Testing Subgraphs in Directed Graphs*

Noga Alon †　　Asaf Shapira ‡

Abstract

Let H be a fixed directed graph on h vertices, let G be a directed graph on n vertices and suppose that at least ϵn^2 edges have to be deleted from it to make it H-free. We show that in this case G contains at least $f(\epsilon, H)n^h$ copies of H. This is proved by establishing a directed version of Szemerédi's regularity lemma, and implies that for every H there is a *one-sided error* property tester whose query complexity is bounded by a function of ϵ only for testing the property P_H of being H-free.

As is common with applications of the undirected regularity lemma, here too the function $1/f(\epsilon, H)$ is an extremely fast growing function in ϵ. We therefore further prove a precise characterization of all the digraphs H, for which $f(\epsilon, H)$ has a polynomial dependency on ϵ. This implies a characterization of all the digraphs H, for which the property of being H-free has a one sided error property tester whose query complexity is polynomial in $1/\epsilon$. We further show that the same characterization also applies to two-sided error property testers as well. A special case of this result settles an open problem raised by the first author in [1]. Interestingly, it turns out that if P_H has a polynomial query complexity, then there is a two-sided ϵ-tester for P_H that samples only $O(1/\epsilon)$ vertices, whereas any one-sided tester for P_H makes at least $(1/\epsilon)^{d/2}$ queries, where d is the average degree of H. We also show that the complexity of deciding if for a given directed graph H, P_H has a polynomial query complexity, is NP-complete, marking an interesting distinction from the case of undirected graphs.

For some special cases of directed graphs H, we describe very efficient one-sided error property-testers for testing P_H. As a consequence we conclude that when H is an undirected bipartite graph, we can give a one-sided error property tester with query complexity $O((1/\epsilon)^{h/2})$, improving the previously known upper bound of $O((1/\epsilon)^{h^2})$. The proofs combine combinatorial, graph theoretic and probabilistic arguments with results from additive number theory.

*A preliminary version of this paper appeared in the Proc. of the 35^{th} Annual ACM Symposium on Theory of Computing (STOC), ACM Press (2003), pp. 700-709.

†Schools of Mathematics and Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel. Email: noga@math.tau.ac.il. Research supported in part by a USA-Israeli BSF grant, by the Israel Science Foundation and by the Hermann Minkowski Minerva Center for Geometry at Tel Aviv University.

‡School of Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel. Email: asafico@math.tau.ac.il. This work forms part of the author's Ph.D. Thesis. Research supported by the Deutsch institute.

1 Preliminaries

1.1 Definitions

All directed graphs (=digraphs) considered here are finite and have no loops and no parallel directed edges. They may have anti-parallel edges, i.e., directed cycles of length 2, or in short, *2-cycles*. We call a cycle obtained from an undirected cycle by directing its edges an *oriented cycle*. An oriented cycle in which all edges point to the same direction is a *directed cycle*. *Oriented paths* and *directed paths* are defined in an analogous manner. A digraph is an *oriented tree* if it does not contain any oriented cycle. A digraph is *bipartite* if it does not contain any oriented cycle of odd length.

Let P be a property of digraphs, that is, a family of digraphs closed under isomorphism. A digraph G with n vertices is *ϵ-far from satisfying* P if no digraph $\tilde{G}$ with the same vertex set, which differs from G in at most ϵn^2 places, (i.e., can be constructed from G by adding and removing at most ϵn^2 directed edges), satisfies P. An *ϵ-tester*, or *property tester*, for P is a randomized algorithm which, given the quantity n and the ability to make queries whether a desired pair of vertices of an input digraph G with n vertices are adjacent or not, distinguishes with probability at least $\frac{2}{3}$ between the case of G satisfying P and the case of G being ϵ-far from satisfying P. Such an ϵ-tester is a *one-sided* ϵ-tester if when G satisfies P the ϵ-tester determines that this is the case (with probability 1). The ϵ-tester is a *two-sided* ϵ-tester if it may determine that G does not satisfy P even if G satisfies it. Obviously, the probability $\frac{2}{3}$ appearing above can be replaced by any constant larger than 1/2, by repeating the algorithm an appropriate number of times.

The property P is called *strongly-testable*, if for every fixed $\epsilon > 0$ there exists a one-sided ϵ-tester for P whose total number of queries is bounded only by a function of ϵ, which is independent of the size of the input digraph. This means that the running time of the algorithm is also bounded by a function of ϵ only, and is independent of the input size.

1.2 Related work

The general notion of property testing was first formulated by Rubinfeld and Sudan [32], who were motivated mainly by its connection to the study of program checking. The study of the notion of testability for combinatorial objects, and mainly for labelled graphs, was introduced by Goldreich, Goldwasser and Ron [24], who showed that all graph properties describable by the existence of a partition of a certain type, and among them k-colorability, have efficient ϵ-testers. The fact that k-colorability is strongly testable is, in fact, implicitly proven already in [16] for $k = 2$ and in [30] (see also [2]) for general k, using the Regularity Lemma of Szemerédi [33], but in the context of property testing it is first studied in [24], where far more efficient algorithms are described. These have been further improved in [7].

In [5] it is shown that every first order graph property without a quantifier alternation of type "$\forall\exists$" has ϵ-testers whose query complexity is independent of the size of the input graph (but has a huge dependence on ϵ). In [1] it is shown that there is a one-sided error ϵ-tester for checking H-freeness for **undirected graphs** H, whose query complexity is polynomial in $1/\epsilon$, **if and only if** H is bipartite.

The notion of property testing has been investigated in other areas as well, including the context of regular languages, [6], functions [23], [9], [3], computational geometry [18], [4] as well as graph and hypergraph coloring [17], [9], [15]. See [31] and [22] for surveys on the topic.

2 The Main Results

For a fixed connected digraph H (with at least one edge), let P_H denote the property of being H-free. Therefore, G satisfies P_H if and only if it contains no (not necessarily induced) subgraph isomorphic to H. Our first result is that for each fixed digraph H, the property P_H is strongly-testable.

Theorem 1 *For every fixed digraph H, the property P_H is strongly-testable.*

The proof relies on a variant of the regularity lemma of Szemérédi [33] adapted for directed graphs, which we formulate and prove. This version of the regularity lemma might prove useful for other problems. The application for getting the strong-testability of each property P_H is similar to the proof for the undirected case, given (implicitly) in [2], see also [5], [1].

The one-sided ϵ-tester for P_H for arbitrary digraphs H, has query-complexly bounded by a function which, though independent of the size of the input digraph G, has a huge dependency on ϵ and the size of H. For some digraphs H, however, there are more efficient ϵ-testers; for example, if H is a single directed edge, it is easy to see that there is a one-sided ϵ-tester for P_H, which makes only $\Theta(1/\epsilon)$ queries. A natural question is therefore, to decide for which digraphs H can one design a one-sided error property tester for P_H, whose query complexity would be bounded by a polynomial in $1/\epsilon$. In what follows we call P_H *easily testable* if there is a one-sided error property-tester for P_H whose query complexity is polynomial in $1/\epsilon$. If such a property tester does not exist we say that P_H is *hard to test.*

Our main result here is a precise characterization of all digraphs H for which P_H is easily testable. We further show that the same characterization applies to two-sided error ϵ-testers as well. As a special case of the argument we conclude that for an undirected graph H, P_H has a two-sided ϵ-tester whose query complexity is polynomial in $1/\epsilon$ if and only if H is bipartite. This settles an open problem raised in [1]. Somewhat surprisingly, it turns out that if P_H is easily testable, then it has a two-sided error property-tester that samples only $\Theta(1/\epsilon)$ vertices, although any one-sided error ϵ-tester for P_H has to sample at least $(1/\epsilon)^{d/2}$ vertices, where d is the average degree of H.

The characterization of the digraphs H, for which P_H is easily testable, relies on some properties of digraph homomorphisms and cores of digraphs. Let H and K be two digraphs. A function φ mapping vertices of H to vertices of K is a *homomorphism* if it satisfies $(u, v) \in E(H) \Rightarrow (\varphi(u), \varphi(v)) \in E(K)$.

The *core* of a digraph H is the *subgraph* K of H with the smallest number of edges, for which there is a homomorphism from H to K. We can clearly assume that the core does not contain isolated vertices. It is also easy to see that this notion is well defined in the sense that up to isomorphism the core is unique. We refer the reader to [13] and [28] for more background and references on digraph homomorphisms, and to [27] for more information and references on cores of graphs. Our main result is the following precise characterization of the digraphs H for which testing P_H with one-sided error, has query complexity polynomial in $1/\epsilon$. Here, and throughout the paper, we measure query-complexity by the number of vertices sampled, assuming we always examine all edges spanned by them.

Theorem 2 *Let H be a fixed connected digraph on h vertices, and let K be its core.*
(i) If K is a 2-cycle, then for every $\epsilon > 0$, there is a one-sided error ϵ-tester for P_H whose query-complexity is bounded by

$$O((1/\epsilon)^{h/2}).$$

(ii) If K is an oriented tree, then for every $\epsilon > 0$ there is a one-sided error ϵ-tester for P_H whose query-complexity is bounded by

$$O((1/\epsilon)^{h^2}).$$

(iii) If H is not as in (i), (ii), then there exists a constant $c = c(H) > 0$ such that the query-complexity of any one-sided error ϵ-tester for P_H is at least

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

A special case of the first part of the above theorem improves the previous result from [1] which had query complexity $O((1/\epsilon)^{h^2})$.

We also prove the following theorem, that says that in case H is a tree, we can design an optimal ϵ-tester for P_H.

Theorem 3 *If H is an oriented tree, then there is a one-sided error ϵ-tester for P_H, with optimal query complexity*

$$\Theta(1/\epsilon).$$

The result in the last part of Theorem 2 can be extended to two-sided error ϵ-testers as well.

Theorem 4 *Let H be a fixed digraph on h vertices, and let K be its core.*
(i) If K is a 2-cycle or an oriented tree, then the property P_H has a two-sided error ϵ-tester with optimal query complexity

$$\Theta(1/\epsilon).$$

(ii) If K is neither a directed 2-cycle, nor an oriented tree, then there exists a constant $c = c(H) > 0$ such that the query-complexity of any two-sided error ϵ-tester for P_H is at least

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

It is not difficult to show, by considering an appropriate random digraph, that the one-sided error query complexity of P_H for any digraph H with average degree d is at least $(\frac{1}{\epsilon})^{d/2}$. Therefore, the first part of the theorem exhibits an interesting difference between the query complexity of the best one-sided and the best two-sided error ϵ-testers of P_H for many digraphs H. The second part of Theorem 4 implies a similar result for undirected non bipartite graphs, thus solving a problem raised in [1].

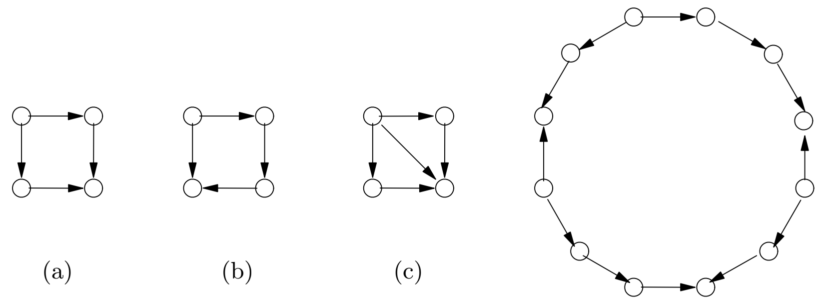
As is apparent from the statement of Theorem 2, the characterization of the digraphs H for which P_H is easily testable, is far more complicated than the characterization for undirected graphs, which states that P_H is easily testable if and only if H is bipartite. The characterization for undirected graphs is also simple in the sense that one can check it in polynomial time. It turns out that the characterization for digraphs is not complicated by chance, and in fact we show that the problem of deciding whether for a given digraph H, the property P_H is easily testable, is NP-complete. This fact follows easily by combining Theorem 2 with a theorem of Hell, Nesetril, and Zhu [28] about cores of digraphs.

Note, that although this implies that the problem of deciding if P_H is easily testable is hard for large digraphs H, this problem is interesting for small fixed digraphs as well, and for those the decision is simple. Thus, for example, Theorem 2 implies that the property P_C has a *polynomial* query complexity in $1/\epsilon$ for the oriented cycle C on the vertices $v_1, \ldots, v_{2k}$, that consists of two edge-disjoint directed paths from v_1 to v_{k+1} (see Figure 1 (a)), as each path is a core of C. Theorem 2 also implies that the property $P_{C'}$ has a *non-polynomial* query complexity in $1/\epsilon$ for every oriented cycle C' that is obtained from the above cycle C, by changing the direction of *any* single edge (see Figure 1 (b)), because in this case the core of C' is the entire digraph. This example shows that the testability of P_H does not rely solely on the structure of H as an undirected graph. Additional comments on this subject appear in Section 8.

2.1 Organization

The paper is organized as follows: In Section 3, we modify some of the ideas used in the proof of Szemerédi's regularity lemma for undirected graphs, in order to prove a more general result that applies also to digraphs. In Section 4 we apply the above lemma in order to prove Theorem 1.

The main result consists of two parts. The first one (Theorem 2, parts (i),(ii)) appears in Section 5, and is proved using probabilistic arguments and tools from extremal graph theory. Unlike the corresponding result for undirected graphs, the techniques required here are rather complicated, and



(a) (b) (c) (d)

Figure 1: (a) Core is a path (b) Core is the entire digraph (c) Core is a triangle (d) Core is the entire digraph although the graph is balanced.

apply some delicate arguments. In this section we also prove Theorem 3. To prove the third part of Theorem 2, we have to construct, for any digraph H as in (iii) and any small $\epsilon > 0$, a digraph G which is ϵ-far from being H-free and yet contains relatively few copies of H. The proof of this part, described in Section 6, uses the approach of [1], but requires some additional ideas. It applies some properties of digraph homomorphisms as well as certain constructions in additive number theory, based on (simple variants of) the construction of Behrend [14] of dense subsets of the first n integers without three-term arithmetic progressions. In Section 7 we describe the proof of Theorem 4. We assume, throughout these three sections, that the underlying undirected graph of the digraph H considered is connected. In the final section, Section 8, we observe that it is easy to extend the results to the disconnected case and discuss the complexity of the problem of deciding whether for a given input digraph H, P_H is polynomially testable. This final section contains some concluding remarks and open problems as well.

Throughout the paper we assume, whenever this is needed, that the number of vertices n of the digraph G is sufficiently large, and that the error parameter ϵ, is sufficiently small. In order to simplify the presentation, we omit all floor and ceiling signs whenever these are not crucial, and make no attempt to optimize the absolute constants.

3 A Regularity Lemma for Digraphs

3.1 Statement of the Lemma

In this section we prove a regularity lemma for digraphs, by using some of the ideas in the proof of Szemerédi's regularity lemma for undirected graphs. For the proof of Szemerédi's regularity lemma the reader is referred to the original proof in [33], and to [19] which was used as a reference for the proof here. In order to state the lemma we need some definitions. Let $G = (V, E)$ be a digraph, and let $X, Y \subseteq V$ be disjoint. Let $\overrightarrow{E}(X, Y)$ denote the set of edges going from X to Y, and let $\overleftarrow{E}(X, Y)$ denote the set of edges going from Y to X. Let $\overline{E}(X, Y)$ denote the set of pairs of edges that form 2-cycles between X and Y. Define

$$\overrightarrow{d}(X, Y) := \frac{|\overrightarrow{E}(X, Y)|}{|X||Y|}, \qquad \overleftarrow{d}(X, Y) := \frac{|\overleftarrow{E}(X, Y)|}{|X||Y|}, \qquad \overline{d}(X, Y) := \frac{|\overline{E}(X, Y)|}{|X||Y|}$$

the *directed densities* of the pair (X, Y). Observe that all three densities of any pair are real numbers between 0 and 1. Given some $\epsilon > 0$, we call a pair (A, B) of disjoint sets $A, B \subseteq V$ ϵ-*regular* if all $X \subseteq A$ and $Y \subseteq B$ with

$$|X| \geq \epsilon|A| \text{ and } |Y| \geq \epsilon|B|,$$

satisfy

$$|\overrightarrow{d}(X, Y) - \overrightarrow{d}(A, B)| \leq \epsilon, \qquad |\overleftarrow{d}(X, Y) - \overleftarrow{d}(A, B)| \leq \epsilon, \qquad |\overline{d}(X, Y) - \overline{d}(A, B)| \leq \epsilon.$$

We will later need the following trivial claim about a regular pair (A, B). The claim simply says that if we take a large enough subset $Y \subseteq B$, then for most vertices in the other side, Y behaves almost like B. In order to state the claim we need the following notation which will be used later as well: $\overrightarrow{N}_Y(v)$ is the set of vertices $y \in Y$ for which $(v, y) \in E$, $\overleftarrow{N}_Y(v)$ is the set of vertices $y \in Y$ for which $(y, v) \in E$ and $\overline{N}_Y(v)$ is the set of vertices $y \in Y$ for which (v, y) is a 2-cycle.

Claim 3.1 *Let (A, B) be an ϵ-regular pair with densities $\overrightarrow{d}, \overleftarrow{d}$ and $\overline{d}$, and let $Y \subseteq B$ be of size at least $\epsilon|B|$. Then for all but at most $3\epsilon|A|$ vertices $v \in A$, the inequalities $\overrightarrow{N}_Y(v) \geq (\overrightarrow{d} - \epsilon)|Y|$, $\overleftarrow{N}_Y(v) \geq (\overleftarrow{d} - \epsilon)|Y|$ and $\overline{N}_Y(v) \geq (\overline{d} - \epsilon)|Y|$ hold.*

Proof. Assume that for some X, such that $|X| \geq 3\epsilon|A|$, for all $v \in X$ at least one of the inequalities does not hold. Then for some $Z \subseteq X$, such that $|Z| \geq \epsilon|A|$, for all $v \in Z$ the same inequality does not hold. Hence, the pair (Z, Y) contradicts the ϵ-regularity of the pair (A, B). ∎

Consider a partition $\{V_0, V_1, \ldots, V_k\}$ of V in which one set V_0 has been singled out as an exceptional set (V_0 may be empty). We call such a partition an ϵ-*regular partition* of a digraph G if it satisfies the following three conditions:

(i) $|V_0| \leq \epsilon|V|$;

(ii) $|V_1| = \ldots = |V_k|$;

(iii) all but at most ϵk^2 of the pairs (V_i, V_j) with $1 \leq i < j \leq k$ are ϵ-*regular*.

Our objective is to prove the following generalization of Szemerédi's regularity lemma.

Lemma 3.1 *For every $\epsilon > 0$ and every $m \geq 1$ there exists an integer $DM = DM(m, \epsilon)$ such that every digraph of order at least m admits an ϵ-regular partition $\{V_0, V_1, \ldots, V_k\}$ with $m \leq k \leq DM$.*

The statement of the lemma for symmetric digraphs, that is, digraphs in which (u, v) is a directed edge if and only if (v, u) is a directed edge, is equivalent to the statement of the regularity lemma for undirected graphs.

3.2 The Regularity Lemma for Undirected Graphs

We start with the regularity lemma for undirected graphs, and some of the definitions used in the course of its proof. In the context of undirected graphs there is only one density between a pair of disjoint subsets $A, B \subseteq V$, and it is defined as $d(A, B) := |E(A, B)|/|A||B|$, where $E(A, B)$ is the set of edges between A and B. A pair of disjoint sets $A, B \subseteq V$ is *ϵ-regular* if all $X \subseteq A$ and $Y \subseteq B$ with $|X| \geq \epsilon|A|$ and $|Y| \geq \epsilon|B|$, satisfy $|d(X, Y) - d(A, B)| \leq \epsilon$.

An *ϵ-regular partition* is defined in a way analogous to the definition of a regular partition for digraphs. The following is Szemerédi's regularity lemma for undirected graphs

Lemma 3.2 **[33]** *For every $\epsilon > 0$ and every $m \geq 1$ there exists an integer $M = M(m, \epsilon)$ such that every graph of order at least m admits an ϵ-regular partition $\{V_0, V_1, \ldots, V_k\}$ with $m \leq k \leq M$.*

The proof for undirected graphs uses the following definitions that will be used in our proof as well. Let $G = (V, E)$ be a graph and $n = |V|$. For disjoint sets $A, B \subseteq V$ we define

$$q(A, B) = \frac{|A||B|}{n^2} d^2(A, B).$$

For a partition $P = \{C_1, \ldots, C_k\}$ of V we let

$$q(P) = \sum_{i<j} q(C_i, C_j).$$

However, if $P = \{C_0, C_1, \ldots, C_k\}$ has an exceptional set C_0, we treat C_0 as a set of singletons and define

$$q(P) = q(P'),$$

where $P' = \{C_1, \ldots, C_k\} \cup \{\{v\} : v \in C_0\}$.

It can be easily shown that for any partition P,

$$q(P) \leq \frac{1}{2}. \tag{1}$$

We say that a partition P' *refines* a partition P, if any (non exceptional) set in P is the union of some sets in P'. We will also need the following lemmas from [19] that establish relations between partitions and their refinements.

Lemma 3.3 *If P and P' are partitions of V, and P' refines P, then $q(P') \geq q(P)$.*

Lemma 3.4 *Let $0 \leq \epsilon \leq 1/4$ and let $P = \{C_0, C_1, \ldots, C_k\}$ be a partition of V, with exceptional set C_0, of size $|C_0| \leq \epsilon n$ and $|C_1| = \ldots = |C_k|$. If P is not ϵ-regular, then there is a partition $P' = \{C'_0, C'_1, \ldots, C'_\ell\}$ of V with exceptional set C'_0, where $k \leq \ell \leq k4^k$, such that $|C'_0| \leq |C_0| + n/2^k$,$C_0 \subseteq C'_0$, all other sets C'_i have equal size, and*

$$q(P') \geq q(P) + \epsilon^5/2.$$

Comment: Although the above claim in [19] does not explicitly state it, the partition P' is a refinement of P.

Note that combining Lemma 3.4 with (1), the proof of the regularity lemma for undirected graphs is immediate (up to some technicalities). We can apply Lemma 3.4 over and over again until we get an ϵ-regular partition. This must happen after at most $1/\epsilon^5$ iterations.

3.3 The Proof of Lemma 3.1

Given a digraph $G = (V, E)$, and a partition of V, $P = \{C_1, \ldots, C_k\}$, consider a partition of E into 3 (not necessarily disjoint) sets

$$\overrightarrow{E} = \{(u, v) \in E : u \in C_i, v \in C_j, i < j\},$$

$$\overleftarrow{E} = \{(u, v) \in E : u \in C_i, v \in C_j, i > j\},$$

$$\overline{E} = \{(u, v) \in E : (v, u) \in E, u \in C_i, v \in C_j, i \neq j\}.$$

Now we can view a partition P as three different partitions $\overrightarrow{P}, \overleftarrow{P}, \overline{P}$, of undirected graphs (all three partition V in the same way, but the sets of edges among the partition sets are different). The first is obtained by removing any edge that does not belong to $\overrightarrow{E}$, and considering the directed edges as undirected. The second is obtained by removing any edge that does not belong to $\overleftarrow{E}$, and again considering the directed edges as undirected. The third is obtained by removing any edge that does not belong to $\overline{E}$, and considering each cycle of length 2 as an undirected edge. We can also define the values $q(\overrightarrow{P})$, $q(\overleftarrow{P})$ and $q(\overline{P})$, as the function $q(\cdot)$ on a partition of V with edge sets $\overrightarrow{E}$, $\overleftarrow{E}$ and $\overline{E}$ respectively, by considering the directed edges and cycles of length 2, as undirected edges.

The key observation now, is that if the above three partitions are ϵ-regular in the context of undirected graphs, then P is an ϵ-regular partition in the context of directed graphs. Thus we can view the task of obtaining an ϵ-regular partition in a digraph, as the task of obtaining a partition that is ϵ-regular in the sense of undirected graphs, over three subsets of E. We next refer to $\overrightarrow{P}$, $\overleftarrow{P}$ and $\overline{P}$ sometimes not as a specific partition, but as the *set* of partitions of $\overrightarrow{E}$, $\overleftarrow{E}$ and $\overline{E}$ respectively, obtained in the course of creating the ϵ-regular partition.

Proof (of Lemma 3.1): Let $G = (V, E)$ be given. For any partition P of V, we can define the partitions $\overrightarrow{P}, \overleftarrow{P}$ and $\overline{P}$ as described above. Also note that all three values $q(\overrightarrow{P}), q(\overleftarrow{P})$ and $q(\overline{P})$ are always at most 1/2 by (1). Thus we can apply Lemma 3.4, circularly once for each partition until all three are ϵ-regular. For example, when we apply Lemma 3.4 to $\overrightarrow{E}$, we choose a new partition of V, according to the previous $\overrightarrow{P}$, and this induces a new partition of $\overleftarrow{P}$ and $\overline{P}$ as well. By the condition of Lemma 3.4 and the comment following it, this cannot happen more than $s = 3 \cdot 1/\epsilon^5 = 3/\epsilon^5$ times, before we obtain an ϵ-regular partition of the digraph G. Observe that, for example, when we apply Lemma 3.4 to $\overrightarrow{P}$, we do not necessarily increase $q(\overleftarrow{P})$ by $\epsilon^5/2$ (In fact, it might even be the case that $\overleftarrow{P}$ was an ϵ-regular partition of $\overleftarrow{E}$ and now it is not!), but by Lemma 3.3 and the comment following Lemma 3.4, we also do not decrease its value. Hence, in each iteration one of the values $q(\overrightarrow{P}), q(\overleftarrow{P}), q(\overline{P})$ is increased by at least $\epsilon^5/2$, while the other two do not decrease. An important technicality is that as the definitions of the partitions $\overrightarrow{P}, \overleftarrow{P}$ and $\overline{P}$ depend on the serial numbers given to the partition sets of $V(G)$ (see beginning of the subsection), we must make sure that if, for example, edge (u, v) was part of partition $\overrightarrow{P}$ then it does not "move" to another partition, say, $\overleftarrow{P}$. To this end, we can simply give consecutive serial numbers in the new partition, to all the subsets of a set that belongs to the previous partition.

We are left only with the simple technicalities of making sure that C_0 does not get too large, and of defining the function $DM(m, \epsilon)$. These are straightforward, and are left to the reader. See ,e.g., [19] pages 159-160. ■

Note that our process for obtaining the regular partition does *not* apply the regularity lemma for undirected graphs recursively, and that the bound for the function $DM(\epsilon, k)$ in the lemma for digraphs is similar to the bound of the function $M(\epsilon, k)$ in the lemma for undirected graphs, that is, both are towers of 2's of height $O(1/\epsilon^5)$. By a result of Gowers [26], both functions must grow at least as fast as a tower of 2's of height $poly(1/\epsilon)$.

4 Testing for Arbitrary Subgraphs

In this section we use our version of Szemerédi's regularity lemma, Lemma 3.1 from the previous section, in order to prove Theorem 1. To this end, we prove the following lemma, which is similar to previously known results for undirected graphs. See, for example, Theorem 2.1 in [29], and Lemma 3.2 in [5].

Lemma 4.1 *For every fixed ϵ and h, there is a positive constant $c(h, \epsilon)$ with the following property: for every fixed digraph H of size h, and for every digraph G of a large enough size n that is ϵ-far from being H-free, G contains at least $c(h, \epsilon)n^h$ copies of H.*

Proof. Let ϵ_1 be a constant whose value will be decided later. On inputs $1/\epsilon_1$ and ϵ_1, Lemma 3.1 returns an ϵ_1-regular partition with $|V_0| \leq \epsilon_1 n$ and partition sets $V_1, \ldots, V_t$, $|V_i| = k$ such that $1/\epsilon_1 \leq t \leq DM(1/\epsilon_1, \epsilon_1)$. Obtain from G the digraph G' by removing the following sets of edges:

- Edges that touch V_0. There are at most $(\epsilon_1 n)^2 + 2\epsilon_1 n^2 < 3\epsilon_1 n^2$ edges of this type.
- Edges within some set V_i. There are at most $t(n/t)^2 = n^2/t \leq \epsilon_1 n^2$ such edges.
- Edges between non ϵ_1-regular sets. There are at most $\epsilon_1 t^2 \cdot 2n^2/t^2 \leq 2\epsilon_1 n^2$ such edges.
- If for some pair of partition sets, one of the densities $\overrightarrow{d}, \overleftarrow{d}, \overline{d}$ is less than $\epsilon/4$, remove all corresponding edges (i.e. all edges that define that density). There are at most $\binom{t}{2}\epsilon n^2/t^2 \leq \epsilon n^2/2$ such edges.

Altogether we have removed less than $\epsilon n^2/2 + 6\epsilon_1 n^2$ edges from G. Thus, as G is ϵ-far from being H-free, for *any* $\epsilon_1 \leq \epsilon/13$ the digraph G' is obtained from G by removing less that ϵn^2 edges, and therefore still contains a copy of H. Moreover, for each directed edge (u, v) in H, u and v belong to an ϵ_1-regular pair $(U, V), u \in U, v \in V$, such that $\overrightarrow{d}(U, V) \geq \epsilon/4$. The same applies to a pair of edges $(u, v), (v, u)$ in H but this time with respect to the density $\overline{d}(U, V)$.

Having established the existence of one such H, we show that there are actually many more copies of H, provided that ϵ_1 is sufficiently small. Let $u_1, \ldots, u_h$ be the vertices of the copy of H in G, and assume that $u_i \in V_{\sigma(i)}$. We wish to show that for a small enough $\epsilon_1 \leq \epsilon/13$ we can build $c(h, \epsilon_1)n^h$ copies of H, where for each copy, u_i will belong to $V_{\sigma(i)}$. This would imply the lemma.

For our scheme to work we need to take $\epsilon_1 \leq \epsilon/13$ small enough that it satisfies,

$$(3h + 1)\epsilon_1 \leq (\epsilon/4 - \epsilon_1)^h. \tag{2}$$

Note, that we must also take $\epsilon_1 \leq \epsilon/13$ so that we will be able to assume the properties of G' discussed above. Also, note that the value of ϵ_1 is a function of ϵ and h only, and is independent of n.

The idea is to build the copies iteratively, where in iteration $1 \leq i \leq h$, we find many candidates to play the role of u_i. To this end, we keep a set $C_{i,j} \subseteq V_{\sigma(i)}$, which includes the vertices that may play the role of u_i after we have already found vertices for $u_1, \ldots, u_j$. Initially, $C_{i,0} = V_{\sigma(i)}$, $|C_{i,0}| = k$. Consider the stage when we come to select the vertices that will play the role of u_j. When we select a vertex to be u_j we have to update the sets $C_{i,j}$. For example, if for $i > j$ (u_j, u_i) is an edge of H, then after selecting v to be u_j we have to update $C_{i,j} = \overrightarrow{N}_{C_{i,j-1}}(v)$. The updates are equivalent for the other two cases where there is an edge (u_i, u_j) and when there are two edges $(u_i, u_j), (u_j, u_i)$.

The crucial observation now, is that we made sure that all edges of H go between ϵ_1-regular pairs, and moreover we have a relatively high density in the direction of these edges. Therefore, if

$|C_{i,j-1}| \geq \epsilon_1 |V_{\sigma(i)}|$ then by Claim 3.1 all but at most $3\epsilon_1|V_{\sigma(j)}|$ vertices in $V_{\sigma(j)}$ are such that the three inequalities of Claim 3.1 hold (with $d = \epsilon/4$ and $\epsilon = \epsilon_1$). That is,

$$|C_{i,j}| \geq (\epsilon/4 - \epsilon_1)|C_{i,j-1}|. \tag{3}$$

As H contains h vertices, and each $i > j$ excludes at most $3\epsilon_1|V_{\sigma(j)}|$ from being u_j, then altogether we have at least $|C_{j,j-1}| - 3\epsilon_1 h|V_{\sigma(j)}|$ candidates for the role of u_j. For our scheme to work we must make sure that $|C_{i,j}| \geq \epsilon_1|V_{\sigma(i)}|$ so that we may apply Lemma 3.1. But, by our previous assumptions the following holds for any $i > j$,

$$|C_{i,j}| - 3\epsilon_1 h|V_{\sigma(j)}| \geq (\epsilon/4 - \epsilon_1)^h k - 3\epsilon_1 hk \geq \epsilon_1 k.$$

The first inequality follows from (3) and the second from (2). We thus get that $|C_{i,j}| \geq \epsilon_1 k = \epsilon_1|V_{\sigma(i)}|$ as needed. In particular $|C_{j,j-1}| \geq \epsilon_1 k$, thus we have $\epsilon_1 k$ choices when we come to choose u_j. Finally as Lemma 3.1 partitions V into a constant number of sets we get that,

$$k = \frac{n - |V_0|}{t} \geq \frac{n(1-\epsilon_1)}{DM(1/\epsilon_1, \epsilon_1)}$$

Thus, for each iteration i, we have at least

$$\epsilon_1 k = \frac{\epsilon_1(1-\epsilon_1)n}{DM(1/\epsilon_1, \epsilon_1)}$$

choices for u_i. Therefore, as ϵ_1 is a function of ϵ and h only by (2), G' contains at least

$$\left(\frac{\epsilon_1(1-\epsilon_1)}{DM(1/\epsilon_1, \epsilon_1)}\right)^h n^h = c(h,\epsilon)n^h$$

copies of H. As G' is a subgraph of G, G contains at least as many copies. ■

The proof of Theorem 1 now follows easily.

Proof (of Theorem 1): The tester simply picks, say, $4/c(h,\epsilon)$ sets of vertices of G, where each set consists of h vertices, at random. If at least one of these sets spans a copy of H, it reports that G is not H-free, else, it declares that G is H-free. If G is H-free, then the algorithm will certainly report that this is the case. If G is ϵ-far from being H-free then, by the above lemma, the algorithm will find a copy of H with probability at least 2/3. ■.

5 Easily Testable Digraphs

In this section we prove parts (i) and (ii) of Theorem 2 as well as Theorem 3. We first show that the property of being H-free is easily testable, whenever the core of H is a 2-cycle. We then prove the same for all digraphs H for which the core of H is a tree. In Section 6 we show that for any other digraph H, the property of being H-free is hard to test.

We next prove that if the core of a digraph H is a 2-cycle, then testing H-freeness has query complexity polynomial in $1/\epsilon$. Observe, that the core of a digraph cannot be a bipartite digraph with at least one 2-cycle, and not be a 2-cycle, because there is a homomorphism from any such digraph to a 2-cycle.

Proof of Theorem 2, part (i) Let H be a bipartite digraph with at least one 2-cycle, with color classes of size s and t, and assume $s \leq t$. Our tester samples some c/ϵ^s vertices, for an appropriate $c = c(s,t)$, and reports that G is not H-free if and only if there is a copy of H spanned by a subset of these vertices. Clearly, if G is H-free, the algorithm will report this is the case. If G is ϵ-far from being H-free it must contain at least ϵn^2 cycles of length 2, as otherwise we can remove an edge from each of these 2-cycles and obtain an H-free digraph (using the fact that H contains a 2-cycle), while removing less than ϵn^2 edges. Now, consider the undirected graph G', obtained from G by putting an edge (u,v) in G' if and only if (u,v) is a 2-cycle in G. We show how to find in G' a set of vertices that spans a copy of $K_{s,t}$. From the definition of G', it implies that in G the same set spans a copy of H.

Randomly and independently, pick s vertices (with repetitions). The expected number of vertices that are connected to all the chosen vertices is

$$\sum_v \left(\frac{d_v}{n}\right)^s \geq n\left(\frac{\sum_v d_v}{n^2}\right)^s \geq n(2\epsilon)^s,$$

where d_v is the degree of v, the first inequality follows from convexity of the function x^s, and the second from our assumption that G' contains at least ϵn^2 edges.

It follows that with probability at least $\frac{1}{2}(2\epsilon)^s$, at least $\frac{1}{2}(2\epsilon)^s n$ vertices are adjacent to all the s chosen vertices, as otherwise the expectation would have been smaller than $n(2\epsilon)^s$. Therefore, after $10/(2\epsilon)^s$ rounds in which s vertices are chosen, with probability at least $15/16$ at least $\frac{1}{2}(2\epsilon)^s n$ of the vertices are adjacent to all the s vertices chosen in one of the rounds. Fix these s vertices. If we now choose another vertex, it has probability at least $\frac{1}{2}(2\epsilon)^s$ of being adjacent to all these s vertices. We conclude that the expected number of additional vertices that we need to sample, in order to find t vertices that are connected to the s fixed ones, is at most $2t/(2\epsilon)^s$. By Markov's inequality, after sampling $8t/(2\epsilon)^s$ vertices, the probability of not finding a set of t vertices that is connected to all the s vertices is at most $1/4$. The algorithm has probability at most $1/16$ of failing to find the s vertices in the first step, a probability of at most $1/4$ of failing to find the t vertices in the second step, and a probability of $o(1)$ that in each of the two steps, the chosen set does not consist of distinct vertices (notice that we sampled with repetitions). Altogether, the failure probability is at most $1/3$, hence, the algorithm finds a copy of $K_{s,t}$ with probability at least $2/3$. As for the sample size, the first part uses a sample of size $10s/(2\epsilon)^s$, while the second is of size $8t/(2\epsilon)^s$. Altogether, we use a sample of size $O((1/\epsilon)^s) = O((1/\epsilon)^{h/2})$. This completes the proof of Theorem 1, part (i). ■

Comment: By the above proof, every digraph G on sufficiently many vertices with $\Omega(n^2)$ 2-cycles, contains a copy of every fixed bipartite digraph. Therefore there is a very simple and efficient **two-sided error** algorithm for testing P_H, for every H whose core is a 2-cycle, which simply samples $O(1/\epsilon)$ pairs of vertices and accepts iff they span no edge.

We now proceed with the proof of Theorem 2 part (ii). In the proof we will use the following construction of a digraph G' obtained from a digraph G which is ϵ-far from being H-free. The process is described with respect to some tree K, which is a connected subgraph of H. We therefore denote $G' = G'(G, K)$. The reason to make the description general is that we will later use it with respect to different trees. Let G be a digraph that is ϵ-far from being H-free, and let K be some subtree of H. Let us also name the vertices of K as $1, \ldots, t$. We define the digraph $G' = G'(G, K)$ in the following constructive manner with respect to K: assign each vertex v of G a list $L(v)$ containing the numbers $1, \ldots, t$. This list should eventually contain $i \in \{1, 2, \ldots, t\}$ if and only if there is a homomorphism $\varphi : K \mapsto G'$ in which $\varphi(i) = v$. We also define $N^+(v, i)$ to be the set of vertices u, for which there is an edge (v, u), and $i \in L(u)$. We define $N^-(v, i)$ analogously only with respect to incoming edges into v. The process executes the following two operations while it can: (i) If for some directed edge (i, j) in K, there is a vertex v in G, for which $i \in L(v)$ and $|N^+(v, j)| < \frac{\epsilon}{2t}n$, remove all edges $\{(v, u) : u \in N^+(v, j)\}$, remove i from $L(v)$, and update all the sets $N^-(\cdot, i)$ of vertices in G. (ii) If for some directed edge (i, j) in K, there is a vertex v in G, for which $j \in L(v)$ and $|N^-(v, i)| < \frac{\epsilon}{2t}n$, remove all edges $\{(u, v) : u \in N^-(v, i)\}$, remove j from $L(v)$, and update all the sets $N^+(\cdot, j)$ of vertices in G.

Lemma 5.1 *If G is ϵ-far from being H-free, and K is a connected subgraph of H which is a tree, then the digraph $G' = G'(G, K)$ described above satisfies the following properties: (1) It contains a copy of K. (2) $i \in L(v)$ if and only if there is a homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$.*

Proof. As K is a subgraph of H, and G is ϵ-far from being H-free, we may show that G' satisfies (1), simply by showing that the above process for obtaining G', does so by removing less than ϵn^2 edges. To this end, consider any vertex v. Each execution of items (i) and (ii) removes an element from $L(v)$, therefore we can execute them at most t times on v. As in each execution we remove less than $\frac{\epsilon}{2t}n$ edges, it follows that the process removes less than ϵn edges that touch v, and altogether less than ϵn^2 edges.

To prove (2) we first prove the implication that asserts that if $i \notin L(v)$ then there is no homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$. We proceed by induction on m, the number of steps of the process. At the beginning, all the lists are full, therefore the desired property trivially holds. Assume it holds for m steps and consider step $m + 1$: if we execute (i), then some i was removed from some $L(v)$, after removing all edges that go from v to vertices $N^+(v, j)$ for some j that is a neighbor of i in K. It follows from the induction hypothesis, that no homomorphism can map j to an out-neighbor of v, and therefore, as i and j are neighbours in K, no homomorphism can map i to v. The case of executing (ii) is identical. To prove the second implication, assume that at the end of the process, for some vertex v, we have $i \in L(v)$ but there is no homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$. Let K' be the largest connected subgraph of K that contains i, for which there is a homomorphism $\varphi : K' \mapsto G'$ that satisfies $\varphi(i) = v$ and for all $j \in K'$ $j \in L(\varphi(j))$. As K is connected, there is some vertex $i' \in K'$ that is connected by an edge to $j' \in K \setminus K'$ in K. By the maximality of K', there is no edge connecting $\varphi(i')$ to a vertex q for which $j' \in L(q)$. This is impossible, as it means that the process should have removed i' from $L(\varphi(i'))$. ■

We now turn to the proof of Theorem 2, part (ii). The proof is based on a variant of a powerful probabilistic technique, which may be called *dependent random choice*, and which has already found several recent combinatorial applications. See, e.g., [8] and some of its references. Given a subset of vertices $V_i \subseteq V(G)$ and a vertex $v \in V(G)$, let $N(v, i)$ denote the set of neighbors of v within V_i. We need the following lemma.

Lemma 5.2 *Let $G = (V, E)$ be an undirected graph on n vertices, and let $V_1, V_2, \ldots, V_{d+1}$ be (not necessarily disjoint) subsets of V. Put $\alpha = |V_1|/n$. Assume that for every vertex $v \in V_1$ and for every $2 \le k \le d+1$, $|N(v,k)| \ge \epsilon |V_k|$. Then, sampling $32h\log(1/\delta)/(\alpha\epsilon^d)$ vertices from G, finds with probability at least $1-\delta$, an h-tuple of distinct vertices $s = \{v_1, \ldots, v_h\} \subseteq V_1$, that satisfies*

$$\left|\bigcap_{i=1}^{h} N(v_i, k)\right| \ge \frac{1}{4}\epsilon^{dh}|V_k|, \quad \forall\ 2 \le k \le d+1. \tag{4}$$

Proof. The result is trivial for $h = 1$, and we thus assume that $h \ge 2$. For $2 \le k \le d+1$, choose uniformly and independently a vertex t_k from each set V_k. Let X be the set of vertices $v \in V_1$, for which $t_k \in N(v, k)$ for all $2 \le k \le d+1$. For each $v \in V_1$, let X_v be an indicator random variable for the event that $v \in X$. It follows from the assumption on the large number of neighbours of each vertex of V_1 in each set V_k, that

$$E(|X|) = \sum_{v \in V_1} E(X_v) \ge \epsilon^d |V_1|.$$

By Jensen's inequality, it follows that

$$E(|X|^h) \ge E(|X|)^h \ge \epsilon^{dh}|V_1|^h.$$

Therefore, there is an expected number of at least $\epsilon^{dh}|V_1|^h$ h-tuples $s = (v_1, \ldots, v_h)$ (where the vertices v_i are not necessarily distinct) of vertices in V_1, with the property that $t_k \in N(v_i, k)$, for all $2 \le k \le d+1$ and $1 \le i \le h$. We now turn to show, that the expected number of these h-tuples that violate (4) is small. To this end, define Z to be the set of all h-tuples $s \in V_1^h$, that do not satisfy (4), and let Y be the set of all members of Z that lie in X^h. For each $s \in Z$ let Y_s denote the indicator random variable for the event that $s \in X^h$. Note that $|Y| = \sum_{s \in Z} Y_s$. Thus

$$E(|Y|) = \sum_{s=(v_1,\ldots,v_h)\in Z} E(Y_s) = \sum_{s\in Z}\prod_{k=2}^{d+1}\frac{\left|\bigcap_{i=1}^{h} N(v_i,k)\right|}{|V_k|} \le \sum_{s\in V_1^h}\frac{1}{4}\epsilon^{dh} \le \frac{1}{4}\epsilon^{dh}|V_1|^h,$$

where the first inequality follows from our assumption that for some k, $\left|\bigcap_{i=1}^{h} N(v_i,k)\right| < \frac{1}{4}\epsilon^{dh}|V_k|$. We conclude that,

$$E(\frac{1}{2}|X|^h - |Y|) = \frac{1}{2}E(|X|^h) - E(|Y|) \ge \frac{1}{2}\epsilon^{dh}|V_1|^h - \frac{1}{4}\epsilon^{dh}|V_1|^h = \frac{1}{4}\epsilon^{dh}|V_1|^h.$$

Therefore, there is some choice of $t_2, \ldots, t_{d+1}$, for which the sets X and Y satisfy,

$$|X|^h - |Y| \ge \frac{1}{2}|X|^h + \frac{1}{4}\epsilon^{dh}|V_1|^h.$$

Fix one such choice of $t_2, \ldots, t_{d+1}$. The above inequality implies that more than half of the h-tuples in X^h satisfy (4), and that X is of size at least $\frac{1}{4^{1/h}}\epsilon^d|V_1| \ge \frac{\alpha}{2}\epsilon^d n$. Therefore, a randomly chosen vertex from G, has probability at least $\frac{\alpha}{2}\epsilon^d$ to lie in X. It follows that the expected number of samples needed to find an h-tuple from X is at most $2h/(\alpha\epsilon^d)$. Hence, by Markov's inequality, choosing $8h/(\alpha\epsilon^d)$ random vertices, finds an h-tuple from X with probability at least $\frac{3}{4}$. As at least half of the h-tuples in X^h satisfy (4), it follows that with probability at least $\frac{3}{8}$ we find an h-tuple satisfying (4). This is not necessarily an h-tuple of distinct vertices. But the probability of finding an h-tuple with non distinct vertices is $o(1)$, as $|X| = \Omega(n)$. Therefore with probability at least $\frac{1}{4}$ we find an h-tuple of distinct vertices satisfying (4). Thus, choosing $32h\log(1/\delta)/(\alpha\epsilon^d)$ vertices finds such an h-tuple with probability at least $1-\delta$ as needed. ■

Proof of Theorem 2, part (ii) As in the proof of part (i), (and as can be done for any one-sided property tester for a problem which is closed under taking induced subgraphs), the algorithm simply samples the stated number of vertices randomly and reports that G is H-free if and only if it finds no copy of H on them. Clearly, if G is H-free, the answer is correct. Let G be ϵ-far from being H-free, and let K denote the core of H which is, by assumption, a tree. Number the vertices of K by $1, \ldots, k$ in a BFS order, and let h_i be the number of vertices of H that are mapped to $i \in \{1, 2, \ldots, k\}$. Note that if i and j are neighbors in K, it does not necessarily hold, that all the vertices of H that are mapped to i, are adjacent to all the vertices of H that are mapped to j, but it does hold, that all existing edges are in the same direction. We will show however, that we can find a subgraph of G whose vertex set consists of subsets $|U_1| = h_1, \ldots, |U_k| = h_k$ such that if $(i,j) \in E(K)$ then all the vertices of U_i are connected to all the vertices of U_j. Such a subgraph clearly contains a copy of H.

Let $N(i)$ be the neighbours of vertex i in K, that appear after it in the BFS order, and $d_i = |N(i)|$. Apply the process described before the proof of Lemma 5.1 with respect to K, that is, obtain

$G' = G'(G,K)$. It follows from Lemma 5.1 that G' contains a copy of K. Let $v_1,\ldots,v_k$ be such a copy. By Lemma 5.1, for all $1 \le i \le k$, $i \in L(v_i)$. Denote by V_i the set of vertices u_i for which $i \in L(u_i)$. Clearly $v_i \in V_i$. In order to make the presentation simple, from now until the end of the proof, we will not specify the direction of an edge between $u_i \in V_i$ and $u_j \in V_j$, although we will always be speaking about an edge that is directed as the direction of an edge between i and j in K.

Let $N(1) = \{2,\ldots,d_1+1\}$ be the d_1 neighbors of vertex 1 in K, hence, G' contains the edges $(v_1,v_2),\ldots,(v_1,v_{d_1+1})$. From the definition of the process for obtaining G', it follows that for every $2 \le i \le d_1+1$, there are at least $\frac{\epsilon}{2h}n$ vertices $u_1 \in V_1$, for which there is an edge (u_1,v_i) and $1 \in L(u_1)$, and in particular, $|V_1| \ge \frac{\epsilon}{2h}n$. It follows again from the definition of the process, that for every $u_1 \in V_1$, and for every $2 \le i \le d_1+1$, u_1 has at least $\frac{\epsilon}{2h}n$ neighbors in V_i, implying that $|V_i| \ge \frac{\epsilon}{2h}n$. As $|V_i| \le n$, it follows that, *each* vertex in V_1 has at least $\frac{\epsilon}{2h}|V_i|$ neighbours in *each* V_i. We can continue this way to conclude that for $1 \le i \le k$, $|V_i| \ge \frac{\epsilon}{2h}n$, and that *every* $u_i \in V_i$ has at least $\frac{\epsilon}{2h}|V_j|$ neighbors in V_j, for *every* $j \in N(i)$. Finally note that as G' is a subgraph of G, all of the above applies also to G.

The previous paragraph implies, that we can apply Lemma 5.2 on the sets $V_1,\ldots,V_{d_1+1}$, with $\delta = \frac{1}{4h}$, $\alpha = \frac{\epsilon}{2h}$, $h = h_1$ and ϵ being $\epsilon/(2h)$, to conclude that sampling some $c_1(h)/(\epsilon^{d_1+1})$ vertices of G, finds, with probability at least $1 - \frac{1}{4h}$, an h_1-tuple s_1, of distinct vertices from V_1, such that for $2 \le j \le d_1+1$ they have at least $c'_1(h)\epsilon^{h_1 d_1}|V_j| \ge c"_1(h)\epsilon^{h_1 d_1+1}n$ common neighbors in V_j. The actual constants $c_1(h), c'_1(h), c''_1(h)$ as well as the constants that will appear at the rest of this proof can be derived from the statement of Lemma 5.2 and are omitted in order to keep the presentation simple. For $2 \le j \le d_1+1$, denote by V'_j this set of common neighbors of the vertices of s_1. Now each V'_j is of size at least $c"_1(h)\epsilon^{h_1 d_1+1}n$. By construction of G', every vertex in V_j, has at least $\frac{\epsilon}{2h}|V_t|$ neighbours in V_t, for every $t \in N(j)$. As $V'_j \subseteq V_j$, the same also applies to the vertices of V'_j. For $2 \le j \le d_1+1$, we can now apply Lemma 5.2 to V'_j as follows. Take $\delta = \frac{1}{4h}$, $\alpha = |V'_j|/n \ge c"_1(h)\epsilon^{h_1 d_1+1}$, $h = h_j$, $d = d_j$ and ϵ as before. We conclude that sampling $c_2(h)/(\epsilon^{d_j+d_1 h_1+1})$ finds, with probability at least $1 - \frac{1}{4h}$, an h_j-tuple s_j of distinct vertices from V'_j, with the property, that all the vertices of s_1 are adjacent to all the vertices of s_j, and the vertices of s_j have at least $c'_2(h)\epsilon^{d_j h_j}|V_t|$ common neighbors in V_t, for every $t \in N(j)$.

We now turn to generalizing the above for all $1 \le i \le k$, but before doing so we must take care of the following minor technicality; we must make sure that we do not sample the same vertex twice when we look for the copy of H, as it must consist of distinct vertices. We therefore remove from each V'_j the previously used vertices. As H is of fixed size, each V'_j is still of essentially its previous size.

Observe, that as each vertex in V_i has at least $\frac{\epsilon}{2h}|V_t|$ neighbours in V_t, for every required t, and we made sure that we do not sample the same vertex twice, we can safely generalize the above sampling technique as follows. For every $2 \le i \le k$, let p_i be the (single) neighbor of i in K that precedes it in the BFS order. Therefore, for every $2 \le i \le k$ we can sample some $c_3(h)/(\epsilon^{d_i+d_{p_i}h_{p_i}+1})$ vertices, to find, with probability at least $1 - \frac{1}{4h}$, an h_i-tuple s_i, with the properties, that every member in s_{p_i} is adjacent to every member of s_i, and the vertices of s_i have at least $c'_3(h)\epsilon^{d_i h_i}|V_t|$ common neighbors in V_t for every $t \in N(i)$. Observe, that as $k \leq h$, the probability that at least one of these k samples failed is at most $k/4h \leq 1/4$. Therefore, with probability at least $3/4$ we have found k sets $s_1, \ldots, s_k$ of sizes $h_1, \ldots, h_k$, respectively, such that for every edge (i, j) in K, we have all the edges going from s_i to s_j. This digraph clearly contains a copy of H, as needed. As for the total number of vertices sampled, note that we do not sample more than h times the size of the largest sample we use. The first sample, the one used to find s_1 is of size $c_1(h)/(\epsilon^{d_1+1}) = O((1/\epsilon)^{d_1+1})$. For $2 \leq i \leq k$, we use a sample of size $O((1/\epsilon)^{d_i+d_{p_i}h_{p_i}+1})$. If we define $\overline{h} = \max_{2 \leq i \leq k}\{d_i + d_{p_i}h_{p_i} + 1\}$, then the total sample size is $O((1/\epsilon)^{\overline{h}})$. As it is clear that for every tree of size h, $\overline{h} \leq h^2$, we conclude that our ϵ-tester has indeed a query complexity of $O((1/\epsilon)^{h^2})$. ■

It is worth observing that in the proof of Theorem 2 part (ii), we did not explicitly use the fact that the core of the considered digraph H is a tree. Rather, we only needed the fact that $V(H)$ can be homomorphically mapped to some subgraph which is a tree. However, one can easily see that if such a homomorphism exists, then the core of H must also be a tree. We now turn to prove Theorem 3, that states that in case H is an oriented tree, we can design an optimal one-sided error ϵ-tester that simply samples a subset of $O(1/\epsilon)$ vertices, and checks if they span a copy of H.

Proof of Theorem 3 If G is H-free, the algorithm clearly reports it. Let G be ϵ-far from being H-free. Consider a *DFS* ordering of the vertices of H, and number the vertices of H accordingly $1, \ldots, h$. It follows that vertex i has exactly one neighbor from $1, \ldots, i-1$. Apply the process described before the proof of Lemma 5.1 with respect to H itself, that is, obtain $G' = G'(G, H)$. It follows from Lemma 5.1 that G' contains a copy of H. Let $v_1, \ldots, v_h$ be such a copy. By Lemma 5.1, for all $1 \leq i \leq h$, $i \in L(v_i)$. Without loss of generality, assume H contains the edge $(1, 2)$. Therefore G' contains an edge (v_1, v_2), and by Lemma 5.1 $1 \in L(v_1)$ and $2 \in L(v_2)$. From the definition of the process for obtaining G', it follows that there are at least $\frac{\epsilon}{2h}n$ vertices u_1, for which there is an edge (u_1, v_2) and $1 \in L(u_1)$. It follows again from the definition of the process, that for each such u_1, there are at least $\frac{\epsilon}{2h}n$ vertices u_2 for which there is an edge (u_1, u_2) and $2 \in L(u_1)$. We can continue this way inductively to conclude that for every homomorphism mapping the subgraph of H spanned by the vertices $1, \ldots, i$ into G', there are at least $\frac{\epsilon}{2h}n$ possibilities for extending this homomorphism, to a homomorphism from the subgraph of H spanned by $1, \ldots, i+1$ into G'. As H is of fixed size, and n is assumed to be large enough, it follows that for each *injective* homomorphism mapping the subgraph of H spanned by the vertices $1, \ldots, i$ into G', there are at least $\frac{\epsilon}{2h}n - i \geq \frac{\epsilon}{3h}n$ possibilities for extending this injective homomorphism, to an injective homomorphism from the subgraph spanned by $1, \ldots, i+1$ into G'. Finally, observe that as G' is a subgraph of G, all the above applies also to G.

We now turn to the actual proof. We show that a random subset of $9h^2/\epsilon$ vertices, contains a copy of H with probability at least $2/3$. We choose this set one vertex at a time (with repetitions). From the above discussion, it follows that each randomly chosen vertex v, has probability at least $\epsilon/3h$ of having the property that there is a copy of H in G in which v plays the role of vertex 1. More generally, it follows from the above discussion, that for every $1 \leq i \leq h-1$, if we have found vertices $v_1, \ldots, v_{i-1}$ having the property that there is a copy of H in G in which $v_1, \ldots, v_{i-1}$ play the role of vertices $1, \ldots, i-1$, then there are at least $\frac{\epsilon}{3h}n$ vertices u in G, such that there is a copy of H in G, in which $v_1, \ldots, v_{i-1}$ play the role of $1, \ldots, i-1$ respectively, and u plays the role of v_i. Therefore, each randomly chosen vertex has probability at least $\epsilon/3h$ of decreasing the number of vertices that are required in order to complete a copy of H, *regardless* of any history. By linearity of expectation, and the fact that the expected number of trials needed to find each new vertex is geometrically distributed, it follows that the expected number of trials needed to find a copy of H is $3h^2/\epsilon$. By Markov's inequality, it follows that the probability of not finding a copy of H after $9h^2/\epsilon$ trials, is at most $1/3$, as needed. Note, that the failure probability is in fact exponentially small in h/ϵ, but we do not need this stronger estimate here.

To show that the result is optimal, we show how to construct, for every tree H, a digraph G_H, that is ϵ-far from being H-free, yet in order to find a copy of H, one must sample $\Omega(1/\epsilon)$ vertices of G_H. Given a tree H of size h, construct a digraph G_H as follows: Let K be the core of H (which is obviously a tree), and let k denote its size. We also denote by t the number of vertices that are mapped to vertex k of K in a homomorphism from H to K. The digraph G_H contains $k-1$ sets of vertices $V_1, \ldots, V_{k-1}$ of size $\frac{n-\epsilon 2kn}{k-1}$ each, and one subset V_k of size $\epsilon 2kn$. For each edge (i,j) in K, G_H contains an edge (v_i, v_j) for every $v_i \in V_i$ and $v_j \in V_j$. To show that G_H is ϵ-far from being H-free, observe that there are

$$(\epsilon 2kn)^t \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t}$$

natural homomorphisms from H into G_H, and at least half of them are injective (there are $o(n^h)$ homomorphisms that are not injective), that is, at least half of them define a copy of H. On the other hand, each edge e in G_H, is in the image of at most

$$(\epsilon 2kn)^{t-1} \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t-1}$$

of these homomorphisms from H to K. Therefore, for a large enough n, one must remove at least

$$\frac{1}{2}(\epsilon 2kn)^t \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t} \cdot (\epsilon 2kn)^{1-t} \left(\frac{n-\epsilon 2kn}{k-1}\right)^{1-h+t} \geq \epsilon kn \frac{n-\epsilon 2kn}{k-1} \geq \epsilon n^2$$

edges, in order to make G_H H-free, and hence G_H is ϵ-far from being H-free. The first multiplicand comes from the number of copies of H in G_H which is at least half of the number of homomorphisms from H to G_H, while the second comes from the number of copies of H which share a given edge. In order to establish that a digraph is not H-free, a one-sided error property-tester must find a copy of H. Now, by the minimality of K, each copy of H in G_H must have a vertex from V_k. Therefore, in order to find a copy of H with probability $2/3$, one must find a vertex in V_k with at least this probability. As proved by Goldreich and Trevisan in [25], we may assume without loss of generality that any one sided error property tester for P_H samples uniformly at random a subset of vertices, and answers by only inspecting edges spanned by this set. Finally, to find a vertex from V_k with probability at least $2/3$, one must sample uniformly at random at least $\Omega(1/\epsilon)$ vertices. Thus, we obtain a lower bound of $\Omega(1/\epsilon)$ as required. ■

6 Hard to Test Digraphs

In this section we apply the approach used in [1], together with some additional ideas, in order to prove Theorem 2 part (iii). This approach uses techniques from additive number theory, based on the construction of Behrend [14] of dense sets of integers with no three-term arithmetic progressions, together with some properties of homomorphisms of digraphs.

A linear equation with integer coefficients

$$\sum a_i x_i = 0 \tag{5}$$

in the unknowns x_i is *homogeneous* if $\sum a_i = 0$. If $X \subseteq M = \{1, 2, \ldots, m\}$, we say that X *has no non-trivial solution to* (5), if whenever $x_i \in X$ and $\sum a_i x_i = 0$, it follows that all x_i are equal. Thus, for example, X has no nontrivial solution to the equation $x_1 - 2x_2 + x_3 = 0$ if and only if it contains no three-term arithmetic progression. The following lemma is proved in [1] (Lemma 3.1), following the method of [14]:

Lemma 6.1 *For every fixed integer $r \geq 2$ and every positive integer m, there exists a subset $X \subset M = \{1, 2, \ldots, m\}$ of size at least*

$$|X| \geq \frac{m}{e^{10\sqrt{\log m \log r}}}$$

with no non-trivial solution to the equation

$$x_1 + x_2 + \ldots + x_r = r x_{r+1}. \tag{6}$$

Let $C = (v_1, \ldots, v_{r+1}, v_1)$ be an arbitrary *oriented* cycle of length $r+1$. We next apply the construction in the above lemma to construct, for every integer $r+1 \geq 3$, a relatively dense digraph consisting of pairwise edge disjoint copies of C, which does not contain too many copies of C of a special structure (see statement of lemma below). Let m be an integer, let $X \subset \{1, 2, \ldots m\}$ be a set satisfying the assertion of Lemma 6.1, and define, for each $1 \leq i \leq r+1$, the set V_i to consist of the vertices $\{1, 2, \ldots im\}$ where, with a slight abuse of notation, we think on the sets $V_1, \ldots, V_{r+1}$ as being pairwise disjoint. The reason we use this notation is that we will next refer to the vertices of these sets as integers. In order to avoid confusion, when we will later on refer to a vertex we will always state to which of the sets $V_1, \ldots, V_{r+1}$ it belongs.

Let $T = T(X, C)$ be the family of all $r+1$-partite digraphs on the classes of vertices $V_1, V_2, \ldots, V_{r+1}$, whose edges are defined as follows: For each j, $1 \leq j \leq m$, and for each $x \in X$ the vertices $j \in V_1, j + x \in V_2, j + 2x \in V_3, \ldots, j + rx \in V_{r+1}$ form an oriented cycle of length $r + 1$ in this order, whose edges are directed as the edges of C. Therefore, if C contains the directed edge (v_i, v_{i+1}), then $(j + (i-1)x, j + ix)$ is an edge from V_i to V_{i+1} for all $1 \leq j \leq m, x \in X$, in any member of T. If C contains the reverse edge (v_{i+1}, v_i), then $(j + ix, j + (i-1)x)$ is an edge from V_{i+1} to V_i for all $1 \leq j \leq m, x \in X$ in any member of T. The same applies to the edges between V_1 and V_{r+1}. If (v_i, v_{i+1}) is an edges in C, then any digraph in T does not contain any additional edges going from V_i to V_{i+1}. If (v_{i+1}, v_i) is an edge in C, then any digraph in T does not contain any additional edges going from V_{i+1} to V_i. The same applies to V_1, V_{r+1}. Besides the above set of edges and restrictions, the members of T may contain any other edges between V_i, V_j.

Lemma 6.2 *For every integer $r \geq 2$, and every m, any member of $T(X, C)$ defined above has precisely $m|X|$ ($< m^2$) copies of the cycle C, such that the vertex that plays the role of v_i in the copy of C, belongs to V_i.*

Proof: We only have to show that any member of T does not contain any additional copies of C, for which the vertex that plays the role of v_i in the copy of C, belongs to V_i. Let C' be such a copy of C. Therefore, there are $j \leq m$ and elements $x_1, x_2, \ldots, x_{r+1} \in X$, such that the vertices of the cycle are $j \in V_1, j + x_1 \in V_2, j + x_1 + x_2 \in V_3, \ldots, j + x_1 + x_2 + \ldots + x_r \in V_{r+1}$ and $x_1 + x_2 + \ldots + x_r = rx_{r+1}$ (remember that all edges between V_1 and V_{r+1} are of the form $(j, j + rx)$ or $(j + rx, j)$). However, by the definition of X this implies that $x_1 = x_2 = \ldots = x_{r+1}$, implying the desired result. ■

Comment: Note that the members of $T(X, C)$ may contain many additional copies of C, which do not satisfy the restriction described in the statement of the lemma.

An s-blow-up of a digraph $K = (V(K), E(K))$ is the digraph obtained from K by replacing each vertex of K by an independent set of size s, and each edge e of K by a complete bipartite directed subgraph whose vertex classes are the independent sets corresponding to the ends of the edge, and whose edges are directed according to the direction of e.

Lemma 6.3 *Let $H = (V(H), E(H))$ be a digraph with h vertices, let $K = (V(K), E(K))$ be another digraph on at most h vertices, and let $T = (V(T), E(T))$ be an s-blow-up of K. Suppose there is a homomorphism*

$$\varphi : V(H) \mapsto V(K)$$

from H to K and suppose $s \geq h$. Let $R \subset E(T)$ be a subset of the set of edges of T, and suppose that each copy of H in T contains at least one edge of R. Then

$$|R| \geq \frac{|E(T)|}{|E(K)||E(H)|} > \frac{|E(T)|}{h^4}.$$

Proof: Let $g : V(H) \mapsto V(T)$ be a random injective mapping obtained by defining, for each vertex $v \in V(K)$, the images of the vertices in $\varphi^{-1}(v) \in V(H)$ randomly, in a one-to-one fashion, among all s vertices of T in the independent set that corresponds to the vertex v. Obviously, g maps adjacent vertices of H into adjacent vertices of T, and hence the image of g contains a copy of H in T. Each edge of H is mapped to one of the corresponding s^2 edges of T according to a uniform distribution, and hence the probability it is mapped onto a member of R does not exceed $|R|/s^2$. It follows that the expected number of edges of H mapped to members of R is at most $\frac{|R||E(H)|}{s^2}$, and as, by assumption, this random variable is always at least 1, we conclude that $\frac{|R||E(H)|}{s^2} \geq 1$. The desired result follows, since $s^2 = |E(T)|/|E(K)|$. ■

Claim 6.1 *If K, the core of H, is neither a tree nor a 2-cycle, then K contains an oriented cycle C of length at least 3. Moreover, any homomorphism from H to K, maps a copy of C from H to the copy of C in K.*

Proof: Let k denote the number of vertices of K, and let us number its vertices $\{v_1, v_2, \ldots, v_k\}$ such that the first $r+1 \geq 3$ vertices $v_1, v_2, \ldots v_{r+1}$ form an oriented cycle C in this order. One such cycle must exist as K is by assumption neither a tree nor a 2-cycle. Remember, that as was explained in the discussion before the proof of Theorem 2, part (i), the core cannot have only 2-cycles, and not be a 2-cycle. By the minimality of K, every homomorphism φ of K into itself must be an automorphism, that is $(u, v) \in E(K) \Leftrightarrow (\varphi(u), \varphi(v)) \in E(K)$ (otherwise H would have a homomorphism into a subgraph with a smaller number of edges). We claim that *any* homomorphism of H into K maps a copy of C from H to the vertices $v_1, v_2, \ldots v_{r+1}$ of K. Indeed, any homomorphism of H into K, induces also a homomorphism of K into K. Therefore, some $r+1$ vertices of K are mapped to $v_1, v_2, \ldots, v_{r+1}$, and these vertices must span a cycle in K and therefore in H, as this homomorphism is an automorphism from K to K by the previous argument. ■

Lemma 6.4 *For every fixed digraph $H = (V(H), E(H))$ on h vertices whose core is neither an oriented tree nor a 2-cycle, there is a constant $c = c(H) > 0$, such that for every positive $\epsilon < \epsilon_0(H)$ and every integer $n > n_0(\epsilon)$, there is a digraph G on n vertices which is ϵ-far from being H-free, and yet contains at most $\epsilon^{c \log(1/\epsilon)} n^h$ copies of H.*

Proof: Let K be the core of H, and let k denote the number of vertices of K. Also, let us number its vertices $\{v_1, v_2, \ldots, v_k\}$ such that the first $r+1 \geq 3$ vertices $v_1, v_2, \ldots v_{r+1}$ form an oriented cycle C

in this order as guaranteed by Claim 6.1. Given a small $\epsilon > 0$, let m be the largest integer satisfying

$$\epsilon \le \frac{1}{h^8 e^{10\sqrt{\log m \log h}}}. \tag{7}$$

It is easy to check that this m satisfies

$$m \ge \left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)} \tag{8}$$

for an appropriate $c = c(h) > 0$. Let $X \subset \{1, 2, \ldots, m\}$ be as in Lemma 6.1. We next define a digraph F from K in a way similar to the one described in the paragraph preceding Lemma 6.2. Let $V_1, V_2, \ldots V_k$ be pairwise disjoint sets of vertices, where $|V_i| = im$ and we denote the vertices of V_i by $\{1, 2, \ldots, im\}$. For each j, $1 \le j \le m$, for each $x \in X$ and for each directed edge (v_p, v_q) of K, let $j + (p-1)x \in V_p$ have an outgoing edge pointed to $j + (q-1)x \in V_q$. In other words, F consists of $m|X|$ copies of K, where the vertices of each copy form an arithmetic progression whose first element is j and whose difference is x. It follows that each pair of these copies shares at most one vertex in F. In particular, these copies are edge disjoint. It thus follows that the number of edges in F satisfies

$$|E(F)| = m|X||E(K)|.$$

Note that the induced subgraph of F on the union of the first $(r+1)$ vertex classes, belongs to the family of digraphs $T(X, C)$ considered in Lemma 6.2, where $C = (v_1, \ldots, v_{r+1}, v_1)$ is the oriented cycle on the first $r+1$ vertices of K, which was defined above. Finally, define

$$s = \left\lfloor \frac{n}{|V(F)|} \right\rfloor = \left\lfloor \frac{2n}{k(k+1)m} \right\rfloor$$

and let G be the s-blow-up of F (together with some isolated vertices, if needed, to make sure that the number of vertices is precisely n). Note that the number of edges of G satisfies,

$$|E(G)| = \frac{4n^2|E(F)|}{k^2(k+1)^2m^2} = \frac{4n^2|X||E(K)|}{k^2(k+1)^2m} \ge \frac{n^2|X||E(K)|}{k^4m} \ge \frac{n^2|E(K)|}{k^4e^{10\sqrt{\log m \log r}}} \tag{9}$$

where the last inequality follows from the lower bound on $|X|$ that is guaranteed by Lemma 6.1.

Since F consists of $m|X|$ edge disjoint copies of K, G consists of pairwise edge disjoint s-blow-ups of K, hence, by Lemma 6.3, one has to delete at least a fraction of $1/h^4$ of its edges to destroy all copies of H in it. Therefore, one must delete at least

$$\frac{1}{h^4} \cdot |E(G)| \ge \frac{n^2|E(K)|}{h^4k^4e^{10\sqrt{\log m \log r}}} \ge \frac{n^2|E(K)|}{h^8e^{10\sqrt{\log m \log h}}} \ge \epsilon n^2 \tag{10}$$

edges in order to destroy all copies of H. The first inequality follows from (9), the second from the fact that $r \le h$ and $k \le h$ and the third from (7). We conclude that G is ϵ-far from being H-free.

We next claim that any copy of H in G must contain a copy of C such that for $1 \leq i \leq r+1$, the vertex that plays the role of v_i belongs to the blow-up of the vertices of V_i. To see this, note that there is a natural homomorphism of G onto K, obtained by first mapping G homomorphically onto F (by mapping each class of s vertices into the vertex of F to which it corresponds), and then by mapping all vertices of V_i to v_i. This homomorphism maps each copy of H in G homomorphically into K, and hence, by Claim 6.1, maps a copy of C that belongs to the considered digraph H, to the first $r+1$ vertices of K. The definition of the homomorphism thus implies the assertion of the claim.

As the vertex that plays the role of v_i in the copy of C must belong to the blow-up of the vertices of V_i for $1 \leq i \leq r+1$, it follows from Lemma 6.2 that the number of such cycles is at most

$$m^2 s^{r+1} = m^2 \left(\frac{2n}{k(k+1)m}\right)^{r+1} \leq n^{r+1}/m,$$

and this implies that the total number of copies of H in G does not exceed $n^h/m = \epsilon^{c\log(1/\epsilon)} n^h$, implying the desired result. ■

Proof of Theorem 2, part (iii): Let H be a digraph on h vertices whose core is neither an oriented tree nor a 2-cycle, and suppose $\epsilon > 0$. Given a one-sided error ϵ-tester for testing H-freeness we may assume, without loss of generality, that it queries all pairs of a uniformly at random chosen set of vertices (otherwise, as explained in [5], every time the algorithm queries about a vertex pair we make it query also about all pairs containing a vertex of the new pair and a vertex from previous queries. See also [25] for a more detailed proof of this statement.) As the algorithm is a one-sided-error algorithm, it can report that G is not H-free only if it finds a copy of H in it. By Lemma 6.4 there is a digraph G on n vertices which is ϵ-far from being H-free and yet contains at most $\epsilon^{c\log(1/\epsilon)} n^h$ copies of H. The expected number of copies of H inside a uniformly at random chosen set of x vertices in such a digraph is at most $x^h \epsilon^{c\log(1/\epsilon)}$, which is far smaller than 1 unless x exceeds $(1/\epsilon)^{c'\log(1/\epsilon)}$ for some $c' = c'(H) > 0$, implying the desired result. ■

7 Two-Sided Error ϵ-Testers

In this section we present the proof of Theorem 4. Applying the second part of the theorem for the case of undirected graphs, shows that if H is an undirected, non-bipartite graph, then there is no two-sided ϵ-tester for testing H-freeness whose query complexity is smaller than $(1/\epsilon)^{c\log 1/\epsilon}$ for an appropriate $c = c(H) > 0$. This settles an open problem raised in [1]. For the proof we need the following easy application of a theorem of Erdős from [20].

Lemma 7.1 *Let H be a fixed digraph on h vertices, let K be its core, and denote by k the size of K. For every constant $0 < \gamma < 1$ and for every sufficiently large n, every digraph G on n vertices that contains γn^k copies of K, contains also a copy of H.*

Proof: Let φ be a homomorphism from $V(H)$ to $V(K)$, denote by $t_1, \ldots, t_k$ the vertices of K, and let $S_1, \ldots, S_k$ be the sets $\varphi^{-1}(t_1), \ldots, \varphi^{-1}(t_k)$, respectively. Define a k-uniform hypergraph T as follows: take a random partition of $V(G)$ into k subsets, $V_1, \ldots, V_k$, where each vertex of G is chosen uniformly and independently to be in one of the groups. For each copy of K in G, in which the vertices $u_{i_1}, \ldots, u_{i_k}$ play the role of $t_1, \ldots, t_k$ respectively, put an edge in T that contains $u_{i_1}, \ldots, u_{i_k}$ if and only if $u_{i_1} \in V_1, \ldots, u_{i_k} \in V_k$. Observe, that by linearity of expectation, if G contains γn^k copies of K, the expected number of edges in T is $\gamma k^{-k} n^k$. Therefore, one partition which defines at least this many edges must exist. Fix one such partition, and the hypergraph T' which it defines. In [20] it is proved that any k-uniform hypergraph on n vertices with at least $n^{k-h^{1-k}}$ edges, contains a copy of a complete k-partite k-uniform hypergraph, where each partition class is of size h. It follows that for large enough n, T' contains a copy of such hypergraph on some hk vertices $\{v_1^1, \ldots, v_h^1\} \subseteq V_1, \ldots, \{v_1^k, \ldots, v_h^k\} \subseteq V_k$. It is now easy to see that G must contain a copy of H where for the role of the vertices of S_i we can choose any $|S_i|$ vertices from $\{v_1^i, \ldots, v_h^i\}$. ■

Proof of Theorem 4, part (i): Let H be a fixed digraph with core K, and let k be the size of K. If K is a 2-cycle, then a two-sided error ϵ-tester for testing P_H with query complexity $O(1/\epsilon)$ was described in the comment following the proof of Theorem 2 part (i). Assume now that K is an oriented tree. Our two-sided error ϵ-tester for P_H works as follows: Given a digraph G, the algorithm samples c/ϵ vertices, for an appropriate c, and reports that the digraph is not H-free if and only if they span a copy of K. We turn to show that the algorithm answers correctly with probability at least $2/3$. Assume G is ϵ-far from being H-free. Then it is clearly also ϵ-far from being K-free, therefore applying Theorem 3 to P_K, we conclude that a randomly chosen set of c/ϵ vertices, with an appropriate c, finds a copy of K with probability at least $2/3$. Assume G does not contain a copy of H. It follows from Lemma 7.1 that it contains $o(n^k)$ copies of K, and therefore a randomly chosen set of any constant size (independent of n), and in particular of size $O(1/\epsilon)$, has probability $o(1)$ of finding a copy of K.

To show that the result is optimal, we apply Yao's principle [34]. We first prove the case of K being an oriented tree. Applying Yao's principle to our setting, we first have to define for every n, two distributions of digraphs D_1, D_2, where all the digraphs in D_1 are ϵ-far from being H-free, and all the digraphs in D_2 are H-free. In order to define the two distributions we use the digraph G_H whose description appears at the end of the proof of Theorem 3. Note that this digraph is constructed using the core K, which is a tree. D_1 is a uniform distribution on all the $n!$ digraphs that are obtained from G_H by a permutation of its vertices. By the computation at the end of the proof of Theorem 3 it follows that all the digraphs in D_1 are ϵ-far from being H-free. To define D_2 we first define G'_H to be the digraph that is obtained from G_H by removing all the edges that touch V_k (see the definition of G_H). D_2 is now a uniform distribution on all the $n!$ digraphs that are obtained from G'_H by a permutation of its vertices. As G'_H is clearly H-free, all the digraphs in D_2

are H-free. To finish the proof we must show that no deterministic algorithm that samples less than $\Omega(1/\epsilon)$ vertices (adaptively) can tell the difference between these two distributions with probability that exceeds, say, 1/3. Recall that by the definition of G_H and G'_H, as long as the algorithm does not look at a vertex from V_k, it sees the *same* digraph. As V_k is of size $\epsilon 2kn$, the probability that a deterministic algorithm that samples less than, say, $1/(10\epsilon k)$ vertices finds a vertex from V_k is smaller than 1/3. Therefore, with probability at least 2/3 the two distributions D_1, D_2 will look identical to any deterministic algorithm sampling less than $\Omega(1/\epsilon)$ vertices, as needed.

The proof for the case of K being a 2-cycle is analogous, and involves taking a permutation of a complete bi-directed bipartite graph on vertex sets of sizes $\epsilon 4n$ and $n - \epsilon 4n$, and a digraph with no edges. The rest of the details are left to the reader. ■

A close inspection at the proofs of Theorem 3 and Theorem 2 part (i), shows that if G is ϵ-far from being H-free, and the core of H, K, is either a 2-cycle or an oriented tree, then sampling $O(1/\epsilon)$ vertices finds a copy of K with probability $1 - o(1)$ where the $o(1)$ term tends to 0 as ϵ tends to zero. On the other, the proof of Theorem 4, part (i), shows that if G is H-free, then the algorithm does not find a copy of K with probability $1 - o(1)$ where the $o(1)$ term tends to 0 as n tends to infinity (even if $\epsilon > 0$ is relatively large). Therefore, in some sense the test has "almost" one-sided error, as even for large values of ϵ the failure probability in case G is H-free is still $o(1)$, as n tends to infinity.

Proof of Theorem 4, part (ii): Let H be a fixed digraph whose core K is neither a directed 2-cycle nor an oriented tree. We apply Yao's principle again in order to prove the lower bound.

Given n and ϵ, let X, m and the sets V_i be as in the proof of Lemma 6.4. Construct the digraph F just as in the proof of Lemma 6.4, and remember that it consists of $m|X|$ pairwise edge disjoint copies of K (though it may well contain additional copies of K). Recall, also, that K contains a cycle C of length $r + 1 \geq 3$, and that each copy of K in F contains a copy of this cycle in which the i-th vertex lies in V_i for all $1 \leq i \leq r + 1$. Let $\mathcal{C}$ denote the set of these edge disjoint copies of C, and note that by Lemma 6.2 there are no other copies of C in F, in which the i-th vertex lies in V_i, besides the $m|X|$ members of $\mathcal{C}$.

To construct D_1 which consists of digraphs that are ϵ-far from being H-free with probability $1 - o(1)$, we first construct F'_1 by removing each of the $m|X|$ edge disjoint cycles that belong to $\mathcal{C}$ with probability $\frac{1}{r+1}$. We then create G_1 by taking an s blow up of F'_1 adding isolated vertices, if needed. Finally, D_1 consists of all randomly permuted copies of such digraphs G_1. It follows from a standard Chernoff bound, that with probability $1 - o(1)$, at least $m|X|(1 - 2/(r + 1))$ copies of C are left in F'_1, where the $o(1)$ term tends to 0, as ϵ tends to 0. Similar to the derivation of (10), it is easy to show that if $m|X|/2(r + 1)$ of these copies of C are left in F'_1, the digraph G_1 is ϵ-far from being H-free. It follows that with probability $1 - o(1)$, a member of D_1 is ϵ-far from being H-free. The distribution D_2 of digraphs that are H-free, is defined by first constructing F'_2 by removing from each member $C \in \mathcal{C}$ one randomly chosen edge (out of the $r + 1$ edges of the cycle). We then create G_2 by taking an s blow up of F_2' adding isolated vertices, if needed. Finally, D_2 consists of all randomly permuted copies of such digraphs G_2, which are clearly H-free.

Now consider a set of vertices S in G_1 (or G_2) and its natural projection to a subset of $V(F)$, which we also denote by S with a slight abuse of notation. Suppose S has the property that it does not contain more than two vertices from any one of the copies of C that belong to $\mathcal{C}$.

If this property holds, then each edge spanned by S is contained in a different copy of $C \in \mathcal{C}$ (if it is contained in such a cycle at all). Therefore, each edge that lies in such a cycle, has probability $1 - \frac{1}{r+1}$ of being in F_1', and these probabilities are mutually independent. Similarly, each such edge has probability $1 - \frac{1}{r+1}$ of being in F_2' and these probabilities are also mutually independent. It follows that sampling a digraph G from D_1, and looking at the induced digraph on a set S with the above property, has *exactly* the same distribution as sampling a digraph G from D_2, and looking at the induced digraph on S.

To complete the proof we have to show that no deterministic algorithm can distinguish between the distributions D_1 and D_2 with constant probability. To this end, it is clearly enough to show that any deterministic algorithm that looks at a digraph spanned by less than $(1/\epsilon)^{c' \log 1/\epsilon}$ vertices, has essentially the same probability of seeing any digraph regardless of the distribution from which the digraph was chosen. By the discussion in the previous paragraph, this can be proved by establishing that, with high probability, a small set of vertices does not contain three vertices from the same copy of C. For a fixed ordered set of three vertices in S, consider the event that they all belong to the same copy of C. The first two vertices determine all the vertices of one of these copies uniquely. Now, the conditional probability that the third vertex is also a vertex of the same copy is $(r+1)/|V(F)| \leq r/m$. By the union bound, the probability that the required property is violated is at most

$$r|S|^3/m \leq r|S|^3 \epsilon^{c \log 1/\epsilon}.$$

This quantity is $o(1)$ as long as $|S| = o((1/\epsilon)^{\frac{c}{3} \log 1/\epsilon})$, where here we applied the lower bound on the size of m given in (8). Therefore, if the algorithm has query complexity $o((1/\epsilon)^{c' \log 1/\epsilon})$ for some absolute positive constant c', it has probability $1 - o(1)$ of looking at a subset on which the distributions D_1 and D_2 are identical, thus, the probability that it distinguishes between D_1 and D_2 is $o(1)$. ■

A slightly more complicated argument than the above can give two distributions D_1 and D_2, such that the digraphs in D_1 are *always* ϵ-far from being H-free, while the digraphs in D_2 are always H-free. The idea is to first partition the $m|X|$ copies of C into pairs, assuming for simplicity that $m|X|$ is even. To create D_1, we randomly pick from each pair of copies of C a single copy, and delete two randomly chosen edges from this copy. To create D_2, we do exactly the same as we did in the proof above. It is easy to appropriately modify the proof above in order to show that any deterministic algorithm with query complexity $o((1/\epsilon)^{c \log 1/\epsilon})$ cannot distinguish between D_1 and D_2

(see [10] for more details). As this argument has no qualitative advantage, we described the simpler one given above.

Observe that for digraphs H whose core K is neither an oriented tree nor a 2-cycle, we can give the above lower bound for testing P_H, but no better upper bound than the one given by Theorem 1. However, following the arguments in the proof of Theorem 4 (i), it follows that the query complexity of testing P_H with two-sided error is at most the query complexity of testing P_K with two-sided error. Thus, for example, the query complexity of testing the digraph in Figure 1 (c) with two-sided error, is at most the query complexity of testing its induced oriented triangle with two-sided error.

8 Concluding Remarks and Open Problems

- We have shown that for any digraph H, the property P_H of being H-free is strongly testable. In order to prove this result we have first proved a regularity lemma for digraphs, which generalizes Szemerédi's regularity lemma for undirected graphs. This lemma might prove useful for tackling other problems as well. We also gave a precise characterization of all digraphs H for which P_H is easily testable, and showed that the same characterization applies to two-sided error ϵ-testers as well, where here the complexity is polynomial in $1/\epsilon$ if and only if it is $\Theta(1/\epsilon)$. We have addressed the case when H is an oriented tree, and gave an optimal one-sided error ϵ-tester with query complexity $\Theta(1/\epsilon)$ for this case.

- It is not difficult to generalize Theorem 2 to the case of disconnected digraphs. Let H be a disconnected graph whose components we denote by $H_1, \ldots, H_t$, and whose cores we denote by $K_1, \ldots, K_t$. Note that if G is ϵ-far from being H-free, then for all i, it is also ϵ-far from being H_i-free. If $K_1, \ldots, K_t$ are all either trees or 2-cycles, then running the testers for $H_1, \ldots, H_t$ will find disconnected copies of each of $H_1, \ldots, H_t$, and therefore a copy of H. This test will obviously have query complexity polynomial in $1/\epsilon$, and therefore in this case P_H is easily testable. If at least one of the cores is neither a tree nor a 2-cycle then the core of H is neither a tree nor a 2-cycle, hence, it follows directly from the proof of Theorem 2 part (iii) (note that Lemma 6.4 and the proof of Theorem 2 part (iii) do not assume that H is connected) that P_H is not easily testable. Note finally that the above applies also to the case of two-sided error, thus Theorem 4 can also be extended to the case of disconnected digraphs.

- An intriguing problem is that of estimating the best possible (one-sided and two-sided) query complexity of the property P_H^* of not containing any **induced** copy of a fixed digraph H. Very recently we have proved that for any fixed H with at least 5 vertices, the query complexity of any one-sided or two-sided ϵ-tester for P_H^* is not polynomial in $1/\epsilon$. The details appear in a subsequent paper [11]. These results have been further extended to hypergraphs is [12].

- Hell, Nesetril and Zhu proved in [28] that the problem of deciding if the core of a given input digraph is a tree is NP-complete. This, together with Theorem 2 imply the following.

  Proposition 8.1 *The problem of deciding whether for a given digraph H, the property P_H is easily testable, is NP-complete.*

  Therefore, there is no polynomially testable characterization of the digraphs H for which P_H is easily testable (though for every small, fixed H, Theorem 2 can be easily used to decide if H is such a digraph). One interesting class of digraphs for which the problem is solvable in polynomial time, is the class of oriented cycles. An oriented cycle is *balanced* if the number of forward edges is equal to the number of backward edges. It is not difficult to see that if an oriented cycle C is not balanced, then the core of C is C itself, (see, e.g., Figure 1 (b)). However the converse is not true, and while there are balanced cycles whose core is a path, (see, e.g., Figure 1 (a)), there are also balanced cycles C whose core is C itself, (see, e.g., Figure 1 (d)). It is therefore interesting to observe that the problem of deciding whether the core of a given cycle C is C itself or an induced path in it, can be solved in polynomial time using dynamic programming. The details are left to the reader.

  A digraph H is balanced iff every oriented cycle in it is balanced. It is not difficult to see that a digraph H is balanced iff there is a homomorphism mapping H into an oriented tree, and this happens iff there is a homomorphism mapping H into a directed path. It thus follows, by Theorem 2, that if H is not balanced then P_H cannot be tested by a polynomial number of queries (but the converse is not true in general.)

- Lemma 5.1 implies that if G is ϵ-far from satisfying P_H, and the core of H is a tree K of size k, then G contains $\Omega(\epsilon^k n^k)$ copies of K. Having this, we could have used results from the theory of supersaturated graphs and hypergraphs (see [21]) to conclude that there exists a one-sided error ϵ-tester for P_H which uses a sample of size $O((1/\epsilon)^{O(h^k)})$. (An alternative way to deduce this, is to change the statement of Lemma 7.1 and prove that G contains $c(\gamma)n^h$ copies of H for some constant $c(\gamma)$, and not just one). However, our proof of Theorem 2 part (ii) given here provides a far more efficient ϵ-tester that uses a sample of size only $O((1/\epsilon)^{h^2})$. By applying the techniques of [21] we can show that for every fixed digraph H with h vertices whose core K (which is not necessarily a tree) has k vertices, any digraph on n vertices containing at least δn^k copies of the core K, contains at least $\Omega(\delta^{O(h^k)} n^h)$ copies of H.

- Lemma 5.1 implies that if G is ϵ-far from satisfying P_H, and H is a tree of size h, then G contains $\Omega(\epsilon^h n^h)$ copies of H. This can be seen to be essentially optimal by considering an appropriate random digraph. We omit the details.

  As there are many copies of H, we conclude that sampling h vertices finds a copy of H with probability $\Omega(\epsilon^h)$. It follows that one can test P_H simply by sampling $\Theta((1/\epsilon)^h)$ samples of

h vertices each. However, in Theorem 3 we show that a sample of size $O(1/\epsilon)$ suffices. The reason is that sampling h vertices in $O((1/\epsilon)^h)$ rounds fails to take into account all the h-tuples that lie in the sample. In a sample of size $\Theta(1/\epsilon)$ there are $\Theta((1/\epsilon)^h)$ subsets of size h, and it turns out that if we consider all of them, we get essentially the same result as sampling $\Theta((1/\epsilon)^h)$ subsets of size h. In general, showing that if G is ϵ-far from being H-free then it contains $f(\epsilon)n^h$ copies of H, and then designing a ϵ-tester that samples $1/f(\epsilon)$ subsets of size h, usually fails to achieve the query complexity of more efficient ϵ-testers. In many cases, the difference can be substantial, as in our case. In addition, our proof of a test that uses a sample of size $O(1/\epsilon)$ gives a somewhat different proof that for any oriented tree H with h vertices, a digraph that is ϵ-far from being H-free, contains $\Omega(\epsilon^h n^h)$ copies of H.

- Testing H-freeness for H being the complete bipartite undirected graph $K_{s,t}$, is another example of the above mentioned phenomenon. In [1], an ϵ-tester for $K_{s,t}$-freeness which uses a sample of size $O((1/\epsilon)^{st})$ has been established, simply by showing that the graph must contain $\Omega(\epsilon^{st} n^{s+t})$ copies of $K_{s,t}$. Our method here improves this result and shows that a sample of size $O((1/\epsilon)^{\min(s,t)})$ suffices. This nearly matches a lower bound of $\Omega((1/\epsilon)^{\min(s,t)/2})$ which follows by considering an appropriate random graph (see the full version of [9].)

Acknowledgements: We would like to thank one of the referees for a thorough and careful reading of this paper, as well as for valuable comments which significantly improved the presentation of our results.

References

[1] N. Alon, Testing subgraphs in large graphs, Proc. 42^{nd} IEEE FOCS, IEEE (2001), 434-441.

[2] N. Alon, R. A. Duke, H. Lefmann, V. Rödl and R. Yuster, The algorithmic aspects of the Regularity Lemma, Proc. 33^{rd} IEEE FOCS, Pittsburgh, IEEE (1992), 473-481. Also: J. of Algorithms 16 (1994), 80-109.

[3] N. Alon, W. F. de la Vega, R. Kannan and M. Karpinski, Random Sampling and Approximation of MAX-CSP Problems, Proc. of the 34^{th} ACM STOC, ACM Press (2002), 232-239.

[4] N. Alon, S. Dar, M. Parnas and D. Ron, Testing of clustering, Proc. 41 IEEE FOCS, IEEE (2000), 240-250.

[5] N. Alon, E. Fischer, M. Krivelevich and M. Szegedy, Efficient testing of large graphs, Proc. 40^{th} Annual Symp. on Foundations of Computer Science (FOCS), New York, NY, IEEE (1999), 656–666. Also: Combinatorica 20 (2000), 451-476.

[6] N. Alon, M. Krivelevich, I. Newman and M. Szegedy, Regular languages are testable with a constant number of queries, Proc. 40^{th} Annual Symp. on Foundations of Computer Science (FOCS), New York, NY, IEEE (1999), 645–655. Also: SIAM J. on Computing 30 (2001), 1842-1862.

[7] N. Alon and M. Krivelevich, Testing k-colorability, SIAM J. Discrete Math., 15 (2002), 211-227.

[8] N. Alon, M. Krivelevich and B. Sudakov, Turán numbers of bipartite graphs and related Ramsey-type questions, Combinatorics, Probability and Computing 12 (2003), 477-494.

[9] N. Alon and A. Shapira, Testing satisfiability, Proc. of the 13^{th} Annual ACM-SIAM SODA, ACM Press (2002), 645-654. Also: Journal of Algorithms, 47 (2003), 87-103.

[10] N. Alon and A. Shapira, Testing subgraphs in directed graphs, Proc. of the 35^{th} Annual Symp. on Theory of Computing (STOC), San Diego, California, 2003, 700–709.

[11] N. Alon and A. Shapira, A characterization of easily testable induced subgraphs, Proc. of the 15^{th} Annual ACM-SIAM SODA, ACM Press (2004), 935-944.

[12] N. Alon and A. Shapira, Linear equations, arithmetic progressions and hypergraph property testing, manuscript, 2004.

[13] J. Bang-Jensen and P. Hell, The effect of two cycles on the complexity of colorings by directed graphs, Discrete Applied Math. 26 (1990), 1-23.

[14] F. A. Behrend, On sets of integers which contain no three terms in arithmetic progression, *Proc. National Academy of Sciences USA* 32 (1946), 331–332.

[15] A. Bogdanov, K. Obata and L. Trevisan, A Lower Bound for Testing 3-Colorability in Bounded-degree Graphs, Proc. 43^{rd} IEEE FOCS, IEEE (2002), 93-102.

[16] B. Bollobás, P. Erdös, M. Simonovits and E. Szemerédi, Extremal graphs without large forbidden subgraphs, *Annals of Discrete Mathematics* 3 (1978), 29–41.

[17] A. Czumaj and C. Sohler, Testing hypergraph coloring, Proc. of ICALP 2001, 493-505.

[18] A. Czumaj and C. Sohler, Property testing in computational geometry, Proceedings of the 8th Annual European Symposium on Algorithms (2000), 155–166.

[19] Reinhard Diestel, **Graph Theory**, Second Edition, Springer-Verlag, New York, 2000.

[20] P. Erdős, P. On extremal problems of graphs and generalized graphs. Israel J. Math. 2 1964 183-190.

[21] P. Erdős and M. Simonovits, Supersaturated graphs and hypergraphs, Combinatorica 3 (1983), 181-192.

[22] E. Fischer, The art of uninformed decisions: A primer to property testing, The Computational Complexity Column of The Bulletin of the European Association for Theoretical Computer Science 75 (2001), 97-126.

[23] A. Frieze and R. Kannan, Quick approximation to matrices and applications, *Combinatorica* 19 (1999), 175-220.

[24] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connection to learning and approximation, *Proceedings of the* 37^{th} *Annual IEEE FOCS* (1996), 339–348. Also: Journal of the ACM 45 (1998), 653–750.

[25] O. Goldreich and L. Trevisan, Three theorems regarding testing graph properties, Random Structures and Algorithms, 23(1):23-57, 2003.

[26] W. T. Gowers, Lower bounds of tower type for Szemerédi's Uniformity Lemma, *Geometric and Functional Analysis* 7 (1997), 322-337.

[27] P. Hell and J. Nesetril, The core of a graph, Discrete Math 109 (1992), 117-126.

[28] P. Hell, J. Nesetril, and X. Zhu, Duality of graph homomorphisms, in : Combinatorics, Paul Erdős is Eighty, (D. Miklós et. al, eds.), Bolyai Society Mathematical Studies, Vol.2, 1996, pp. 271-282.

[29] J. Komlós and M. Simonovits, Szemerédi's regularity lemma and its applications in graph theory, in : Combinatorics, Paul Erdős is Eighty, (D. Miklós et. al, eds.), Bolyai Society Mathematical Studies, Vol.2, 1996, pp. 295-352.

[30] V. Rödl and R. Duke, On graphs with small subgraphs of large chromatic number, *Graphs and Combinatorics* 1 (1985), 91–96.

[31] D. Ron, Property testing, in: P. M. Pardalos, S. Rajasekaran, J. Reif and J. D. P. Rolim, editors, *Handbook of Randomized Computing*, Vol. II, Kluwer Academic Publishers, 2001, 597–649.

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

[33] E. Szemerédi, Regular partitions of graphs, In: *Proc. Colloque Inter. CNRS* (J. C. Bermond, J. C. Fournier, M. Las Vergnas and D. Sotteau, eds.), 1978, 399–401.

[34] A. C. Yao, Probabilistic computation, towards a unified measure of complexity. Proc. of the 18^{th} IEEE FOCS (1977), 222-227.