. In the special case of one-sided testers, this means that the final decision is by whether or not the induced subgraph satisfies some fixed graph property.

Claim 4.3 *Let Π and T be as in Claim 4.2. Then, without loss of generality, T is composed of two parts as in Claim 4.2, and in the second part the decision applied to the answer is closed under graph isomorphism. That is, when seeing an induced subgraph that equals H, the second part decides exactly as in case it sees a subgraph that it isomorphic to H. Furthermore, if T has one-sided error then, without loss of generality, there exists a graph property Π' such that the final decision of T amounts to checking whether or not the subgraph induced by its vertex sample has property Π'.*

Proof sketch: The proof is very similar to the proof of Claim 4.2. Denoting by q_α the probability that T accepts on answer sequence α, we consider an algorithm T' that accepts with probability that depends only on the class of graphs isomorphic to α. Specifically, let $g(\alpha)$ be the set of graphs that are isomorphic to the graph having an adjecancy matrix that corresponds to α, and let q'_H be the expected value of q_α taken uniformly over all α such that $g(\alpha) \ni H$. Then, T' selects uniformly a set of vertices S, and accepts with probability q'_H, where H is the subgraph induced by S.

As in the proof of Claim 4.2, we show that, for any graph G, the probability that T' accepts G equals the probability that T accepts a random isomorphic copy of G. (Here we use the fact that when T accesses a random isomorphic copy of G and obtains the answer sequence α, it decides as if it has accessed G and obtained an answer that corresponds to a random element in $g(\alpha)$.) Finally, we use again the fact that if T is one-sided then, without loss of generality, each q_α is either 0 or 1. The claim follows. ■

4.2.3 Moving to a deterministic decision

Here, our aim is to make the second part of the tester guaranteed by Claim 4.3 be deterministic (i.e., be determined by whether or not the induced subgraph has some fixed graph property). By the furthermore parts of Claims 4.2 and 4.3, it suffices to consider two-sided error testers (becuase for one-sided error testers the final decision was already shown to be deterministic). Thus, our focus in this subsection is on two-sided error testers. In fact, we can directly handle testers as in Claim 4.2 (i.e., with a decision that is not closed under isomorphism of the induced subgraph), but for sake of simplicity we prefer to use Claim 4.3 as our starting point.

Lemma 4.4 *Let Π and T be as in Claim 4.3, and let t denote the number of vertices selected by T. Then, there exists a tester T' and a graph property Π' so that T' selects a random set of $t' = O(t^2)$ vertices and accepts if and only if the subgraph induced by it has property Π'.*

Proof: Let us denote by p_H the probability that T accepts a graph when seeing H as the induced (t-vertex) subgraph. Consider a new algorithm T' that when given oracle access to a graph G, selects at random a set of $t' = 108t^2$ vertices, denoted S', inspects the subgraph induced by S' and decides as follows:

1. For every t-subset S of S', the new test determines the subgraph of G induced by S, and add p_{G_S} to an accumulated sum, where G_S denotes the subgraph of G induced by S. That is, T' computes the sum
$$\sum_{S \subset S' \text{ s.t. } |S|=t} p_{G_S} \tag{8}$$
2. Accept the graph G if and only if the final sum (i.e., Eq. (8)) is greater that $\frac{1}{2} \cdot \binom{t'}{t}$.

Observe that T' makes a final decision of the required form (i.e., the decision is isomorphism-oblivious and deterministic, and so it can be casted as a graph property). All that is left is to analyze the performance of T'. Intuitively, the expected value associated with each t-subset (of a random t'-subset) reflects the expected decision of T (i.e., its accepting probability), and since the various random values are "sufficiently independent" with high probability their sum reflects the expected decision of T. Thus, the sum is very likely to exceed $\frac{1}{2} \cdot \binom{t'}{t}$ (resp., be below $\frac{1}{2} \cdot \binom{t'}{t}$) if the accepting probability of T is above 2/3 (resp., below 1/3). In the actual analysis we will use Chebyshev's inequality and the fact that two random t-subsets of a t'-set are disjoint with sufficiently large probability. Details follows.

In order to simplify the analysis, we slightly modify the presentation. The modified T (resp., T') operates by selecting uniformly and independently a sequence of t (resp., t') vertices, possibly with repetitions. Assuming that $t(N, \epsilon) < N^{1/5}$, this modification has negligible impact, because repetitions are unlikely to occur. We stress that this modification is not essential to the rest of the analysis; it just simplifies things a little (and, in general, we may assume that $t(N, \epsilon) < \sqrt{N \log N}$, since otherwise algorithm T' sees the entire graph and may just decide accordingly).

Let us denote by $\overline{s} = (s_1, ..., s_{t'})$ the sequence of vertices selected by T'. We view T' as scanning all possible t-subsets $I \subset [t']$, and inspecting (for each such subset I) the subgraph induced by the vertex set $\{s_i : i \in I\}$. For each t-subset $I \subset [t']$, consider a random variable $\zeta_I = \zeta_I(\overline{s})$ representing the contribution of the set I to the accumulated sum (computed by T'), where the underlying proability space is of all possible t'-long vertex sequences $\overline{s}$. Clearly, $\mathsf{E}(\zeta_I) = \mathsf{E}_{\overline{s}}(p_{G_{\{s_i : i \in I\}}})$ equals $\mathsf{E}_{\overline{s}}(p_{G_{\{s_1,...,s_t\}}})$, which in turn equals the probability that T accepts G. Also, for every disjoint I and J, the random variables ζ_I and ζ_J are independent. Let $\mathcal{S}$ denote the set of all t-subsets of $[t']$, and recall that T' accepts G if and only if $\sum_{I \in \mathcal{S}} \zeta_I > \frac{1}{2} \cdot |\mathcal{S}|$. It follows that T' is correct provided that

$$\left| \sum_{I \in \mathcal{S}} \zeta_I - \sum_{I \in \mathcal{S}} \mathsf{E}(\zeta_I) \right| < \frac{1}{6} \cdot |\mathcal{S}| \tag{9}$$

Applying Chebyshev's inequality we will show that Eq. (9) holds with probability at least 2/3. Specifically, let $\overline{\zeta}_I \stackrel{\text{def}}{=} \zeta_I - \mathsf{E}(\zeta_I)$. Then

$$\begin{aligned}
\mathsf{Pr}\left[\left| \sum_{I \in \mathcal{S}} \zeta_I - \sum_{I \in \mathcal{S}} \mathsf{E}(\zeta_I) \right| \geq \frac{1}{6} \cdot |\mathcal{S}| \right] &\leq \frac{\mathsf{Var}(\sum_{I \in \mathcal{S}} \zeta_I)}{|\mathcal{S}|^2/36} \\
&= \frac{36}{|\mathcal{S}|^2} \cdot \mathsf{E}\left(\left(\sum_{I \in \mathcal{S}} \overline{\zeta}_I \right)^2 \right) \\
&= \frac{36}{|\mathcal{S}|^2} \cdot \sum_{I,J \in \mathcal{S}} \mathsf{E}(\overline{\zeta}_I \overline{\zeta}_J) \\
&< \frac{36}{|\mathcal{S}|^2} \cdot \frac{|\mathcal{S}|^2}{108} = \frac{1}{3}
\end{aligned}$$

where the last inequality uses the following facts:

- For any disjoint I and J, it holds that $\mathsf{E}(\overline{\zeta}_I \overline{\zeta}_J) = \mathsf{E}(\overline{\zeta}_I) \cdot \mathsf{E}(\overline{\zeta}_J) = 0$.
- For any (non-disjoint) I and J, it holds that $\mathsf{E}(\overline{\zeta}_I \overline{\zeta}_J) < 1$ (because $|\overline{\zeta}_K| < 1$ for every K).
- The fraction of non-disjoint pairs of t-subsets of $[t']$ among all pairs of t-subsets of $[t']$ is bounded above by $t^2 \cdot \frac{1}{t'} = \frac{1}{108}$.

It follows that T' decides correctly with probability at least 2/3, and the lemma follows. ■

4.2.4 Question: Does Π equal Π'

In general, the question does not make sense since Π is a property of N-vertex graphs, whereas Π' is a property of t-vertex graphs.[4] But the question does make sense if these properties can be expressed in a uniform way that is independent of the size of the graph. Two specific frameworks for such properties were presented in [7] and [2], respectively:

1. The framework of graph partition problems [7, Sec. 9] is briefly reviewed in the introduction (see also Appendix B), and is further studied in Section 5. Goldreich, Goldwasser and Ron [7] showed that every problem Π in this framework can be tested by checking whether a random induced subgraph of poly$(1/\epsilon)$ vertices satisfies a related graph property Π'. In general, Π' is not equal to Π, and the tester has two-sided error probability. (In fact, this is unavoidable

[4] Formally, both Π and Π' are properties of all graphs, but when we fix an N for the above discussion, we actually care of the properties Π and Π' when restricted to N-vertex and t-vertex graphs, respectively.

for some properties Π; for example, the property of the graph consisting of two equal size parts one being a clique and the other being an independent set.) Confining ourselves to properties Π that admit a one-sided error tester of query complexity independent of N, we show that without loss of generality Π' equals Π (see Corollary 5.9). This follows from the characterization of properties Π that admit a one-sided error tester of query complexity independent of N (see Theorem 5.8).

2. Alon *et. al.* [2] considered the class of graph properties that can be expressed by quantified boolean formula over the edge relation E. For example, the property of being triangle-free is represented by the formula

$$\forall x, y, z \, [(x,y) \notin E \vee (y,z) \notin E \vee (z,x) \notin E]$$

   For graph properties Π expressible by formulae of the form $\exists\forall$, Alon *et. al.* [2] presented a tester of query complexity that only depends on ϵ (but grows very rapidly with $1/\epsilon$).[5] Their tester checks whether a random induced subgraph of suitable size satisfies a related graph property Π'. Actually, they showed that for every such property Π can be approximated[6] by a "graph coloring" property Π' that can be tested by checking whether or not a random induced subgraph of suitable size satisfies Π' itself. Thus, the class of general "graph coloring" properties can be tested by checking whether a random induced subgraph of suitable size satisfies the very same property.

In addition, as observed by Alon [1], for any property Π that is closed under taking induced subgraphs (i.e., if G has property Π then so does every induced subgraph of G), if Π is easily testable then it is easily testable by a canonical tester (as above) that uses $\Pi' = \Pi$. That is, if such a property Π is testable by $q(\epsilon)$ queries (i.e., independent of the size of the graph) then it so testable by inspecting a random induced subgraph of size $\text{poly}(q(\epsilon))$ and accepting iff the subgraph has property Π. For details, see Appendix E.

4.3 Putting it together

Combining Lemma 4.1, Claim 4.2, Claim 4.3 and Lemma 4.4, we obtain:

Theorem 4.5 (Theorem 2, restated): *Let Π be any graph property. If there exists a tester with query complexity $q(N,\epsilon)$ for Π then there exists a tester for Π that uniformly selects a set of $O(q(N,\epsilon)^2)$ vertices and accepts iff the induced subgraph has property Π', where Π' is some fixed graph property. Furthermore, if the original tester has one-sided error then so does the new tester, and furthermore a sample of $2q(N,\epsilon)$ vertices suffices*

5 On General Graph Partition Problems that are Testable with One-Sided Error

We refer to the framework of graph partition problems [7, Sec. 9]. Recall that in this framework a testing problem is parameterized by a sequence of corresponding pairs of lower and upper bounds:

[5] For properties expressible by formulae having only universal quantifiers, the query complexity is a tower of $\text{poly}(1/\epsilon)$-many exponents. For properties expressible by formulae of the form $\exists\forall$, the query complexity is even worse (but still depends only on $1/\epsilon$).

[6] Loosely speaking, every graph having one of these properties is close to a graph having the other property.

For some (implicit) parameter k, the sequence contains k pairs of vertex-sets densities and $k + \binom{k}{2}$ pairs of edge-densities, and the problem is to determine whether there exists a k-partition of the vertices so that the number of vertices in each component of the partition as well as the number of edges within each component and between each pair of components falls between the corresponding lower and upper bounds. If such a partition exists, we call it a witness partition. For example, k-colorability falls into this framework by requiring that the density of edges within each of the k parts equals zero (and making no other requirements).[7] For further details see Appendix B.

Goldreich, Goldwasser and Ron [7] showed that every graph partition property (i.e., problem in the above framework) can be tested by making $\text{poly}(1/\epsilon)$ queries, but in general the tester has two-sided error probability. They also gave one-sided error testers for k-colorability (which operate by checking whether a random induced $\text{poly}(1/\epsilon)$-vertex subgraph is k-colorable).[8] Our main goal in this section is to characterize the set of graph partition properties that admit a one-sided error tester of query complexity that is independent of N.

A few technicalities: Throughout the discussion, we consider only admissible sequences of parameters (cf. [7, Def. 9.3.1]): These are sequences for which the set of graphs that have the property is infinite (i.e., contains at least one N-vertex graph, for infinitely many N's). We avoid integrality problems by allowing upto $k-1$ vertices to be split between the k parts of the partition (see [7, Rem. 9.1]). Also, following [7], we consider vertex-densities as fractions of N, and edge-densities as fractions of N^2 (rather than of $\binom{N}{2}$). Finally, for greater expressibility, we allow self-loops and count them as half edges: For example, using these conventions, a ρN-vertex clique in an N-vertex graph has edge density $\frac{\rho^2}{2}$, which is independent of N. (Note that the latter convention is not consistent with the exposition in the previous sections.) For further discussion of the latter two conventions, see Appendix B. We comment that the analysis can be carried out also under the alternative conventions mentioned above (and in Appendix B), but the exposition would be more cumbersome and the final result would be slightly different: the second case in Theorem 5.8 would not be possible.

The starting point: Using Theorem 4.5, we may confine ourselves to canonical testers (of one-sided error) that operate by inspecting the subgraph induced by a uniformly selected set of vertices, where the size of the vertex set is independent of N. Recall that the inspection (of the subgraph) consists of determining whether or not it has some graph property Π' (not necessarily equal to the property Π being tested).

5.1 Trivial graph properties

A graph property is called trivial if for every $\epsilon > 0$ and for all sufficiently large N, every N-vertex graph is ϵ-close to having the property. We may discard trivial graph properties from our discussion, since they have a "trivial" tester, which (provided N is big enough) accepts all N-vertex graphs. Furthermore, within the framework of graph partition problems, trivial properties are satisfied by all but finitely many graphs.

---

[7]Formally, the k lower-bound and upper-bound pairs on vertex-set sizes are all trivial (i.e., all equal $(0, 1)$), and so are the $\binom{k}{2}$ pairs of bounds on of edge-densities between pairs of components. The only non-trivial pairs of bounds are those referring to edge-densities within each of the k components (i.e., all these pairs equal $(0, 0)$).

[8]An improved bound on the size of the sample was later presented by Alon and Krivelevich [4]: They showed that a sample of size $\tilde{O}(1/\epsilon)$ (rather than $\tilde{O}(1/\epsilon^2)$) suffices for $k = 2$, and size $\tilde{O}(1/\epsilon^2)$ (rather than $\tilde{O}(1/\epsilon^3)$) suffices for any constant $k > 2$.

Lemma 5.1 *Let* Π *be any trivial* graph partition property. *Then, for all sufficiently large* N*, every* N*-vertex graph has property* Π.

The proof can be found in Appendix C.

5.2 Some graph partition properties that are trivial

We first identify (and discard from the rest of the discussion) a class of graph partition properties that contains only trivial properties.

Claim 5.2 *Let* Π *be a graph partition property that is testable by a canonical tester with one-sided error and query-complexity independent of* N*. Suppose that the graph* $G = ([N], E)$ *has property* Π*, and let* $(V_1, ..., V_k)$ *be a witness partition of* G*. If for some* i *the subgraph induced by* V_i *is neither a clique nor an independent set, then* Π *is trivial.*

Proof: Let $G = ([N], E)$, $(V_1, ..., V_k)$ and i be as in the hypothesis. Then the number of edges with both endpoints in V_i is greater than zero and smaller than $\frac{|V_i|^2}{2}$. Suppose for a moment that for some integer t the said number were at least $\frac{t^2}{2}$ and at most $\frac{|V_i|^2}{2} - \frac{t^2}{2}$. Then, for every t-vertex graph H, there would exist a graph G_H having property Π such that G_H would contain H as an induced subgraph. (The graph G_H is derived from G by modifying the subgraph induced by V_i so that the number of edges is maintained and the subgraph induced by the first t vertices of V_i equals H. This is certainly possible, because both the number of edges and non-edges in H is at most $\frac{t^2}{2}$)

Now, suppose that (for some $\epsilon > 0$) the canonical tester selects a sample of $t = t(\epsilon)$ vertices. Then for every t-vertex graph H there exists an N-vertex graph G_H satisfying Π (as above) so that when given oracle access to G_H, with positive probability, the tester sees H as the induced subgraph. Since G_H satisfies Π and the tester has one-sided error, it must accept upon seeing H as the induced subgraph. But this holds for every t-vertex graph H, and so the tester must always accept (no matter which induced subgraph it sees). It follows that, for every $N' \geq t$, the tester accepts (with probability $1 > \frac{2}{3}$) any N'-vertex graph, and thus every N'-vertex graph is ϵ-close to having property Π. If we can repeat the above argument for every $\epsilon > 0$ then it will follow that Π is trivial. To do so we must show that for every t, which will be set to equal the size of the vertex-sample selected by the tester on distance parameter ϵ, there exist a graph and a k-partition as in the claim's hypothesis so that the number of edges with both endpoints in V_i is at least $\frac{t^2}{2}$ and at most $\frac{|V_i|^2}{2} - \frac{t^2}{2}$ (rather than just greater than zero and smaller than $\frac{|V_i|^2}{2}$).

Fixing t, we first consider a graph $G = ([N], E)$ with $(V_1, ..., V_k)$ and i be as in the claim's hypothesis. Let e denote the number of edges with both endpoints in V_i (i.e., $0 < e < \frac{|V_i|^2}{2}$). If $\frac{t^2}{2} < e < \frac{|V_i|^2}{2} - \frac{t^2}{2}$ then we are done. Otherwise, for $f = \frac{t^2}{2}$, we consider the graph $G' = ([f] \times [N], E')$, where

$$E' = \{(\langle i, u\rangle, \langle j, v\rangle) : (u, v) \in E \ \wedge \ i, j \in [f]\} \tag{10}$$

Clearly, the graph G' has property Π, as is witnessed by the partition $(V'_1, ..., V'_k)$, where $V'_j \stackrel{\text{def}}{=} \{\langle i, v\rangle : v \in V_j \wedge i \in [f]\}$ (which preserves the relative densities of vertices and edges with respect to the witness partition $(V_1, ..., V_k)$). Furthermore, the number of edges in E' with both endpoints in V'_i is at least $f = \frac{t^2}{2}$ and at most $\frac{|V'_i|^2}{2} - \frac{t^2}{2}$. The claim follows. ■

5.3 Non-trivial graph partition properties – the two cases

Below, we refer to the lower and upper bound parameters that appear in the specification of property Π. Recall that there are bounds that refer to the fraction of vertices in each part, and bounds referring to the fraction of edges inside parts or between parts.

Claim 5.3 *Let* Π, $G = ([N], E)$ *and* $(V_1, ..., V_k)$ *be as in Claim 5.2.*

1. *If for some* i *the subgraph induced by* V_i *is an independent set then no edge lower-bound parameter is positive.*

2. *Suppose that no edge lower-bound parameter is positive. Then, if for some* i *the subgraph induced by* V_i *is a clique then* Π *is trivial.*

Proof: We start with Part (1). As in the proof of Claim 5.2, we may assume that the size of V_i is greater than the size of the vertex-sample chosen by the tester. It follows that when given oracle G, with positive probability, the tester will see an induced subgraph that is an independent set. Because the tester has one-sided error, it must accept in this case, and thus it always accepts an oracle N-vertex graph that is an independent set. Repeating the argument for any $\epsilon > 0$, it follows that, for sufficiently large N, the N-vertex independent set graph is ϵ-close to having property Π. This contradicts the possibility that some edge lower-bound parameter is positive (i.e., $c > 0$), because it would have meant that (for some constant $c > 0$) the independent set is too far (i.e., c-far) from having property Π

Turning to Part (2), we first observe that if for some i the subgraph induced by V_i is a clique then (for every $\epsilon > 0$ and all sufficiently large N) the N-vertex clique graph is ϵ-close to having property Π. (The proof is similar to the main part of the above argument.) Now, consider a witness partition, denoted $(V_1', ..., V_k')$, of a graph G' satisfying Π that is ϵ-close to the N-vertex clique graph, and let j be such that $|V_j'| \geq N/k$. Then, the number of edges with both endpoints in V_j' must be at least $\frac{|V_j'|^2}{2} - \epsilon N^2 \geq \frac{(N/k)^2}{2} - \epsilon N^2 > 1$ (where the last inequality holds for $\epsilon < 1/2k^2$ and sufficiently large N). Using the hypothesis that no edge lower-bound parameter is positive, it follows that omitting edges from G' results in a graph that also has property Π. In particular, by possibly omitting a single edge residing in V_j', we can obtain a graph G'' satisfying Π so that $(V_1', ..., V_k')$ is also a witness partition of G'' and so that the subgraph of G'' induced by V_j' is neither a clique nor an independent set. Using Claim 5.2, Part (2) follows. ■

Corollary 5.4 *Let* Π *be a graph partition property that is testable by a canonical tester with one-sided error and query-complexity independent of* N, *and suppose that* Π *is not trivial. Then exactly one of the following two cases holds:*

1. *Every graph having property* Π *is* k*-colorable, and all edge lower-bound parameters in the specification of* Π *are zero. Furthermore, all upper-bounds referring to edges inside parts must be zero.*

2. *Every graph having property* Π *can be* k*-partitioned so that each part is a clique.*

Proof: Using Claim 5.2, the parts in a witness partition of any graph G having property Π must be either cliques or independent sets. Suppose first that for *some* G having property Π, some part of the witness partition of G is an independent set. Then, by Part 1 of Claim 5.3, all edge lower-bounds in the specification of Π are zero. Using Part 2 of Claim 5.3, in this case no part in the witness partition of *any* graph G' having property Π (regardless if $G' = G$ or not) can be a clique, and so (using Claim 5.2 again) all parts in the witness partition of every graph having property Π are independent sets (and so the graph is k-colorable). The only other case allowed for (non-trivial) Π is the one described in Item 2 of the corollary. The main part of the corollary follows.

To finish the proof we show that in the first case (i.e., Π implying k-colorability) all the upper-bounds on the number of edges inside parts must be zero. Suppose on the contrary that the i^{th} upper-bound referring to edges inside parts equals $c > 0$, and consider a witness partition $(V_1, ..., V_k)$ of a sufficiently large graph $G = ([N], E)$ having property Π. Specifically, we need $\frac{(k+1)^2}{2} \leq c \cdot N^2$ and $|V_i| \geq k+1$, where the latter can be obtained by amplifying the graph and the witness partition (as in Eq. (10)). Indeed, each of the V_j's is an independent set, but we can easily modify G to a graph G' that satisfies Π and yet contains a $(k+1)$-clique (in contradiction to the hypothesis that all graphs satisfying Π are k-colorable). This is done by putting a $(k+1)$-clique inside V_i, which does not violate the edge density upper-bound of the i^{th} part (and thus guarantees that the modified graph satisfies Π). The furthermore-part of the corollary follows. ■

Below, we consider the two cases of Corollary 5.4. We refer to the second case of Corollary 5.4 (i.e., a graph is k-partitioned so that each part is a clique) as to a graph is *covered by k cliques*.

5.3.1 Graph partition properties that imply k-colorability

We call an lower-bound (resp., upper-bound) parameter trivial if it equals 0 (resp., 1). That is, a trivial bound parameter is satisfied by any k-partition of any graph.

The notion of a relaxation. We say that property Π' is a relaxation of Π if every graph satisfying property Π also satisfies property Π'. For $\epsilon > 0$, we say that property Π' is an ϵ-relaxation of Π if Π' is a relaxation of Π and every sufficiently large graph satisfying property Π' is ϵ-close to satisfying Π. We say that Π' is a 0-relaxation of Π if, for every $\epsilon > 0$, property Π' is an ϵ-relaxation of Π.

The notion of 0-relaxation is related to the notion of indistinguishability defined by Alon *et. al.* [2]. Note that if Π' is a 0-relaxation of Π then every tester for Π' is almost a tester for Π in the following sense: For every value of $\epsilon > 0$ the Π'-tester may behave improperly with respect to Π on finitely many graphs. Thus, as far as property testing is concerned, we may consider 0-relaxations of a property instead the property itself.

We conjecture that, within the framework of graph partition problems, all but finitely many graphs that satisfy a 0-relaxations of a property also satisfy property itself. This conjecture was already proven for the special case of trivial properties (see Lemma 5.1),[9] and is established next for a more general special case that suffices for our needs.

Lemma 5.5 *Let* Π *and* Π' *be* graph partition properties. *Suppose that all edge lower-bound parameters in the specification of both* Π *and* Π' *are zero, and that each edge upper-bound parameter in the specification of* Π' *is either zero or one. Further suppose that* Π' *is a 0-relaxation of* Π. *Then, for every sufficiently large graph, the graph has property* Π *if and only if it has property* Π'.

The proof can be found in Appendix D.

[9]The set of all graph (i.e., the property satisfied by all graphs) is a 0-relaxation of any trivial graph property. Lemma 5.1 can be restated as saying that, for any trivial graph partition property Π and all sufficiently large N, any N-vertex graph (i.e., that is a graph and thus satisfies the 0-relaxation) satisfies property Π.

Towards a characterization. The main step towards characterizing graph partition properties that imply k-colorability is the following characterization of their 0-relaxations.

Lemma 5.6 *Let Π be as in Corollary 5.4. Suppose that every graph having property Π is k-colorable. Then, there exists a graph partition property Π'' that is a 0-relaxation of Π so that the only non-trivial bounds in the specification of Π'' are upper-bounds that equal zero.*[10] *Furthermore, these zero upper-bounds must include the upper-bounds referring to edges inside each part.*

Proof: As a first step, consider a specification of a property Π' derived from the specification of Π as follows: All edge upper-bounds that equal zero in the specification of Π are set to zero also in the specification of Π', and all other edge bounds in Π' are trivial. (Recall that, by Part 1 of Corollary 5.4, all edge lower-bounds in Π are trivial, and so this holds also for Π'.) The vertex bounds of Π are maintained in Π'. (We will deal with them at the second stage.)

Recall that in Π all the upper-bounds referring to edges inside parts must be zero. Thus, all edge bounds of property Π' are as required. Clearly, Π' is a relaxation of Π and so to establish that Π' is a 0-relaxation of Π we need to show that, for every $\epsilon > 0$, every sufficiently large graph having property Π' is ϵ-close to have property Π.

Fixing any $\epsilon > 0$, we consider a sufficiently large N so that the vertex-sample chosen by the tester on distance parameter ϵ is smaller than the numbers implied by all positive non-trivial edge upper-bounds of Π. That is, if $c > 0$ be the smallest positive edge upper-bound of Π, then we set $N > t(\epsilon)/\sqrt{c}$, where $t(\epsilon)$ is the size of the sample chosen by the tester.

Let $G' = ([N], E')$ be an arbitrary N-vertex graph satisfying Π'. Consider a vertex sample, denoted S, taken by the tester (for Π) on distance parameter ϵ and access to the oracle G'. We first show that the (small) subgraph of G' induced by S can be embedded in a graph G that satisfies Π, where G is derived from G' by omitting almost all edges. Specifically, we consider the graph $G = ([N], E)$ obtained by letting $E = \{(u, v) \in E' \cap (S \times S)\}$. The only bounds of Π that can be violated by a graph having property Π' are positive (non-trivial) edge upper-bounds, because all other bounds of Π equals those of Π'. But these bounds cannot be violated by G, because G has very few edges (i.e., G has less than $|S|^2$ edges, and N was chosen so that $|S|^2 < c \cdot N^2$). It follows that any induced subgraph that can be seen by the test (for Π) when given access to the oracle G', is also seen by the test with positive probability when given access to some oracle representing a graph that has property Π. Using the one-sided error feature of the test (for Π), it follows that the test accepts G' with probability 1, and hence G' must be ϵ-close to having property Π. This concludes the proof that Π' is a 0-relaxation of Π.

We now turn to the next step: Starting from Π', we obtain Π'' by possibly modifying the vertex bounds, and leaving all edge bounds intact. Specifically, we set all non-zero vertex-bounds to be trivial (i.e., 0 for lower-bounds and 1 for upper-bounds), and maintain zero vertex upper-bounds and lower-bounds (in case they are present in Π and Π').[11] Using an argument as in the first stage, it follows that Π'' is a 0-relaxation of Π'. Specifically, Π'' is a relaxation of Π', and for every $\epsilon > 0$ and sufficiently large graph satisfying Π'', it is the case that the very same graph is ϵ-close to Π'. (Intuitively, looking at the witness partition w.r.t Π'', observe that the only bounds of Π' that can be violated by that partition are non-zero (vertex) bounds of Π'; but this cannot be detected with one-sided error from an $o(N)$-size vertex-sample.) The lemma follows. ■

---

[10] In case all vertex upper-bounds in Π are positive (see Footnote 11), the only non-trivial bounds in the specification of Π'' are edge upper-bounds (which equal zero).

[11] Recall that zero lower-bounds are trivial, whereas zero upper-bounds on vertex-density are non-trivial but quite idiotic (because they merely mean that we specify a k'-partition problem, for some $k' < k$, rather than a k-partition problem).

5.3.2 Graph partition properties that imply a cover by k cliques

Lemma 5.7 *Let Π be as in Corollary 5.4. Suppose that every graph having property Π can be k-partitioned so that each part is a clique. Then, for sufficiently large N, an N-vertex graph has property Π if and only if it is an N-vertex clique.*

(Recall that we consider only properties that are satisfied by some graphs.)

Proof: Let $G = ([N], E)$ be an arbitrary graph having property Π. As shown in the proof of Claim 5.2 (see also below), the graph G can be assumed to be large enough so that some part in its witness partition is larger than the vertex-sample taken by the tester (on distance parameter ϵ). Let us denote the size of that sample by $t = t(\epsilon)$. Since the tester has one-sided error (and G contains a t-vertex clique), the tester must accept when the subgraph induced by the vertex-sample is a t-vertex clique. It follows that the N-vertex clique is ϵ-close to Π. Below we shall show that no other N-vertex graph can be accepted by the tester. One consequence of this is that the N-vertex clique must have property Π (because, otherwise no N-vertex graph has property Π).

Suppose, towards the contradiction, *that $G = ([N], E)$ has property Π but is not the N-vertex clique.* We consider an amplified version of G, denoted $G' = ([t] \times [N], E')$, where E' is as in Eq. (10). Then, on one hand G' has property Π (with witness partition induced by that of G).[12] On the other hand, G' contains as an induced subgraph a $2t$-vertex graph consisting of a pair of t-cliques (corresponding to any missing edge in G). However, for every $i = 1, ..., t-1$, with positive probability the tester given oracle access to G' sees an induced subgraph consisting of two cliques, one of size i and the other of size $t - i$ (and no additional edges). Since the tester has one-sided error (and G' has property Π), the tester must accept in each of these cases. It follows that when given oracle access to any N-vertex graph consisting of two cliques (and no additional edges), the tester will always accept. Thus, subject to the contradiction hypothesis,[13] we have:

Claim 1: For all sufficiently large N and every $1 < M < N$, every graph that consists of an M-vertex clique and an $(N - M)$-vertex clique (and no additional edges) is ϵ-close to having property Π.

Next, we consider the edge lower-bounds in the specification of Π; that is, let $\mathtt{l}_{i,i}$ (resp., $\mathtt{l}_{i,j}$) denote the lower-bound referring to edge density within the i^{th} part (resp., between the i^{th} and the j^{th} parts). Specifically, these lower-bounds require that the number of edges within the i^{th} part is at least $\mathtt{l}_{i,i} \cdot N^2$ (resp., between the i^{th} and the j^{th} parts is at least $\mathtt{l}_{i,j} \cdot N^2$).

Motivation: For simplicity we consider the case $k = 2$ (observing that the lemma is trivial in case $k = 1$). Furthermore, for simplicity, we first assume that $\epsilon = 0$. Using Claim 1, it follows that the graph consisting of two $(N/2)$-vertex cliques (and no additional edges) has property Π. The only witness partition possible for this graph is the one in which each clique is in a different part, and thus each part is allowed to have at least one half of the number of vertices. Furthermore, both $\mathtt{l}_{1,1} \leq 1/8$ and $\mathtt{l}_{2,2} \leq 1/8$ (since each part contains a $(N/2)$-vertex clique, and so has only $\frac{(N/2)^2}{2} = \frac{N^2}{8}$ edges). Next, we consider the graph consisting of one $(N/3)$-vertex clique and one $(2N/3)$-vertex clique, which (by Claim 1) also has property Π. Again, the witness partition of this graph has each clique is in a different part, and it follows that either $\mathtt{l}_{1,1} \leq 1/18$ or $\mathtt{l}_{2,2} \leq 1/18$ (since the part containing the $(N/3)$-vertex clique has only $\frac{(N/3)^2}{2} = \frac{N^2}{18}$ edges). Suppose, without loss of generality, that $\mathtt{l}_{2,2} \leq 1/18$. But now it follows that also the graph consisting of of one $(N/2)$-vertex clique, one $(N/3)$-vertex clique, and $N/6$ isolated vertices satisfies Π (e.g., consider

[12] Observe that the parts of the induced partition are cliques, and that the fraction of edges between the parts is exactly as in the witness partition of G.

[13] That is, assuming that $G = ([N], E)$ has property Π but is not the N-vertex clique.

the witness partition in which the $(N/2)$-vertex clique is in one side and the rest of the graph is in the other). This contradicts the lemma's hypothesis by which every graph having property Π can be covered by k cliques, and so cannot have a large independent set. Thus, the contradiction hypothesis (by which there exists a graph $G = ([N], E)$ that has property Π but is not the N-vertex clique) must be false. The analysis is easily extended to small $\epsilon > 0$, but extending it to arbitrary $k > 2$ is more involved.

The heart of the actual analysis (for arbitrary $k \geq 2$ and $\epsilon > 0$) is stated and proven next:

Claim 2: Subject to the contradiction hypothesis (see Footnote 13), there must be a set $C \subset [k]$ such that

$$\sum_{i \leq j \in C} \mathtt{1}_{i,j} \leq 3k^2\epsilon \tag{11}$$

Furthermore, for all sufficiently large N, there exists an N-vertex graph H' satisfying Π and a witness partition $(V_1', ..., V_k')$ of H' such that

$$\sum_{i \in C} |V_i'| \geq k\sqrt{\epsilon}N \tag{12}$$

Proof: For N as in Claim 1, consider an N-vertex graph H that consists of a pair of cliques (and no additional edges), where the smaller clique is of size $2k\sqrt{\epsilon} \cdot N$. Since (By Claim 1) H is ϵ-close to having property Π, we may consider a witness partition $(V_1', ..., V_k')$ of a graph H' that satisfies Π and is ϵ-close to H. Since each V_i' is a clique in H', the subgraph of H induced by V_i' misses at most ϵN^2 (because H is ϵ-close to H'). Thus, each V_i' is "dominated by one of the cliques of H" in the sense that either it contains at most $\sqrt{\epsilon}N$ vertices of the small clique (of H) or it contains at most $\sqrt{\epsilon}N$ vertices of the large clique (since otherwise the subgraph of H induced by V_i' misses more than $(\sqrt{\epsilon}N)^2$ edges). It follows that there exist $C \subset [t]$ such that the parts with index in C contain at most $k\sqrt{\epsilon}N$ vertices of the large clique and all but at most $k\sqrt{\epsilon}N$ vertices of the small clique (e.g., C contain the indices of all parts that each have at most $\sqrt{\epsilon}N$ vertices of the large clique, and so $\overline{C} \stackrel{\rm def}{=} [t] \setminus C$ (which contains only parts with more than $\sqrt{\epsilon}N$ vertices of the large clique) contains parts that each have at most $\sqrt{\epsilon}N$ vertices of the small clique). In particular, the number of vertices residing in parts with an index in C is at least $2k\sqrt{\epsilon} \cdot N - k\sqrt{\epsilon} \cdot N = k\sqrt{\epsilon} \cdot N$, and Eq. (12) follows.

Turning to Eq. (11), we observe that the number of edges (in H) having both endpoints residing in parts having an index in C is at most $\frac{1}{2} \cdot ((2k\sqrt{\epsilon} \cdot N)^2 + (k\sqrt{\epsilon} \cdot N)^2) = \frac{5k^2\epsilon}{2} \cdot N^2$. It follows that the corresponding number in H' is at most $\frac{5k^2\epsilon}{2} \cdot N^2 + \epsilon N^2 < 3k^2\epsilon N^2$, where $\frac{k^2}{2} > 1$ is due to $k \geq 2$. Since H' satisfies Π (and in particular its edge lower-bounds), Eq. (11) follows. $\Box$

We stress that the above two claims holds for any value of $\epsilon > 0$, subject to the contradiction hypothesis (see Footnote 13). Thus, for every sequence $\epsilon_1, \epsilon_2, ..., \epsilon_{2^k+1}$ of positive numbers, there exists a set $C \subset [k]$ such that for some $p < q$ and all sufficiently large N, there exist an N-vertex graph H' satisfying Π and a witness partition $(V_1', ..., V_k')$ such that

$$\sum_{i \leq j \in C} \mathtt{1}_{i,j} \leq 3k^2\epsilon_q \tag{13}$$

$$\sum_{i \in C} |V_i'| \geq k\sqrt{\epsilon_p}N \tag{14}$$

It follows that there exists $i \in C$ such that $|V_i'| \geq \sqrt{\epsilon_p} N$ and $\mathtt{l}_{i,i} \leq 3k^2\epsilon_q$. Selecting the sequence of ϵ_j's so that $\epsilon_j = (7k^2)^{-j}$, we have $\epsilon_q \leq \epsilon_{p+1} = \epsilon_p/(7k^2)$, and so

$$\begin{aligned} & \frac{|V_i'|^2}{2} \geq \frac{(\sqrt{\epsilon_p}N)^2}{2} = \frac{\epsilon_p N^2}{2} \geq \frac{7k^2\epsilon_q}{2} \cdot N^2 > 3k^2\epsilon_q N^2 + \frac{(k+1)^2}{2} \\ \Rightarrow \quad & \frac{|V_i'|^2}{2} > \mathtt{l}_{i,i} \cdot N^2 + \frac{(k+1)^2}{2} \end{aligned} \tag{15}$$

Eq. (15) allows us to modify H' so that the resulting graph also satisfies Π but has an independent set of size $k+1$, which contradicts the (contradiction) hypothesis that every graph having property Π can be k-partitioned into cliques. Specifically, recall that H' has property Π and that $(V_1', ..., V_k')$ is a witness partition. In particular, the subgraph induced by V_i' is a clique, but we can omit $\frac{(k+1)^2}{2}$ from it without violating any of the bounds of Π (i.e., the only relevant bound is the lower bound on the number of edges inside V_i', but Eq. (15) asserts that this bound will continue to hold even if we omit $\frac{(k+1)^2}{2}$ edges with both endpoints in V_i'). This allows to omit all edges among a set of $k+1$ vertices belonging to V_i', resulting in a graph H'' that still satisfies Π (and has an independent set of size $k+1$). Thus, the graph H'' violates the lemma's hypothesis (that graphs satisfying Π can be k-partitioned into cliques). It follows that the contradiction hypothesis (by which there exists a graph $G = ([N], E)$ that has property Π but is not the N-vertex clique) must be false. The lemma follows. ■

5.4 The characterization theorem and a corollary

Combining Corollary 5.4, and Lemmas 5.5–5.7, we obtain:

Theorem 5.8 (Theorem 3, restated): *Let Π be a graph partition property that is testable with one-sided error and query-complexity independent of N, and suppose that Π is not trivial. Then exactly one of the following two cases holds:*

1. *There exists a k-vertex graph H so that for all sufficiently large graphs G, the graph G satisfies Π if and only if its vertices can be k-partitioned such that there are no edges among vertices residing in the same part and so that there are edges between vertices of the i^{th} part and j^{th} part only if (i, j) is an edge of H.*
2. *For sufficiently large N, an N-vertex graph has property Π if and only if it is an N-vertex clique.*

Proof: By Theorem 4.5, we may assume that the tester is canonical, and apply Corollary 5.4. Assuming that Π is as in Case 1 of Corollary 5.4, we apply Lemma 5.6 and conclude that Π has a 0-relaxation Π' such that the only non-trivial bounds in the specification of Π' are upper-bounds that equal zero, and that all graphs satisfying Π' are k-colorable. Thus, we are allowed to apply Lemma 5.5 to this pair (Π, Π') and conclude that every sufficiently large graph having property Π is k-colorable, and (without loss of generality) the only non-trivial bounds in the specification of Π are upper-bounds that equal zero. Defining H so that $(i, j) \in [k] \times [k]$ is an edge if and only if the upper-bound referring to edges between the i^{th} and j^{th} part is trivial, we obtain the condition of Case 1 of the current theorem.

The only other possibility is that Π is as in Case 2 of Corollary 5.4. Applying Lemma 5.7, we obtain the condition of Case 2 of the current theorem. ■

The property checked by the canonical tester: We are now ready to address the question posed at Section 4.2.4. Using Theorem 5.8, we show that, in the context of graph partition problems, if a property is testable by a one-sided error tester of complexity that only depends on ϵ then it can be so tested by a canonical tester that accepts iff the induced subgraph has the very same property (rather than some other graph property).

Corollary 5.9 *Let* Π *be a graph partition property that is testable by a canonical tester with one-sided error and query-complexity independent of* N*. Then* Π *can be tested with one-sided error by checking whether or not a random* $\mathrm{poly}(1/\epsilon)$*-vertex induced subgraph has the property* Π.

Proof: The conclusion holds vacuously for trivial properties. Thus, using Theorem 5.8, we need to consider only two cases (corresponding to the two items). We observe that in the second case (i.e., the clique property) one may simply accept if and only if the subgraph induced by a random sample of $t = O(1/\epsilon)$ vertices is a clique. Clearly, if the N-vertex graph is a clique then so is any induced subgraph. On the other hand, if $G = ([N], E)$ is ϵ-far from being an N-vertex clique, then at least ϵN^2 edges must be missing from G. Considering some fixed matching of $t/2$ pairs of vertices in the sample, we conclude that the probability that the corresponding $t/2$ edges are present in G is at most $(1-\epsilon)^t < \frac{1}{3}$ (since the matched sample-pairs are independently distributed among all possible vertex-pairs).

We now turn to the more interesting case in which Π is as in Case 1 of Theorem 5.8. It follows that for any G having property Π, every subgraph of G also has property Π. More interestingly, following the technique of [7, Cor. 7.2], we show that if G is ϵ-far from having Π then a random $\mathrm{poly}(1/\epsilon)$-vertex induced subgraph is unlikely to have property Π'. Following is a brief sketch of the argument.

Our starting point is the general graph partition property test of [7, Sec. 9], which uses a sample of $t = \mathrm{poly}(1/\epsilon)$ vertices, and has two-sided error that can be bounded by $1/5$. Consider what happens when we first select a random $\mathrm{poly}(t)$-vertex induced subgraph, and then invoke the abovementioned tester on the induced subgraph. In this case, if the induced subgraph has property Π, then the tester must accept with probability at least $4/5$. On the other hand, in case the tester is invoked on G itself (which is ϵ-far from satisfying Π) the tester accepts with probability at most $1/5$. The punch-line is that the sample viewed by the tester is distributed almost identically in the two cases.[14] It follows that, with probability greater than $2/3$, the random induced subgraph does not have property Π. The corollary follows. ■

Acknowledgments

We are grateful to Yevgeniy Dodis for raising the question addressed in Section 3, to Michael Krivelevich for making a comment that initiated the work presented in Section 4, to Noga Alon for insisting that a better transformation of adaptive testers to non-adaptive ones should be possible in this context, and to Dana Ron for offering a simplification to the proof of Lemma 4.1. We thank Noga Alon, Michael Krivelevich, Dana Ron, and two members of the FOCS'01 PC for helpful comments.

[14]For sufficiently large t, the distribution obtained by first selecting a sample of size t^3 from a huge set, and then selecting a sample of size t in the first sample is very close to the distribution obtained by selecting the small sample directly from the huge space. (In fact, using additional ideas, one may use a first sample of size $O(t)$ rather than t^3; cf. [7, Cor. 7.2].)

References

[1] N. Alon. Private communication, 2001.

[2] N. Alon, E. Fischer, M. Krivelevich and M. Szegedy. Efficient Testing of Large Graphs. In *Proc. of the 40th FOCS*, pages 645–655, 1999.

[3] N. Alon, O. Goldreich, J. Hastad, and R. Peralta. Simple constructions of almost k-wise independent random variables. *Journal of Random structures and Algorithms*, Vol. 3 (3), pages 289–304, 1992.

[4] N. Alon, and M. Krivelevich. Testing k-Colorability. Preprint, 1999.

[5] Z. Bar-Yossef, R. Kumar and D. Sivakumar. Sampling Algorithms: Lower Bounds and Applications. In *Proc. of the 33rd STOC*, pages 266–275, 2001.

[6] O. Goldreich. Combinatorial Property Testing – A Survey. In *DIMACS Series in Disc. Math. and Theoretical Computer Science*, Vol. 43 (Randomization Methods in Algorithm Design), pages 45–59, 1998.

[7] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, pages 653–750, July 1998.

[8] J. Naor and M. Naor. Small-bias Probability Spaces: Efficient Constructions and Applications. *SIAM J. on Computing*, Vol 22, 1993, pages 838–856.

[9] D. Ron. Property Testing (A Tutorial). To appear in *Handbook on Randomization*. Available from `http://www.eng.tau.ac.il/~danar/papers.html`.

[10] R. Rubinfeld and M. Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.

Appendix A: The transformation of Alon *et. al.* [2]

The following transformation consists of two steps. First we transform any tester into one that operates in a non-adaptive manner (i.e., determines its queries independently of the answers it has obtained to previous queries). This step is totally generic and merely uses the fact that the oracle answers are binary. Next, we transform any non-adaptive tester into one that inspects all vertex-pairs in some random set of vertices. This step is less generic and uses the fact that we are dealing with testers of graph properties.

The entire transformation presented in this section has appeared before in the work of Alon *et. al.* [2], and is inferior to the transformation presented in Section 4.1. We include the current description both for sake of self-containment (and because the description in [2] is quite laconic) and as a warm-up towards the second transformation (presented in Section 4.1).

A.1 Moving to non-adaptive testers

A non-adaptive oracle machine is one that determines its queries based merely on its input and random-coins, independently of the answers to prior queries. A straightforward (and generic) way of eliminating adaptivity in (deterministic) oracle machines is to consider all possible queries that can be made as determined by all possible answers to prior queries. This extends easily to probabilistic oracle machines. Thus, for each fixed contents of the random-tape, the i^{th} adaptive query can be replaced by 2^{i-1} non-adaptive queries that correspond to the queries made per each possible outcome of the previous $i-1$ queries. We stress that the non-adaptive machine makes these 2^{i-1} queries, but its next move depends only on the single answer given to the relevant query (being the actual i^{th} query that the adaptive machine would have asked). Thus, we obtain:

Claim A.1 *Let Π be any graph property. If there exists a tester with query complexity $q(N,\epsilon)$ for Π, then there exists a non-adaptive tester with query complexity $\sum_{i=1}^{q(N,\epsilon)} 2^{i-1} < 2^{q(N,\epsilon)}$ for Π.*

A.2 Moving to testers that inspect a random induced subgraph

The next transformation is far less generic than the previous one. It capitalized on the fact that, by definition, graph properties are preserved under isomorphism (i.e., renaming of vertex names). Thus, when testing graph properties, the names of vertices are not important, and so we may assume (w.l.o.g) that the tester selects random vertices.

Claim A.2 *Let Π be any graph property. If there exists a non-adaptive tester with query complexity $q(N,\epsilon)$ for Π, then there exists such a tester that selects a random subset of $2q(N,\epsilon)$ vertices, denoted R, makes the queries $\{(u,v) : u,v \in R\}$, and decides based on the oracle answers (and its internal coin tosses). Thus, the query complexity of the new tester is less than $2q(N,\epsilon)^2$.*

Proof: Let T be a non-adaptive tester for Π, and suppose that T has query complexity $q(N,\epsilon)$. Then, depending on its internal coin tosses r, the tester makes queries $(u_r^1, v_r^1), ..., (u_r^t, v_r^t)$, where $t \le q \stackrel{\text{def}}{=} q(N,\epsilon)$. Let $V_r = \{u_r^i, v_r^i : i \in [t]\}$ be the set of vertices appearing in queries of T on coins r. Augment V_r by some fixed vertices (e.g., out of the set $[2q]$) so that the resulting set V_r has cardinality $2q$. Then, we can transform T into a related test T' that, on internal coin tosses r, makes the queries $\{(u,v) : u,v \in V_r\}$, and ignores the answers obtained for pairs not queried by T. Clearly, T' is also a tester for Π. Consider now an oracle machine, denoted T'', that behaves as follows.

1. First, T'' uniformly selects a permutation π over the vertex set (i.e., $[N]$).
2. Next, T'' invokes T' and emulates the oracle of T' by queries to its own oracle so that when T' makes the query (u, v), machine T'' makes the query $(\pi(u), \pi(v))$ and answers T' accordingly.

We first observe that T'' makes queries to all pairs in a random set of $2q$ vertices. This follows because, for every fixed (T'-internal coin sequence) r and random permutation π, the set $\{\pi(v) : v \in V_r\}$ is a random set of $2q$ vertices. Next we show that T'' maintains the testing features of T' (and thus of T):

- Let $G = ([N], E)$ be a graph having property Π. Then, for any permutation π, it is the case that the graph $\pi(G) \stackrel{\text{def}}{=} ([N], \{(\pi(u), \pi(v)) : (u, v) \in E)$ has property Π. Thus, T' must accept the graph $\pi(G)$ with probability at least $2/3$. This means that conditioned on π being chosen in Step 1 (of T''), algorithm of T'' accepts the graph G with probability at least $2/3$ (because, in this case, T'' just emulates for T' an oracle access to the graph $\pi(G)$). Since this holds for every π, it follows that T'' accepts G with probability at least $2/3$.
- Suppose that a graph $G = ([N], E)$ is ϵ-far from having property Π. Then, for any permutation π, the graph $\pi(G)$ is ϵ-far from having property Π. (See Claim 2.1.) The rest of the argument follows analogously to the above (where here we refer to an upper bound on the accepting probability). It follows that T'' accepts G with probability at most $1/3$.

The claim follows. ■

Appendix B: The framework of Graph Partition Problems

A graph partition property is defined by a sequence of pairs of non-negative numbers. For some integer k, we have $k + k + \binom{k}{2}$ pairs providing upper and lower bounds on the fraction of vertices in each part of the k-partition as well as on the fraction of edges within parts and between parts. Specifically, consider the sequence of pairs

$$(\mathtt{l}_1, \mathtt{u}_1), ..., (\mathtt{l}_k, \mathtt{u}_k), (\mathtt{l}_{1,1}, \mathtt{u}_{1,1}), ..., (\mathtt{l}_{k,k}, \mathtt{u}_{k,k}), ((\mathtt{l}_{i,j}, \mathtt{u}_{i,j}))_{1 \le i < j \le k} \tag{16}$$

This sequence corresponds to a graph property that is satisfied by all graphs $G = ([N], E)$ having a k-partition $(V_1, ..., V_k)$ such that the following two conditions hold:

$$\begin{aligned} \mathtt{l}_i \cdot N \;\le\; |V_i| \;\le\; \mathtt{u}_i \cdot N \qquad & \forall i \\ \mathtt{l}_{i,j} \cdot N^2 \;\le\; |E \cap (V_i \times V_j)| \;\le\; \mathtt{u}_{i,j} \cdot N^2 \qquad & \forall i \le j \end{aligned}$$

That is, $\mathtt{l}_i$ (resp., $\mathtt{u}_i$) is a lower bound (resp., upper bound) on the fraction of vertices in the i^{th} part, $\mathtt{l}_{i,i}$ (resp., $\mathtt{u}_{i,i}$) is a lower bound (resp., upper bound) on the fraction of edges having both endpoints in the i^{th} part, and $\mathtt{l}_{i,j}$ (resp., $\mathtt{u}_{i,j}$) is a lower bound (resp., upper bound) on the fraction of edges crossing between the i^{th} part and the j^{th} part, for $i < j$.

Certainly, some sequences of parameters give rise to graph partition properties that are not satisfied by any graph. We discard these cases from our discussion (calling them *non-admissible*; see [7, Def. 9.3.1]). In particular, we will consider only sequences as in Eq. (16) satisfying $0 \le \mathtt{l}_i \le \mathtt{u}_i \le 1$ and $0 \le \mathtt{l}_{i,j} \le \mathtt{u}_{i,j} \le 1$, for all i, j.

A technicality: integrality problems. Following [7, Rem. 9.1], we avoid integrality problems by allowing upto $k-1$ vertices to be split between the k parts of the partition (and count these fractional vertices and edges in the natural way). Had we not followed this convention, the set of N-vertex graphs satisfying a graph partition property could be empty for some values of N and non-empty for others.

A technicality: counting edges and self-loops. Note that the edge bounds impose bounds in terms of multiples of N^2 (rather than of $\binom{N}{2}$, which may be more natural). This convention is adopted for greater expressibility. For example, using this convention, a full bipartite graph with $N/2$ vertices on each side has edge density $\frac{(N/2)^2}{N^2} = \frac{1}{4}$, which is independent of N. (In contrast, if we consider multiples of $\binom{N}{2}$, then such a graph will have edge density $\frac{(N/2)^2}{\binom{N}{2}}$, which is not independent of N, and consequently the the corresponding condition could not have been expressed as a graph partition problem.) For similar reasons, we allow self-loops and count them as half edges.[15] This way, a ρN-vertex clique in an N-vertex graph has edge density $\frac{\binom{\rho N}{2}+\frac{\rho N}{2}}{N^2} = \frac{\rho^2}{2}$, which is independent of N. Consequently, the property of having such a clique can be expressed as a graph partition problem.

Indeed, using multiples of $\frac{N^2}{2}$ rather than N^2 would be more natural, but both more cumbersome and less in agreement with the presentation in [7, Sec. 9]. Clearly, the last choice is immaterial.

Appendix C: Proof of Lemma 5.1

Recall that by the lemma's hypothesis Π is a trivial graph partition property. Our aim is to show that, for all sufficiently large N, every N-vertex graph has property Π. This follows by combining the following three claims.

Claim C.1 *The specification of* Π *does not contain any positive lower-bound regarding edges.*

Proof: Suppose on the contrary that Π contains a positive lower-bound regarding edges. Then, for some $c > 0$, each N-vertex graph satisfying Π must have at least $c \cdot N^2$ edges, which contradicts the hypothesis that Π is trivial (because, in this case, it cannot hold that for every $\epsilon > 0$ and all sufficiently large N, the N-vertex independent set is ϵ-close to having property Π). ■

Claim C.2 *For all sufficiently large N, the N-vertex clique has property* Π.

The proof of Claim C.2 is postponed to the end of this section.

Claim C.3 *Let Ξ be a graph partition property with a specification that does not contain any positive lower-bound regarding edges. If the N-vertex clique has property Ξ, then all N-vertex graphs have property Ξ.*

[15] Unfortunately, the text of [7, Sec. 9] is unclear regarding this aspect, which is essential for the claim that the ρ-clique problem can be expressed as a graph partition problem. Furthermore, for simplicity, in [7, Sec. 5] self-loops are disallowed. We stress that all the results of [7] are preserved if one allows self-loops (which can be ignored by all algorithms). Finally, we note that counting self-loops as half edges is consistent with [7], where each (non-self-loop) edge is counted twice, in correspondence to its two occurrences in the adjacency matrix of the graph. (Using this correspondence justifies counting self-loops once, which is half the count relative to edges that are not self-loops.) In this paper we chose to get rid of the annoying convention of counting each (non-self-loop) edge twice, and the result is the annoying convention by which a self-loop is counted half a time.

Proof: Consider a witness partition of the N-vertex clique, denoted C_N. Then this partition is also a witness partition of any N-vertex subgraph of C_N, because all edge lower-bounds are non-positive (i.e., zero). The claim follows. ■

Proof of Claim C.2: We consider the sequence of bounds in the specification of property Π, and refer to the notation in Eq. (16). For these bounds (i.e., $\mathtt{l}_i, \mathtt{u}_i, \mathtt{l}_{i,j}, \mathtt{u}_{i,j}$'s), consider the following system of equations in variables $x_1, ..., x_k$:

$$\sum_{i=1}^{k} x_i = 1 \tag{17}$$

$$\mathtt{l}_i \leq x_i \leq \mathtt{u}_i \qquad \forall i \tag{18}$$

$$\frac{1}{2} \cdot x_i^2 \leq \mathtt{u}_{i,i} \qquad \forall i \tag{19}$$

$$x_i \cdot x_j \leq \mathtt{u}_{i,j} \qquad \forall i < j \tag{20}$$

We first claim that the above system has a solution. Otherwise, there exists a constant $\epsilon > 0$ so that any solution satisfying Eqs. (17)&(18) violates one of the other equations by at least ϵ (i.e., either $\frac{1}{2} \cdot x_i^2 \geq \mathtt{u}_{i,i} + \epsilon$ or $x_i \cdot x_j \geq \mathtt{u}_{i,j} + \epsilon$ for some i, j). We will show that this contradicts the hypothesis that, for sufficiently large N, the N-vertex clique is $(\epsilon/2)$-close to having property Π: Let $(V_1, ..., V_n)$ be a witness partition of a graph G having property Π and being $(\epsilon/2)$-close to the N-vertex clique, and set $x_i = |V_i|/N$ for $i = 1, ..., k$. Then this setting satisfies Eqs. (17)&(18). Also, for every i, we must have $\frac{|V_i|^2}{2} - \frac{\epsilon}{2} \cdot N^2 \leq \mathtt{u}_{i,i} \cdot N^2$ (because, on one hand, the witness partition of G respects all bounds of Π, and on the other hand G may miss at most $\frac{\epsilon}{2} \cdot N^2$ edges (because it is $(\epsilon/2)$-close to the N-vertex clique)). Thus, we have $\frac{|V_i|^2}{2} < \epsilon \cdot N^2 + \mathtt{u}_{i,i} \cdot N^2$, and $\frac{1}{2} \cdot x_i^2 < \mathtt{u}_{i,i} + \epsilon$ follows. Similarly, $|V_i| \cdot |V_j| - \frac{\epsilon}{2} \cdot N^2 \leq \mathtt{u}_{i,j} \cdot N^2$ and $x_i \cdot x_j < \mathtt{u}_{i,j} + \epsilon$ follows for every $i < j$. This contradicts the above hypothesis that any solution satisfying Eqs. (17)&(18) violates one of the other equations by at least ϵ, and we conclude that the system of Eqs. (17)–(20) has a solution.

Let $(x_1, ..., x_k)$ be a solution to Eqs. (17)–(20). Then avoiding integrality problems (see [7, Rem. 9.1]), we consider for each N a partition $(V_1, ..., V_k)$ of the N-vertex clique so that $|V_i| = x_i N$. Clearly, this partition satisfies all vertex bounds (because these correspond to Eq. (18)). By Claim C.1, the only remaining non-trivial bounds are the edge upper-bounds (because all $\mathtt{l}_{i,j}$'s are zero). But these are shown to be satisfied as follows: For every i, the number of edges with both endpoints in V_i is $\frac{|V_i|^2}{2}$, and we have $\frac{|V_i|^2}{2} = \frac{x_i^2}{2} \cdot N^2 \leq \mathtt{u}_{i,i} \cdot N^2$, where the last inequality is due to Eq. (19). Similarly, for every $i < j$, the number of edges crossing between V_i and V_j is $|V_i| \cdot |V_j|$, and we have $|V_i| \cdot |V_j| = x_i x_j \cdot N^2 \leq \mathtt{u}_{i,i} \cdot N^2$, where the inequality is due to Eq. (20). The claim follows. ■

Appendix D: Proof of Lemma 5.5

Recall the hypotheses of the lemma:

1. Π is a graph partition property such that all edge lower-bound parameters in the specification of Π are zero.

2. Π' is a graph partition property such that all edge lower-bound parameters in the specification of Π' are zero, and each edge upper-bound parameter in the specification of Π' is either zero or one.

3. Π' is a 0-relaxation of Π.

Our aim is to show that, for every sufficiently large graph, the graph has property Π if and only if it has property Π'. (In fact, we only need to show the "if"-direction.) Our proof generalizes the proof of Lemma 5.1 (given in Appendix C). In particular, we do not need to establish an analogue of Claim C.1, because this is already guaranteed in our first hypothesis. The role of the N-vertex clique will be played by each member of a family of certain extremal (dense) graphs (and the type of the upper-bounds in Π' seem important to allow us to focus on this family (and thus perform this extension)). We thus prove analogies of the two other claims of Appendix C: Analogously to Claim C.2, we first show that each member of the extremal family that has property Π' also has property Π. Then, analogously to Claim C.3, we show that this extends to each subgraph of the extremal graphs, and that the latter subgraphs are all the graphs that may have property Π'. The lemma will follows.

Pivotal to the above plan, is the definition of *extremal graphs for* Π'. Since the edge lower-bounds of Π' are trivial and each edge upper-bound in Π' is either zero or one, the extremal graphs are determined by the density of the vertex sets allowed by the vertex bounds (and the type of the edge upper-bounds). That is, using notations as in Eq. (16), let $\mathtt{l}'_i, \mathtt{u}'_i, \mathtt{l}'_{i,j}, \mathtt{u}'_{i,j}$ be the bounds in the specification of Π'. (Recall that all $\mathtt{l}'_{i,j}$ equal zero and each $\mathtt{u}'_{i,j}$ equals either zero or one.) Then a sequence of vertex-set densities, denoted $\overline{\rho} = (\rho_1, ..., \rho_k)$, is permitted by Π' if $\sum_{i=1}^{k} \rho_i = 1$ and $\mathtt{l}'_i \leq \rho_i \leq \mathtt{u}'_i$ for all $i = 1, ..., k$. For each such permitted sequence $\overline{\rho}$ and every N, we consider the extremal graph $G^{(N,\overline{\rho})} = ([N], E^{(N,\overline{\rho})})$ defined by

$$V_i^{(N,\overline{\rho})} \stackrel{\rm def}{=} \left\{ v \in [N] : \sum_{j=1}^{i-1} \rho_j N < v \leq \sum_{j=1}^{i} \rho_j N \right\} \tag{21}$$

$$F^{(\overline{\rho})} \stackrel{\rm def}{=} \{(i,j) : 1 \leq i \leq j \leq k \ \wedge \ \mathtt{u}'_{i,j} = 1\} \tag{22}$$

$$E^{(N,\overline{\rho})} \stackrel{\rm def}{=} \bigcup_{(i,j) \in F^{(\overline{\rho})}} \left(V_i^{(N,\overline{\rho})} \times V_j^{(N,\overline{\rho})} \right) \tag{23}$$

That is, Eq. (21) specifies a (canonical) k-partition that satisfies the vertex bounds of Π' (i.e., is according to the permitted sequence $\overline{\rho}$), Eq. (22) indicates the part-pairs among which edges are allowed, and Eq. (23) mandates all possible edges among each allowed pair of parts. Since all edge lower-bounds in Π' are zero (and thus trivial), they are satisfied by the above k-partition of $G^{(N,\overline{\rho})}$. Since each edge upper-bound in Π' is either zero or one, it is also satisfied that partition (because there are edges between the $i^{\rm th}$ and $j^{\rm th}$ part iff $\mathtt{u}'_{i,j} = 1$). Thus, $G^{(N,\overline{\rho})}$ satisfies Π', and $(V_1^{(N,\overline{\rho})}, ..., V_k^{(N,\overline{\rho})})$ is a witness partition. (Note that we do not rule out the possibility that $F^{(\overline{\rho})}$ contains pairs of the form (i,i), although the text and notation may suggest otherwise. We comment that in our application of Lemma 5.5, $\mathtt{u}'_{i,i} = 0$ and so $(i,i) \notin F^{(\overline{\rho})}$ for all i.)

Recall that each extremal graph $G^{(N,\overline{\rho})}$ satisfies Π' and that $(V_1^{(N,\overline{\rho})}, ..., V_k^{(N,\overline{\rho})})$ is a witness partition. Next, in Claim D.1, we show that such extremal graph also satisfies Π. Later (see Claim D.2) we show that each N-vertex graph satisfying Π' is a subgraph of an N-vertex (graph that is isomorphic to an) extremal graph, and that this subgraph also satisfies Π. The latter claim (i.e., Claim D.2) is much easier.

Claim D.1 *Every extremal graph satisfies property* Π.

Proof: Let $G^{(N,\overline{\rho})}$ be an extremal graph (for Π'), where $\overline{\rho} = (\rho_1, ..., \rho_k)$. We consider the sequence of bounds in the specification of property Π, and denote these bounds by $\mathtt{l}_i, \mathtt{u}_i, \mathtt{l}_{i,j}, \mathtt{u}_{i,j}$'s. Recall that by our first hypothesis, all $\mathtt{l}_{i,j}$'s equal zero. (Typically, the $\mathtt{l}_i, \mathtt{u}_i, \mathtt{l}_{i,j}, \mathtt{u}_{i,j}$'s are (possibly) more stringent than the corresponding $\mathtt{l}'_i, \mathtt{u}'_i, \mathtt{l}'_{i,j}, \mathtt{u}'_{i,j}$'s. We stress that the following proof does not refer at all to the bounds $\mathtt{l}'_i, \mathtt{u}'_i, \mathtt{l}'_{i,j}, \mathtt{u}'_{i,j}$.)

Consider the following system of equations in the variables $x_{1,1}, ..., x_{k,k}$:

$$\sum_{j=1}^{k} x_{i,j} \;=\; \rho_i \qquad \forall i \tag{24}$$

$$\mathtt{l}_j \;\leq\; \sum_{i=1}^{k} x_{i,j} \;\leq\; \mathtt{u}_j \qquad \forall j \tag{25}$$

$$\frac{1}{2} \cdot \sum_{(i,i)\in F^{(\overline{\rho})}} x_{i,j}^2 + \sum_{(i,i')\in F^{(\overline{\rho})}\setminus\{(i,i):i\in[k]\}} x_{i,j}x_{i',j} \;\leq\; \mathtt{u}_{j,j} \qquad \forall j \tag{26}$$

$$\sum_{(i,i')\in F^{(\overline{\rho})}} x_{i,j}x_{i',j'} \;\leq\; \mathtt{u}_{j,j'} \qquad \forall j < j' \tag{27}$$

We first claim that the above system has a solution. Otherwise, there exists a constant $\epsilon > 0$ so that any solution satisfying Eqs. (24)&(25) violates one of the other equations by at least ϵ. We will show that this contradicts the hypothesis that, for sufficiently large N, the graph $G^{(N,\overline{\rho})}$ (which satisfies property Π') is $(\epsilon/2)$-close to having property Π: Let $(V_1, ..., V_n)$ be a witness partition of a graph G having property Π and being $(\epsilon/2)$-close to the graph $G^{(N,\overline{\rho})}$, and let $V_{i,j} = V_i^{(N,\overline{\rho})} \cap V_j$ for $i, j \in [k]$. Now, set $x_{i,j} = |V_{i,j}|/N$ for $i, j \in [k]$. This setting satisfies Eq. (24), because $\cup_j (V_i^{(N,\overline{\rho})} \cap V_j) = V_i^{(N,\overline{\rho})}$ and $|V_i^{(N,\overline{\rho})}| = \rho_i N$. It also satisfies Eq. (25), because $\cup_i (V_i^{(N,\overline{\rho})} \cap V_j) = V_j$ and $\mathtt{l}_j N \leq |V_j| \leq \mathtt{u}_j N$. We next consider the number of edges in the subgraph of G induced by V_j. Since G may miss at most $\frac{\epsilon}{2} \cdot N^2$ edges of the extremal graph, the number of edges in that induced subgraph is at least

$$\sum_{(i,i)\in F^{(\overline{\rho})}} \frac{|V_{i,j}|^2}{2} + \sum_{(i,i')\in F^{(\overline{\rho})}\setminus\{(i,i):i\in[k]\}} |V_{i,j}| \cdot |V_{i',j}| \;-\; \frac{\epsilon}{2} \cdot N^2$$

But this number must be at most $\mathtt{u}_{j,j}N^2$ (because the witness partition of G respects all bounds of Π). Thus, for every j,

$$\frac{1}{2} \cdot \sum_{(i,i)\in F^{(\overline{\rho})}} x_{i,j}^2 + \sum_{(i,i')\in F^{(\overline{\rho})}\setminus\{(i,i):i\in[k]\}} x_{i,j}x_{i',j} \;\leq\; \mathtt{u}_{j,j} + \frac{\epsilon}{2} \;<\; \mathtt{u}_{j,j} + \epsilon$$

Similarly, for $j < j'$, the number of edges (in G) having one endpoint in V_j and the other endpoint in $V_{j'}$ is at least

$$\sum_{(i,i')\in F^{(\overline{\rho})}} |V_{i,j}| \cdot |V_{i',j'}| - \frac{\epsilon}{2} \cdot N^2$$

and it follows that $\sum_{(i,i')\in F^{(\overline{\rho})}} x_{i,j}x_{i',j'} \;\leq\; \mathtt{u}_{j,j'} + \frac{\epsilon}{2}$. This contradicts the above hypothesis that any solution satisfying Eqs. (24)&(25) violates one of the other equations by at least ϵ, and we conclude that the system of Eqs. (24)–(27) has a solution.

Let $(x_{1,1}, ..., x_{k,k})$ be a solution to Eqs. (24)–(27). Then avoiding integrality problems, we consider for each N a partition $(V_1, ..., V_k)$ of the extremal graph $G^{(N,\overline{\rho})}$ so that $V_{i,j}$ is a partition of $V_i^{(N,\overline{\rho})}$ satisfying $|V_{i,j}| = x_{i,j}N$, and $V_j = \cup_i V_{i,j}$. Clearly, this partition satisfies all vertex bounds (because these correspond to Eq. (25)). By the first hypothesis, the only remaining non-trivial bounds are the edge upper-bounds (because all $\mathtt{l}_{i,j}$'s are zero). But these are shown to be satisfied as follows: For every j, the number of edges with both endpoints in V_j equals

$$\begin{aligned}&\sum_{(i,i)\in F^{(\overline{\rho})}} \frac{|V_{i,j}|^2}{2} + \sum_{(i,i')\in F^{(\overline{\rho})}\setminus\{(i,i):i\in[k]\}} |V_{i,j}|\cdot|V_{i',j}| \\ &\quad = \frac{1}{2}\sum_{(i,i)\in F^{(\overline{\rho})}} x_{i,j}^2\cdot N^2 + \sum_{(i,i')\in F^{(\overline{\rho})}\setminus\{(i,i):i\in[k]\}} x_{i,j}\cdot x_{i',j}\cdot N^2 \;\leq\; \mathtt{u}_{j,j}\cdot N^2\end{aligned}$$

where the last inequality is due to Eq. (26). Similarly, for every $j < j'$, the number of edges crossing from V_j to $V_{j'}$ equals

$$\begin{aligned}\sum_{(i,i')\in F^{(\overline{\rho})}} |V_{i,j}|\cdot|V_{i',j'}| &= \sum_{(i,i')\in F^{(\overline{\rho})}} x_{i,j}\cdot x_{i',j'}\cdot N^2 \\ &\leq \mathtt{u}_{j,j'}\cdot N^2\end{aligned}$$

where the inequality is due to Eq. (27). The claim follows. ■

We call a graph iso-extremal for Π' if it is isomorphic to an extremal graph for Π'. In the following claim we rely on the hypothesis that Π does not contain positive lower-bound regarding edges.

Claim D.2 (On subgraphs of graphs that are iso-extremal for Π')

1. *Every N-vertex graph that satisfies Π' is a subgraph of an N-vertex graph that is iso-extremal for Π'.*

2. *Every N-vertex graph that is a subgraph of an N-vertex graph that is iso-extremal for Π', satisfies Π.*

Combining the two parts of the above claim, Lemma 5.5 follows.

Proof: Let $G = ([N], E)$ be a graph satisfying Π', and let $(V_1, ..., V_n)$ be a witness partition. Then the sequence $(\rho_1, ..., \rho_k)$, where $\rho_i = |V_i|/N$ is permitted by Π'. Furthermore, there are edges in the subgraph induced by V_i only if $\mathtt{u}'_{i,i} > 0$ (which holds iff $\mathtt{u}'_{i,i} = 1$). Similarly, there are edges between V_i and V_j only if $\mathtt{u}'_{i,j} = 1$. Part 1 follows by using any isomorphism that maps V_i to $\{\sum_{j=1}^{i-1}|V_j| + 1, ..., \sum_{j=1}^{i}|V_j|\}$.

Turning to Part 2, we first observe that every graph that is iso-extremal for Π' satisfies Π (because, by Claim D.1, every graph that is extremal for Π' satisfies Π, and (being a graph property) Π is preserved under isomorphism). Since all edge lower-bounds in Π are zero, omitting edges does not violate Π. The claim follows. ■

Appendix E: A Result by Noga Alon

In continuation to the discussion in Section 4.2.4, we present the following result of Alon [1].

Definition E.1 *A graph property Π is said to be* closed under taking induced subgraphs *if, for every graph G having property Π, it holds that every induced subgraph of G has property Π.*

For example, k-colorability is closed under taking induced subgraphs, whereas connectivity is not.

Proposition E.2 *Let* Π *be a graph property that is closed under taking induced subgraphs, and supposed that* Π *is testable by* $q(\epsilon)$ *queries, independent of the size of the graph. Then* Π *is testable by inspecting a random induced subgraph of size* $\mathrm{poly}(q(\epsilon))$ *and accepting if and only if the said subgraph has property* Π.

We stress that the tester referred to in the second part of the hypothesis does not necessarily have one-sided error, whereas the tester derived in the conclusion has one-sided error.

Proof: By Theorem 2, Π is testable by inspecting a random induced subgraph of size $s(\epsilon) \stackrel{\mathrm{def}}{=} \mathrm{poly}(q(\epsilon))$ (and accepting if and only if the said subgraph has a property Π', where Π' some graph property). Let us denote by T the tester obtained this way, and assume[16] that its error probability is bounded by $1/4$. We claim that an algorithm, A, that inspects a random induced subgraph of size $s(\epsilon)$ and accepts iff the said subgraph has a property Π is a valid tester for Π.

Clearly, if G has property Π then A accepts it with probability 1 (becuase all induced subgraphs have property Π). We now turn to the case that G is ϵ-far from Π. Our aim is to prove that, with probability at least $2/3$, a random induced subgraph of size $s(\epsilon)$ of G does not have property Π. The main idea, originating in the proof of [7, Cor. 7.2], is that T cannot distinguish the case in which it is given oracle access to G from the case it is given oracle access to a random induced subgraph of size $s(\epsilon)$ of G. It follows that the latter random subgraph (of a graph being far from having property Π) is unlikely to have property Π, or else a contradiction is reached. Details follow.

Let S be a random set of $s(\epsilon)$ vertices in G, and let G_S denote the corresponding induced subgraph. Consider what happens when T is invokes with oracle access to G_S: If G_S has property Π then the tester T must accept with probability at least $3/4$. Thus,

$$\mathsf{Pr}_S[T^{G_S}(\epsilon)=1] \;\geq\; \frac{3}{4}\cdot \mathsf{Pr}_S[G_S \in \Pi] \tag{28}$$

Now consider what happens when T is invoked with oracle access to G. On one hand, since G is ϵ-far from Π, the tester T must reject with probability at least $3/4$. On the other hand, all that T does is select a random S of size $s(\epsilon)$ and inspect G_S. But inpsecting G_S (as an induced subgraph of itself), for a random S of size $s(\epsilon)$, is exactly what T does when given oracle access to G_S. Thus,

$$\mathsf{Pr}_S[T^{G_S}(\epsilon)=1] \;=\; \mathsf{Pr}[T^{G}(\epsilon)=1] \;\leq\; \frac{1}{4} \tag{29}$$

Combining Eqs. (28) and (29), it follows that $\mathsf{Pr}_S[G_S \in \Pi] \leq 1/3$. The claim follows. ■

[16] This assumption can be justified by first reducing the error probability of the original tester.