

# Testing of bipartite graph properties\*

Noga Alon<sup>†</sup>

Eldar Fischer<sup>‡</sup>

Ilan Newman<sup>§</sup>

June 15, 2005

## Abstract

Alon et. al. [3], showed that every property that is characterized by a finite collection of forbidden induced subgraphs is  $\epsilon$ -testable. However, the complexity of the test is *double-tower* with respect to  $1/\epsilon$ , as the only tool known to construct such tests is via a variant of Szemerédi's Regularity Lemma. Here we show that any property of *bipartite* graphs that is characterized by a finite collection of forbidden induced subgraphs is  $\epsilon$ -testable, with a number of queries that is polynomial in  $1/\epsilon$ .

Our main tool is a new 'conditional' version of the regularity lemma for binary matrices, which may be interesting on its own.

---

\*A preliminary (and weaker) version of these results formed part of [9].

<sup>†</sup>Schools of Mathematics and Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel, and IAS, Princeton, NJ 08540, USA. Email: nogaa@tau.ac.il Research supported in part by a grant from the Israel Science Foundation, by the Hermann Minkowski Minerva Center for Geometry at Tel Aviv University, and by the Von Neumann Fund.

<sup>‡</sup>Department of Computer Science, The Technion, Haifa 32000, Israel. Email: eldar@cs.technion.ac.il

<sup>§</sup>Department of Computer Science, University of Haifa, Haifa 31905, Israel. Email: ilan@cs.haifa.ac.il

# 1 Introduction

Testing of graph properties has become an active research area in the recent years (see for example [3, 7, 14, 13, 1, 4] and the surveys [15, 8]). In particular, it was shown in [3] that every property that is characterized by a finite collection of forbidden induced subgraphs is  $\epsilon$ -testable. However, the complexity of the test is *double-tower* with respect to  $1/\epsilon$ , as the only tool known to prove this testability is a variant of Szemerédi’s Regularity Lemma. Recently Alon and Shapira [1, 4] initiated a study of those graph properties that are characterized by forbidden subgraphs and can be tested ‘very efficiently’, in the sense that they can be tested with only  $\text{poly}(1/\epsilon)$  many queries. Here we concentrate on the family of graph properties that are characterized by forbidden induced subgraphs. We show that any property of *bipartite* graphs that is characterized by a finite collection of forbidden induced subgraphs is  $\epsilon$ -testable with a number of queries that is  $\text{poly}(1/\epsilon)$ .

Our main tool is a new ‘conditional’ version of the regularity lemma for binary matrices (Lemma 1.6 below), which may be interesting on its own. We combine this with some methods similar to those of [10] (which is an expanded version of some results from [9] that do not appear here), to obtain the desired result.

We note that the study of such bipartite graph properties is an extension of the poset model studied in [10], in which the testability of properties is related to the logical complexity of their description. In this case the poset is the 2-dimensional  $n \times n$  grid, which as a poset is the product of two  $n$ -size total orders (lines). The language (syntax) includes the poset relation, the label unary relation (being labeled ‘1’), and in addition, the relations  $\text{row}(x_1, x_2)$  which state that  $x_1$  is on the same row as  $x_2$ , and similarly  $\text{col}(x_1, x_2)$  for columns.  $\forall$ -properties in this model are properties that can be described by a finite formula over a fixed number of variables with only  $\forall$  quantifiers in prenex normal form. Such properties would then correspond to exactly the properties that are characterized by a finite collection of forbidden submatrices (in a similar manner to what was done in [10] for the  $\forall$ -poset model). We call this model the ‘submatrix model’. The submatrix model is closely related to a sub-model of the (not always testable)  $\forall\exists$ -poset model, defined in [10].

The model ‘submatrix’ includes some interesting properties. In particular, the permutation-invariant properties in it are tightly connected to bipartite graph properties that are characterized by a collection of forbidden induced subgraphs:

**Definition 1.1** *For a finite collection  $F$  of 0/1 matrices, we denote by  $\mathcal{S}_F$  all 0/1-matrices that do not contain as a submatrix any row and/or column permutation of a member of  $F$ .*

**Observation 1.2** *Every bipartite graph property (where a bipartite graph is identified with its adjacency matrix in the usual way), that is characterized by a finite collection of forbidden induced subgraphs, is equivalent to a property  $\mathcal{S}_F$  for some finite set  $F$  of matrices. In addition, every  $\mathcal{S}_F$ -property in the ‘submatrix’ model is a bipartite graph property as above. ■*

The main result here is:

**Theorem 1.3** *Let  $F$  be a fixed collection of 0/1 matrices. Property  $\mathcal{S}_F$  is  $(\epsilon, \text{poly}(\frac{1}{\epsilon}))$ -testable for every  $\epsilon > 0$ , by a two sided error algorithm.*

The test above however is not only 2-sided, but it is also very computation-intensive (despite this computation using only a relatively small set of queries as data). However, using the tools presented in [14] (or more accurately, a very slight variation thereof), the existence of the above test implies the existence of a 1-sided test with a much smaller computation time.

**Theorem 1.4** *Let  $F$  be a fixed collection 0/1 matrices. Property  $\mathcal{S}_F$  is  $(\epsilon, \text{poly}(\frac{1}{\epsilon}))$ -testable for every  $\epsilon > 0$ , by a one sided error algorithm whose running time is polynomial in the time it takes to make the queries.*

The derivation of Theorem 1.4 from Theorem 1.3 is in Section 5. To present the test proving Theorem 1.3, we will need some machinery:

Let  $M$  be a 0/1-labeled,  $n \times m$  matrix. We denote by  $R(M)$  and  $C(M)$  the set of rows and the set of columns of  $M$  respectively. For an integer  $r$ , an  $r$ -partition of  $M$  is a partition of the set  $R(M)$  into  $r' \leq r$  parts  $\{R_1, \dots, R_{r'}\}$  and a partition of the set  $C(M)$  into  $r'' \leq r$  parts  $\{C_1, \dots, C_{r''}\}$ . Each submatrix of the form  $R_i \times C_j$  will be called a block (note that the coordinate sets defining the blocks do not necessarily consist of consecutive matrix coordinates). The weight of the  $(i, j)$  block is defined as  $\frac{1}{nm}|R_i||C_j|$ . We also define similar weights for the  $R_i$ 's and  $C_j$ 's, e.g.  $w(R_i) = \frac{1}{n}|R_i|$ .

For a block  $B$  of a 0/1-matrix  $M$ , we say that  $B$  is  $\delta$ -homogeneous if all but a  $\delta$ -fraction of its values are identical. If  $B$  is  $\delta$ -homogeneous we call the value that appears in at least a  $1 - \delta$  fraction of the places the  $\delta$ -dominant value of  $B$ . Note that this value is also  $\alpha$ -dominant for any  $\delta < \alpha < 1/2$ . We say that a value is the dominant value of  $B$  if it is the majority value in  $B$ .

**Definition 1.5** *Let  $\mathcal{P} = \{R_1, \dots, R_{r'}\} \times \{C_1, \dots, C_{r''}\}$  be an  $r$ -partition of  $M$ , and let  $\delta > 0$ . We say that  $\mathcal{P}$  is a  $(\delta, r)$ -partition if the total weight of the  $\delta$ -homogeneous blocks is at least  $1 - \delta$ .*

The key result is that an input that does not admit some  $(\delta, r)$ -partition can be rejected easily, because it will then contain many copies of every possible  $k \times k$  matrix (including the forbidden ones) as submatrices.

**Lemma 1.6** *Let  $k$  be fixed. For every  $\delta > 0$  and  $n \times n$ , 0/1-matrix  $M$ , with  $n > (\frac{k}{\delta})^{O(k)}$ , either  $M$  has a  $(\delta, r)$ -partition for  $r = r(\delta, k) \leq \text{poly}(k/\delta)$ , or for every 0/1-labeled  $k \times k$  matrix  $F$ , a  $g(\delta, k) \geq 1/\text{poly}(k/\delta)$  fraction of the  $k \times k$  submatrices of  $M$  are  $F$ . (The degree of each of the polynomials  $\text{poly}(k/\delta)$  above is a function of  $k$ ).*

This lemma allows us to reduce the testing problem to matrices that admit a  $(\delta, r)$ -partition for certain  $\delta, r$ , as for matrices that do not admit such partitions the lemma asserts that querying a random submatrix will find a counter example with sufficiently high probability. We note that the lemma is a conditional version of Szemerédi's Regularity Lemma ([17], see also [6, Chapter 7]), as a  $(\delta, r)$ -partition is in particular a regular partition in the sense of Szemerédi of the corresponding bipartite graph. The improvement over using directly the Regularity Lemma is achieved because of this conditioning. The proof of the lemma will be presented in Section 4.

We will then construct a test for matrices admitting a  $(\delta, r)$ -partition. This test will be very similar to the 2-sided boolean matrix poset test in [10]. However, the situation in the poset test is that the partition can be fixed in advance, while in our case there is the problem of ‘learning’ enough of the partition by sampling. For this we need some more machinery, that is described in Section 2, along with the framework of the proof of Theorem 1.3.

The plan of the paper is as follows. Section 2 includes some preliminaries, as well as a proof of Theorem 1.3 from two main lemmas, Lemma 1.6 above and Lemma 2.1 that is stated there. The lemmas themselves are proven in Section 4 and Section 3 respectively. Section 5 shows how to extend Theorem 1.3 to Theorem 1.4, and the final Section 6 contains some concluding open problems.

## 2 Partitions, signatures and Theorem 1.3

Assume that  $M$  has a  $(\delta, r)$ -partition. We have no hope, of course, to find it by  $O(1)$  many queries, as we cannot even sample a single point from each row. Hence, we will need here to define the ‘high level features’ of the  $(\delta, r)$ -partitions of  $M$ , that can be detected by sampling. This is given in the following.

Let  $M$  be a matrix with a  $(\delta, r)$ -partition  $\mathcal{P}$  defined by the row partition  $\{R_1, \dots, R_s\}$  and the column partition  $\{C_1, \dots, C_t\}$ ,  $s, t \leq r$ . Then  $\mathcal{P}$  naturally defines a high-level pattern which is an  $s \times t$  matrix of the dominant labels of the blocks. Formally, let  $P$  be a 0/1-labeled,  $s \times t$  matrix. A block  $R_i \times C_j$  is called  $\delta$ -good with respect to  $P$  if it is  $\delta$ -homogeneous and its dominant label is  $P_{i,j}$ .  $P$  is called a  $\delta$ -pattern of  $\mathcal{P}$  if all but at most a  $\delta$  fraction of the weighted blocks in  $\mathcal{P}$  are  $\delta$ -good with respect to  $P$ .

It is immediate from the definition that if a partition has a  $\delta$ -good pattern of size  $s \times t$ , then it is a  $(\delta, r)$ -partition with  $r = \max\{s, t\}$ . Conversely, if  $\mathcal{P}$  is a  $(\delta, r)$ -partition, then it has an  $r \times r$   $\delta$ -pattern (by possibly introducing empty blocks).

As the block sizes of a  $(\delta, r)$ -partition need not be fixed, we will also need information about the weights of  $R_i$  and  $C_j$ ,  $(i, j) \in [s] \times [t]$ : Let  $M$  be an  $n \times n$  matrix with a  $(\delta, r)$ -partition  $\mathcal{P}$  defined by the row partition  $\{R_1, \dots, R_s\}$  and the column partition  $\{C_1, \dots, C_t\}$ . Then a  $\delta$ -signature of  $\mathcal{P}$  is an  $s \times t$ , 0/1-labeled matrix  $P$  and two sequences  $\{\alpha_i\}_1^s, \{\beta_i\}_1^t$ , where  $P$  is a  $\delta$ -pattern of  $\mathcal{P}$ , and in addition  $\sum_{i=1}^s |\frac{|R_i|}{n} - \alpha_i| \leq \delta$  and  $\sum_{j=1}^t |\frac{|R_j|}{n} - \beta_j| \leq \delta$

Note that the signature of a partition is closed under permutations of rows and columns, namely, any row/column permutation of  $P$  with the respective permutations of  $\{\alpha_i\}_1^s$  and  $\{\beta_i\}_1^t$  is also a signature of the same matrix. Moreover, a signature of  $M$  is also a signature of all row/column permutations of  $M$ .

The signature of a partition has sufficient properties for constructing a test. Namely, it can be detected by sampling and it carries enough information to allow for a test, as asserted by the following:

**Lemma 2.1** *Let  $\delta < 1/81$  and assume that an  $n \times n$ , 0/1 matrix  $M$  has a  $(\delta, r)$ -partition. By*

sampling  $q = \text{poly}(r)$  many queries, a  $26\delta^{1/6}$ -signature of a  $(16\delta^{1/6}, 10r^2/(4\delta^{2/3}) + 1)$ -partition can be found, with success probability  $\frac{3}{4}$ .

We note that a test for a much closer approximation of the original  $(\delta, r)$ -partition can also be deduced from [13], with exponentially worse running time and query complexity. The proof of Lemma 2.1 is given in Section 3. We end the discussion by showing that together with Lemma 1.6 this indeed implies a 2-sided error test.

**Proof of Theorem 1.3:** Assume that we want to  $\epsilon$ -test  $M$  for a permutation invariant collection of forbidden induced  $k \times k$  submatrices. Blocks will now correspond to partition-blocks: Let  $\delta = (\frac{\epsilon}{300})^6$ , and let  $g = g(\delta, k)$ ,  $r = r(\delta, k)$  be those of Lemma 1.6. For  $4/g$  iterations, independently, we choose  $k$  random rows and  $k$  random columns of  $M$  and query all  $k^2$  points in the  $k \times k$  matrix that is defined by them. If we find a counter example in the queried points we answer ‘No’. Otherwise, by Lemma 1.6 we may assume with probability  $\frac{11}{12}$  that  $M$  has a  $(\delta, r)$ -partition.

If  $M$  has a  $(\delta, r)$ -partition, using Lemma 2.1 we can find an  $\frac{\epsilon}{8}$ -signature of an  $(\frac{\epsilon}{8}, 10r^2/4(\frac{\epsilon}{300})^4 + 1)$ -partition by sampling  $\text{poly}(r, \epsilon) = \text{poly}(\epsilon)$  queries. Let  $P$  with  $\{\alpha_i\}_1^s$  and  $\{\beta_i\}_1^t$  be an  $\frac{\epsilon}{8}$ -signature of such a partition. We form an  $n \times n$  matrix  $M_Q$  that represents our knowledge of  $M$ : We partition the rows of  $M_Q$  into  $s$  parts of weights  $\{\alpha_i\}_1^s$  and the columns into  $t$  parts of weights  $\{\beta_i\}_1^t$ . For every block of  $P$ , we set every entry of the corresponding block of  $M_Q$  to have the same label as in  $P$ . Now, let  $\mathcal{M}_{Q,\epsilon}$  be the set of all matrices that can be obtained from  $M_Q$  by changing at most  $\epsilon n^2/2$  entries in any possible way.

We check if any of the members of  $\mathcal{M}_{Q,\epsilon}$  has the property  $\mathcal{S}_F$ . If there is such a member, the algorithm answers ‘Yes’. Otherwise, if every member  $\mathcal{M}_{Q,\epsilon}$  contains a permutation of a forbidden submatrix, then the answer is ‘No’. Note, this last phase of the algorithm involves no additional queries and is just a computation phase.

To see that the algorithm is correct we first note that if a counter example is found in the first phase of the algorithm, then the input  $M$  does not have the property with probability 1. Hence the algorithm can err only in the second phase.

We claim that, with high probability, (a) some row/column permutation of  $M$  is a member of  $\mathcal{M}_{Q,\epsilon}$ , and (b) every two members of  $\mathcal{M}_{Q,\epsilon}$  are of distance at most  $\epsilon n^2$ . Indeed, assume that the signature that has been found is an  $\frac{\epsilon}{8}$ -signature of an  $(\frac{\epsilon}{8}, 10r^2/4(\frac{\epsilon}{300})^4 + 1)$ -partition of  $M$ . Then  $M_Q$  can be obtained from  $M$  by changing at most an  $\frac{\epsilon}{8}$ -fraction of the entries in each  $\frac{\epsilon}{8}$ -good block, followed by changing any of the entries in the non- $\frac{\epsilon}{8}$ -homogeneous blocks, and finally changing entries that are in strips around every block to compensate for the inaccuracy of the size sequences of the signature (whose sizes sum up to no more than  $\frac{\epsilon}{8}$  for the rows and  $\frac{\epsilon}{8}$  for the columns). The first two types of changes contribute at most an  $\frac{\epsilon}{8}$ -fraction of changes to the whole matrix each, and the last type contributes at most an  $\frac{\epsilon}{4}$ -fraction of changes. Thus  $M$  is at most  $\epsilon n^2/2$ -far from  $M_Q$ , and in particular  $M$  is in  $\mathcal{M}_{Q,\epsilon}$ . This proves (a), while (b) follows automatically from the definition of  $\mathcal{M}_{Q,\epsilon}$  and the triangle inequality.

Hence, we may assume that with probability at least  $\frac{2}{3}$ , the  $\frac{\epsilon}{8}$ -signature is computed correctly and (a) and (b) are satisfied. The failure probability here is the probability of not finding a copy of a forbidden induced submatrix in the case that  $M$  does not admits a  $(\delta, r)$ -partition, which is

at most  $\frac{1}{12}$ , plus the  $\frac{1}{4}$  bound on the error probability of finding a signature of the corresponding partition if such a partition exists, as asserted by Lemma 2.1. We conclude that if  $M$  has the property then certainly one member of  $\mathcal{M}_{Q,\epsilon}$  will have the property (as  $M$  itself is such a member by (a)), and thus the algorithm will accept. On the other hand, if  $M$  is more than  $\epsilon n^2$ -far from having the property, then no member of  $\mathcal{M}_{Q,\epsilon}$  can have the property by (b).

Clearly the query complexity of the test is  $\text{poly}(r, 1/\epsilon)$ , and by our expression for  $r$  it is  $\text{poly}(1/\epsilon)$ , which concludes the proof.  $\blacksquare$

**Remark:** In all the above we discussed forbidden *induced* subgraphs. Not having a forbidden subgraph (rather than induced subgraph) is a monotone decreasing property. In this case, the test is trivial, by density: For a large enough density, a Zarankiewicz (see [19], [12]) type theorem asserts that the answer ‘No’ is correct (as the graph will have a large enough complete bipartite graph), while if the density is low then the answer is trivially ‘Yes’, as the graph is close to the empty (edge-less) one. A thorough treatment of this case is found in [1].

We also remind the reader that although the calculation time (unlike the number of queries) has a bad dependence on the input size (this can be alleviated somewhat, but in light of the following we omit the details), we actually need to use only the mere existence of such a test in Section 5 to conclude that there is a much simpler test with a smaller calculation time. We now turn back to the proofs of Lemma 2.1 and Lemma 1.6.

### 3 $(\delta, r)$ -partitions, row similarity and the proof of Lemma 2.1

Our goal here is to show that by sampling  $\text{poly}(1/\delta, r)$  entries in  $M$ , one can detect the signature of a  $(\delta', r')$ -partition, if a  $(\delta, r)$  partition exists. For this we need a representation of a partition in a ‘local’ way, which is asserted by the following Claim 3.2 and Claim 3.3. To do this, we relate the notion of a  $(\delta, r)$ -partition to relative distances between rows and columns. For the rest of this section we assume that  $\delta$  is smaller than a global constant to be chosen later.

For two vectors  $u, v \in \{0, 1\}^m$  let  $\mu(u, v) = \frac{1}{m} |\{i \mid u_i \neq v_i\}|$ , namely,  $\mu(u, v)$  is the normalized hamming distance between the two vectors. We will use the following definitions.

**Definition 3.1** *Let  $M$  be an  $n \times n$  matrix. We set  $E^R(\mu(r_i, r_j))$  to be the expected value of  $\mu(r_i, r_j)$  where  $r_i, r_j$  are two rows of  $M$  chosen at random. Similarly let  $E^C(\mu(c_i, c_j))$  denote the respective quantity where  $c_i, c_j$  are two columns chosen at random.*

*Given a set of vectors  $V$  (usually either the set of rows or the set of columns of  $M$ ), and a partition  $V_0, \dots, V_s$  of  $V$ , we say that the partition is a  $(\delta, r)$ -clustering of  $V$  if  $s \leq r$ ,  $|V_0| \leq \delta|V|$ , and for every  $1 \leq i \leq r$  and  $u, v \in V_i$  we have  $\mu(u, v) \leq \delta$ .*

There is a close correlation between  $(\delta, r)$ -partitions of  $M$  and  $(\delta, r)$ -clusterings of its rows and columns, as the following two claims show.

**Claim 3.2** *Let  $M$  be a 0/1,  $m \times m$  matrix and assume that  $M$  has a  $(\delta, r)$  partition. Then there exists a  $(4\delta^{1/3}, r)$ -clustering of the rows of  $M$ , as well as a  $(4\delta^{1/3}, r)$ -clustering of the columns of  $M$ .*

**Claim 3.3** *Let  $M$  be a 0/1,  $m \times m$  matrix, and assume that  $\{R_0, \dots, R_s\}$  and  $\{C_0, \dots, C_t\}$  are  $(\delta^2, r)$ -clusterings of the set of rows and the set of columns respectively. Then these clusterings form also a  $(4\delta, r + 1)$ -partition of  $M$  for  $r = \max\{s, t\}$ .*

Moreover, for the above  $R_0, \dots, R_s$  and  $C_0, \dots, C_t$ , a  $4\delta$ -signature for the partition is given by the sequences  $\alpha_i = w(R_i)$ ,  $i = 0, \dots, s$ ,  $\beta_i = w(C_i)$ ,  $i = 0, \dots, t$ , and the  $s \times t$  matrix  $P$  where the  $(i, j)$  entry of  $P$  corresponds to the block  $R_i \times C_j$  and its label is the dominant label of this block.

Before we prove the two claims we need two simple observations, that in some sense correspond to the case “ $r = 1$ ” of the claims:

**Observation 3.4** *Let  $A$  be a 0/1 matrix. If  $A$  is  $\delta$ -homogeneous, then  $E^R(\mu(r_i, r_j)) \leq 2\delta$  and  $E^C(\mu(r_i, r_j)) \leq 2\delta$ .*

**Proof:** As  $A$  is  $\delta$ -homogeneous, we may assume without loss of generality that  $A$  contains less than a  $\delta$  fraction of 0’s. Hence, choosing two rows at random and picking a random place  $i$  in both, the probability that they are not both ‘1’ in this place is at most  $2\delta$ . Thus the expectation of the fraction of the number of places where they differ is bounded by  $2\delta$ , but this expectation is exactly  $E^R(\mu(r_i, r_j))$ . The proof for  $E^C(\mu(r_i, r_j))$  is analogous. ■

**Observation 3.5** *If  $A$  is a 0/1 matrix such that  $E^R(\mu(r_i, r_j)) < \delta$  and  $E^C(\mu(c_i, c_j)) < \delta$ , then  $A$  is  $4\delta$ -homogeneous.*

**Proof:** Assume on the contrary that  $A$  is not  $4\delta$ -homogeneous. This implies that when choosing two points from  $A$  independently and uniformly at random, with probability at least  $4\delta$  they will not have the same label. This is also a lower bound on the fraction of the  $2 \times 2$  submatrices that contain both 0’s and 1’s, as any two points with differing labels can be extended to such a submatrix. On the other hand, if  $E^R(\mu(r_i, r_j)) < \delta$ , then with probability more than  $1 - 2\delta$  both rows of a uniformly random  $2 \times 2$  submatrix are identical, as this matrix can be expressed as choosing two random places from two random rows. By the same token, if  $E^C(\mu(c_i, c_j)) < \delta$  then with probability more than  $1 - 2\delta$  the two columns of a random  $2 \times 2$  matrix are identical. Together these would have implied that less than a  $4\delta$  fraction of the  $2 \times 2$  submatrices have both 0’s and 1’s, a contradiction. ■

**Proof of Claim 3.2:** Assume that  $M$  has a  $(\delta, r)$ -partition defined by the row partition  $R_1, \dots, R_s$  and the column partition  $C_1, \dots, C_t$ ,  $s, t \leq r$ . For a partition block  $B$  and a row  $u$  that intersects  $B$ , let  $u|_B$  be the restriction of  $u$  to the columns in  $B$ . Assume that  $B$  is a  $\delta$ -homogeneous block that contains the rows of  $R_i$ . Then by Observation 3.4,  $E^R(u_B, v_B) \leq 2\delta$  for two rows chosen at

random from  $R_i$ . For a non  $\delta$ -homogeneous block, this expression is at most 1. Let  $w_i = w(R_i) = |R_i|/m$ ,  $i = 1, \dots, s$ , and let  $E_i(\mu(u, v))$  be the expectation of  $\mu(u, v)$  where  $u, v$  are two rows chosen uniformly at random from  $R_i$ . Then the above implies that  $\sum_{i=1}^r w_i E_i(\mu(u, v)) \leq (1-\delta)2\delta + \delta \cdot 1 \leq 3\delta$ , as this sum goes over all blocks and there are at least a  $(1-\delta)$  fraction of 0/1-blocks contributing at most  $2\delta$  each.

Now this implies that the total weight of the  $R_i$ 's for which  $E_i(\mu(u, v)) \geq \delta^{2/3}$  is at most  $3\delta^{1/3}$ . Let  $R_0$  be the union of all these  $R_i$ 's. Let  $R_1, \dots, R_{r'}$  be all other  $R_i$ 's, after renumbering. For every  $i = 1, \dots, r'$ , by our assumption,  $E_i(\mu(u, v)) < \delta^{2/3}$  for randomly chosen  $u, v$ , so there is an  $r_i \in R_i$  for which for at least a  $(1-\delta^{1/3})$  fraction of the  $v$ 's in  $R_i$ ,  $\mu(r_i, v) < \delta^{1/3}$ . Hence if we define for  $1 \leq i \leq r'$  the set  $R'_i = \{v \in R_i | \mu(r_i, v) < \delta^{1/3}\}$  and then define  $R'_0 = \bigcup_{i=1}^{r'} (R_i \setminus R'_i) \cup R_0$ , we obtain that  $R'_0, \dots, R'_{r'}$  is indeed a  $(4\delta^{1/3}, r)$ -clustering for the rows of  $M$ . The proof for the existence of a clustering of the columns is analogous.  $\blacksquare$

**Proof of Claim 3.3:** By the assumptions of the claim,  $|R_0| < \delta^2$ . Also, for any  $i \geq 1$  and any two rows  $u, v \in R_i$ ,  $\mu(u, v) \leq \delta^2$ . Thus for  $i = 1, \dots, s$ ,  $E_i(\mu(u, v)) \leq \delta^2$  where  $E_i$  is the expectation when  $u, v$  are chosen at random from  $R_i$ . Hence for the above partition into rows,  $\sum_{i=0}^s \frac{|R_i|}{m} E_i(\mu(u, v)) \leq \delta^2$  (as for each  $i > 1$  the corresponding term in this average is at most  $\delta^2$ , and for  $i = 0$  the weight of the term is at most  $\delta^2$ ). Similarly we get the analogous inequality for columns. Let  $\mathcal{P}$  be the partition of  $M$  into blocks that is defined by the cross product of the two partitions above.

Recall that  $\frac{|R_i|}{m}, \frac{|C_i|}{m}$  are the weights  $w(R_i), w(C_i)$  of the corresponding sets. Also, for a block  $B$ , let  $E_R(\mu(u|_B, v|_B)), E_C(\mu(u|_B, v|_B))$  be the expectation of  $\mu(\cdot, \cdot)$  for two rows  $u, v$ , respectively columns, chosen at random from  $B$ . By the law of complete probability,  $\sum_{i=0}^s w(R_i) \cdot E_i(\mu(u, v)) = E_B(E_R(\mu(u|_B, v|_B)))$ , where in the right hand side the outer expectation is on blocks of  $\mathcal{P}$  chosen according to their weights, and the inner expectation is on rows chosen at random in the block. Hence, the fact that  $\sum_{i=0}^s w(R_i) E_i(\mu(u, v)) \leq 2\delta^2$  implies that the total weight of all blocks  $B$  for which  $E_R(\mu(u|_B, v|_B)) > \delta$  is bounded by  $2\delta$ . By the same argument, for at most a  $2\delta$  fraction of the blocks  $E_C(\mu(u|_B, v|_B)) > \delta$ . Hence, for at least a  $1-4\delta$  fraction of the blocks (weighted by the block weights), both  $E_R(\mu(u|_B, v|_B)) \leq \delta$  and  $E_C(\mu(u|_B, v|_B)) \leq \delta$ . However, by Observation 3.5 above, each such block is  $4\delta$ -homogeneous, and hence at most a  $4\delta$  fraction of the blocks (measured by weights) are not  $4\delta$ -homogeneous. This implies that  $\mathcal{P}$  is a  $(4\delta, r+1)$ -partition. Also, by definition, the pattern of this partition is, for each block, the  $(1-4\delta)$ -dominant label of this block if there is one, or X otherwise. Moreover, as  $\alpha_i, \beta_i$  are the exact weights of the parts in the partition, we get a  $4\delta$ -signature for it.  $\blacksquare$

We are now ready to present the testing algorithm that yields Lemma 2.1. We start with its essential elements, starting with a trivial observation about approximating distances.

**Claim 3.6** *Let  $u, v \in \{0, 1\}^n$ ,  $\gamma < 1$ . Choose randomly and independently (with repetitions)  $m$  elements of  $[n]$ , naming the resulting (multi-)set  $L = \{l_1, \dots, l_m\}$ . Let  $\tilde{\mu}(u, v) = \frac{1}{m} \sum_{k=1}^m |u(l_k) - v(c_k)|$ , where  $u(i)$  and  $v(i)$  are the  $i$ 'th coordinate of  $u$  and  $v$  respectively. Then  $|\mu(u, v) - \tilde{\mu}(u, v)| \leq \gamma$  with probability at least  $1 - 2\exp(-\gamma^2 m)$ .*

**Proof:** Immediate by a Chernoff type inequality (See e.g [5, Corollary A.1.7]).  $\blacksquare$



We next construct a testing algorithm for an approximate notion of clustering. Testing algorithms for clustering were already investigated in [2]; here we will use a simple self-contained proof for an algorithm that gives an approximation in a very weak sense.

**Lemma 3.7** *If a set  $V$  of vectors over  $\{0, 1\}^n$  has a  $(\delta, r)$ -clustering, then there exists an approximate oracle algorithm that makes  $\text{poly}(r, \delta)$  bit queries (queries of one coordinate of one vector), and provides a  $(4\delta, 10r^2/\delta)$ -clustering of  $V$  in the following sense:*

*The algorithm makes  $\text{poly}(r, \delta)$  queries in a preprocessing step, and with probability at least 0.9 the situation will be that there exists a  $(4\delta, 10r^2/\delta)$ -clustering  $V'_0, \dots, V'_t$  of  $V$ , such that for every  $v \in V$  the algorithm makes  $\text{poly}(r, \delta)$  additional queries and provides the  $i$  for which  $v \in V'_i$ , giving the correct  $i$  for at least a  $(1 - 4\delta)$  fraction of the vectors in  $V$ .*

**Proof:** Suppose that  $V_0, \dots, V_s$  is a  $(\delta, r)$ -clustering of  $V$ . The algorithm starts by selecting uniformly at random  $r' = 10r^2/\delta$  vectors  $v_1, \dots, v_{r'}$  from  $V$ . With probability at least 0.95 (assuming that  $r$  is large enough) the situation is that for every  $1 \leq i \leq r$  for which  $|V_i| \geq \delta/r$ , we have picked at least one vector from  $V_i$ .

We now pick uniformly at random (with repetitions)  $l = 10r' \log r'/\delta$  coordinates from  $1, \dots, n$ , and let  $\tilde{\mu}(\cdot, \cdot)$  denote the corresponding approximated distance. Claim 3.6 implies that for every  $v, v' \in V$ , the probability for  $|\mu(v, v') - \tilde{\mu}(v, v')| > \frac{1}{2}\delta$  is bounded by  $\delta/20r'$ , and so with probability at least 0.95 the situation is that for at least a  $(1 - \delta)$  fraction of the vectors  $v \in V$ ,  $|\mu(v, v_i) - \tilde{\mu}(v, v_i)| \leq \frac{1}{2}\delta$  for every  $1 \leq i \leq r'$ .

Assuming that both of the above events occurred (which is the case with probability at least 0.9), we define  $V'_0, \dots, V'_{r'}$  as follows. Every vector  $v$  that belonged to  $V_0$ , or that belongs to a  $V_i$  of size  $|V_i| < \delta/r$ , or such that there exists some  $v_i$  for which  $|\mu(v, v_i) - \tilde{\mu}(v, v_i)| > \frac{1}{2}\delta$ , is placed in  $V'_0$ . For every other vector we let  $i$  be the index for which  $\tilde{\mu}(v, v_i)$  is minimal (or the smallest such index if there exist several values that minimize  $\tilde{\mu}(v, v_i)$ ), and define  $v$  to be in  $V'_i$ .

We now claim that  $V'_0, \dots, V'_{r'}$  is indeed a  $(4\delta, r')$ -clustering. First, it is easy to see that  $|V'_0| \leq 3\delta|V| < 4\delta|V|$  from the assumption on the size of  $V_0$ , and the guarantee that we have on the number of vectors for which the distance was not well approximated. Now, if  $u, v \in V'_i$  for some  $1 \leq i \leq r'$ , then we first note that  $\mu(u, v_i) \leq 2\delta$ . This is because if we denote by  $1 \leq j \leq r$  the index for which  $u \in V_j$ , then we have  $\mu(u, v_i) \leq \tilde{\mu}(u, v_i) + \frac{1}{2}\delta \leq \tilde{\mu}(u, v_j) + \frac{1}{2}\delta \leq \mu(u, v_j) + \delta \leq 2\delta$ . The same goes for proving that  $\mu(v, v_i) \leq 2\delta$ , and so by the triangle inequality  $\mu(u, v) \leq 4\delta$ . This concludes the claim about  $V'_0, \dots, V'_{r'}$ .

Now we can describe the remainder of the algorithm: After choosing  $v_1, \dots, v_{r'}$  and the  $l$  coordinates as above, the algorithm now queries each of these coordinates from each  $v_i$ , and by this concludes the preprocessing stage. For the oracle stage, given a vector  $v \in V$  the algorithm queries all the  $l$  chosen coordinates of  $v$ , and then calculates  $\tilde{\mu}(v, v_i)$  for every  $i$ . The algorithm then outputs the index  $i$  that minimizes this, or the smallest such index in case there is more than one. It is clear that the algorithm gives the correct index for every vector that is not in  $V'_0$ , whose size is bounded by  $4\delta$ , concluding the proof. ■

We note here that we could also use the above to find an approximate oracle for a  $(4\delta, r)$ -

clustering (instead of a  $(4\delta, 10r^2/\delta)$ -clustering), by trying to get from the set of queried vectors a subset  $V'$ , for which all but at most a  $3\delta$  fraction of the members of  $V$  are  $\delta$ -close to a member of  $V'$  (and verifying the validity of  $V'$  using a polynomial number of additional queries). This would also improve the dependencies in Lemma 2.1, but we omit it as our proofs already ensure the polynomial dependence on  $\epsilon$  without this improvement.

We are now ready to describe the algorithm that proves Lemma 2.1, by finding with probability  $\frac{3}{4}$  a signature of a  $(16\delta^{1/6}, r')$ -partition of  $M$ , if  $M$  has a  $(\delta, r)$ -partition.

### Algorithm Sig

- By Claim 3.2, there exist a  $(4\delta^{1/3}, r)$ -clustering of the rows. We perform the preprocessing stage of the algorithm provided by Lemma 3.7 to obtain an approximate oracle for a  $(16\delta^{1/3}, 10r^2/(4\delta^{2/3}))$ -clustering of the set of rows of  $M$ , denote it by  $R'_0, \dots, R'_{r'}$  for  $r' = 10r^2/(4\delta^{2/3})$ . Similarly, we obtain an approximate oracle for a  $(16\delta^{1/3}, r')$ -clustering  $C'_0, \dots, C'_{r'}$  of the columns.
- We now choose uniformly and independently at random (with repetitions) a (multi-)set  $R$  of  $l = 100r' \log r'/\delta$  rows of  $M$ , and for each of these we use the clustering oracle for  $R'_0, \dots, R'_{r'}$ . For  $1 \leq i \leq r'$ , we set  $\alpha_i$  to be the number of rows from  $R$  for which the oracle answered “ $i$ ”, divided by  $l$ . We do the analogous operation for a set  $C$  of  $l$  columns  $M$  that were uniformly and independently chosen (this time with respect to the oracle for  $C'_0, \dots, C'_{r'}$ ), and use it to set  $\beta_i$  for  $1 \leq i \leq r'$ . Both  $\alpha_0$  and  $\beta_0$  are set to 0, as the above oracles never correctly detect that a row is in  $R'_0$  or a column is in  $C'_0$ .
- Finally, for every  $1 \leq i \leq r'$  and  $1 \leq j \leq r'$  we look at the intersections of all the rows in  $R$  which the oracle located in  $R'_i$ , and all the rows in  $C$  which the oracle located in  $C'_j$ . We query the entries of  $M$  at the intersections, and set  $P_{i,j}$  to be the value (0 or 1) that has the majority of appearances in these queries.

We now claim that this is the required algorithm. First, we note that with probability at least 0.8, the oracles for both the clustering of the rows and the clustering of the columns are valid, as guaranteed by Lemma 3.7. In turn this guarantees that  $R'_0, \dots, R'_{r'}$  and  $C'_0, \dots, C'_{r'}$  form a  $(16\delta^{1/6}, r' + 1)$ -partition of  $M$ , by Claim 3.3. Also, each of the following occurs with probability at least 0.99:

- The difference between every  $\alpha_i$  and the fraction of the rows of  $M$  for which the oracle outputs “ $i$ ” is at most  $\delta/r'$ . This implies that  $\sum_{i=0}^{r'} \left| \frac{|R_i|}{n} - \alpha_i \right| \leq 2 \cdot 16\delta^{1/3} + r' \cdot \delta/r' < 33\delta^{1/3}$ .
- Similarly to the above,  $\sum_{i=0}^{r'} \left| \frac{|C_i|}{n} - \beta_i \right| < 33\delta^{1/3}$ . With the previous item this means that for all but at most a  $10\delta^{1/6}$  fraction of the pairs  $(i, j)$ , both  $\left| \frac{|R_i|}{n} - \alpha_i \right| \leq 7\delta^{1/6}$  and  $\left| \frac{|C_j|}{n} - \beta_j \right| \leq 7\delta^{1/6}$ .
- The fraction of appearances of “1” in the values taken under consideration when calculating  $P_{i,j}$ , differs from the fraction of appearances in the intersections of all rows assigned to “ $i$ ” and all columns assigned to “ $j$ ” (by the oracles) by no more than  $\delta$ . In addition, by the previous

item for all but at most a  $10\delta^{1/6}$  fraction of the pairs  $(i, j)$ , the above fraction differ by no more than  $14\delta^{1/6}$  from the fraction of appearances of “1” in  $R'_i \times C'_j$ , and so (if  $\delta$  is small enough) for the  $16\delta^{1/6}$ -homogeneous blocks among these,  $P_{i,j}$  will get the correct value. Therefore, the (weighted) fraction of wrong  $P_{i,j}$  labels is no more than  $16\delta^{1/6} + 10\delta^{1/6} = 26\delta^{1/6}$ .

Therefore with probability at least  $\frac{3}{4}$  all the above occurs (including the two oracles being valid), and a  $26\delta^{1/6}$ -signature of a  $16\delta^{1/6}$ -partition is obtained.  $\blacksquare$

As a final remark, the proof of Lemma 1.6 also uses an interim lemma about clusterings, Lemma 4.1 below. One could save further on the number of queries in the main theorem if the notion of  $(\delta, r)$ -clustering would be used throughout instead of the notion of  $(\delta, r)$ -partitions, but it would still be polynomial in  $\epsilon$ . However, the notion of  $(\delta, r)$ -partitions is more intuitive, and could have applications outside the scope of this work, so we use it instead.

## 4 Proof of Lemma 1.6

We use the same definition of a  $(\delta, r)$ -clustering (for sets of rows or columns) from the previous section. Claim 3.3 that was proven above implies that if  $A$  has a  $(\delta^2/4, t)$ -clustering for both its rows and its columns, then  $A$  admits a  $(\delta, t + 1)$ -partition. Therefore, the following lemma immediately implies Lemma 1.6. Moreover, it follows that Lemma 1.6 is true even if we insist on the forbidden submatrices obeying also the order of the rows and the columns of the input matrix (which is ignored for our use of a matrix as representing a bipartite graph).

**Lemma 4.1** *Let  $k$  be a fixed integer and let  $\delta > 0$  be a small real. For every  $n \times n$ , 0/1-matrix  $A$ , with  $n > (\frac{k}{\delta})^{O(k)}$ , either  $A$  admits  $(\delta, r)$ -clusterings for both the rows and columns with  $r \leq (k/\delta)^{O(k)}$ , or for every  $k \times k$ , 0/1 matrix  $B$ , at least a  $(\delta/k)^{O(k^2)}$  fraction of the  $k \times k$  (ordered) submatrices of  $A$  are copies of  $B$ .*

We should also note that the above estimate is essentially tight, as shown by a random  $n \times n$  matrix  $A$ , where each entry is independently chosen to be 1 with probability  $2\delta$ , and 0 with probability  $1 - 2\delta$ . The expected number of copies of the  $k \times k$  all 1 matrix in such a matrix is only a  $(2\delta)^{k^2}$  fraction of the total number of  $k \times k$  submatrices, and it is not difficult to check that with high probability  $A$  does not have a  $(\delta, o(n))$ -clustering for either its rows or its columns.

We will prove the lemma only for the clustering of the columns, because the proof for rows is virtually identical. We make no attempts to optimize the absolute constants and omit all floor and ceiling signs to simplify the presentation. In order to prove the above lemma, we first need the following simple corollary of Sauer’s Lemma [16, 18].

**Lemma 4.2** *For every  $t > 10k$ , every  $t \times t^{2k-1}$ , binary matrix  $M$  with no repeated columns contains every possible  $k \times k$ , binary matrix as a submatrix.*

**Proof:** By Sauer's Lemma [16, 18], every set of  $s = 1 + \sum_{i=0}^{k-1} \binom{t}{i}$  consecutive columns of  $M$  contains a  $k \times 2^k$  submatrix that has no repeated columns (and thus contains all  $2^k$  possible binary vectors as columns). Note that  $s < t^{k-1}$  and  $s(1 + (k+1)\binom{t}{k}) \leq t^{2k-1}$ . Thus  $M$  can be partitioned into at least  $(1 + (k+1)\binom{t}{k})$  blocks of size  $t \times s$ , each consisting of  $s$  consecutive columns. Considering these  $1 + (k-1) \cdot \binom{t}{k}$  pairwise disjoint consecutive blocks, we now find in each of them a  $k \times 2^k$  submatrix with no repeated columns. Considering now the set of  $k$  rows in each such submatrix, we obtain by the pigeonhole principle  $k$  such submatrices of size  $k \times 2^k$ , one following the other, and all having the same set of rows. This implies the desired result, as we can choose from each of the submatrices a desired column, and thus construct any  $k \times k$  matrix. ■

We now turn to the proof of Lemma 4.1. Fix  $\delta$  and  $k$ , and suppose that  $n$  is large enough (as a function of  $\delta$  and  $k$ , to be chosen later). Let  $t$  be the smallest integer for which  $(1 - \frac{1}{2}\delta)t^{4k-2} < 0.1$ . A simple computation shows that  $t = O(\frac{k}{\delta} \log(\frac{k}{\delta}))$ . Define  $T = t^{2k-1}$  and suppose that  $A$  is an  $n \times n$  matrix with 0/1 entries which does not have a  $\delta$ -clustering of the columns of size  $T$ . We have to show that in this case  $A$  must contain many copies of every  $k \times k$  matrix  $B$ .

Indeed let  $S$  be a random set of columns of  $A$  obtained by choosing, randomly, uniformly and independently (with repetitions)  $\tau = 5T/\delta$  columns of  $A$ . We choose  $n$  such that  $n > 10(\frac{5T}{\delta})^2$ . Note that in particular for such an  $n$ , with probability at least 9/10 no column is chosen more than once.

**Claim 4.3** *With probability at least 0.9,  $S$  contains  $T$  columns so that the Hamming distance between any pair of them is at least  $\frac{1}{2}\delta n$ .*

**Proof:** Let us choose the members of  $S$  one by one, and construct, greedily, a subset  $S'$  of  $S$  consisting of columns so that the Hamming distance between any pair of them is at least  $\frac{1}{2}\delta n$  as follows. The first member of  $S$  belongs to  $S'$ , and for all  $i > 1$ , the  $i$ 'th chosen column of  $S$  is added to  $S'$  if its Hamming distance from every previous member of  $S'$  is at least  $\frac{1}{2}\delta n$ . Since, by assumption, there is no  $(\delta, T)$ -clustering of the columns of  $A$ , as long as the cardinality of  $S'$  is smaller than  $T$ , the probability that the next chosen member of  $S$  will be added to  $S'$  is at least  $\delta$  (given any history of the previous choices); otherwise it would mean that the balls of radius  $\frac{1}{2}\delta n$  around the members of  $S'$  form a  $\delta$ -clustering. It thus follows that the probability that by the end of the procedure, the cardinality of  $S'$  will still be smaller than  $T$ , is at most the probability that a Binomial random variable with parameters  $5T/\delta$  and  $\delta$  will have value at most  $T$ . Hence this probability is smaller than 0.1, which implies the assertion of the claim. ■

The role of  $S'$  as above is indicated in the following claim. Let  $R$  be a random set of  $t$  rows of  $A$ , obtained by choosing  $t$  rows randomly and independently, with uniform distribution.

**Claim 4.4** *Let  $S'$  be a fixed set of  $T$  columns of  $A$  for which the pairwise Hamming distance is at least  $\frac{1}{2}\delta n$ . Then, with probability at least 0.9, all the projections of the members of  $S'$  on the rows in  $R$  are distinct.*

**Proof:** Let  $S'$  be a fixed set of  $T$  columns of  $A$  so that the Hamming distance between every pair is at least  $\frac{1}{2}\delta n$ . For any two fixed columns  $c_1, c_2 \in S'$  and a random row  $r$  we have that

$\text{Prob}_r(c_1[r] = c_2[r]) \leq 1 - \frac{1}{2}\delta$ , where  $c[j]$  is the  $j$ th coordinate of  $c$ . Hence, the expected number of pairs of members of  $S'$  whose projections on  $R$  are identical is at most  $\binom{T}{2}(1 - \frac{1}{2}\delta)^t < 0.1$ , where the last inequality follows from the choice of  $t$ . The desired result follows. ■

We can now conclude the proof of Lemma 4.1 as follows. Fix  $B$  to be any  $k \times k$ , 0/1 matrix. Choosing a random  $t \times \tau$  submatrix  $C$  of  $A$  is just like choosing a set  $R$  of  $t$  random rows and a set  $S$  of  $\tau$  random columns. By Claim 4.3, with probability at least 0.9, the set  $S$  of  $\tau$  columns contains a subset of the columns  $S'$  of size  $T$  that has pairwise distances at least  $\delta n$ . Given that this happens, by Claim 4.4 with probability 0.9 all the  $t$  projections of  $S'$  on the  $t$  rows of  $C$  are distinct. Hence with probability at least 0.8 (the probability that both events above hold) Lemma 4.2 assures that  $C$  contains  $B$  as a submatrix.

Now choosing a random  $k \times k$  submatrix of  $A$  can be viewed as first choosing a random  $t \times \tau$  matrix  $C$  as above and then choosing a random subset of  $k$  columns and  $k$  rows in  $C$ . Hence the probability that such a random  $k$  by  $k$  matrix will be identical to  $B$  is at least  $0.8 / \left( \binom{t}{k} \binom{\tau}{k} \right) = \left( \frac{\delta}{k} \right)^{O(k^2)}$ . ■

## 5 A 1-sided test for bipartite graphs

The test presented above for bipartite graphs is not only 2-sided, it is also computation-intensive. However, a slight variation of the tools presented in [14] changes the situation, so that the mere existence of any (possibly 2-sided) test for a bipartite graph property that is characterized by forbidden induced subgraphs implies the existence of a very simple 1-sided test.

**Definition 5.1** *A property  $\mathcal{P}$  of graphs is called hereditary if whenever a graph  $G$  satisfies  $\mathcal{P}$  and  $H$  is an induced subgraph of  $G$ , then  $H$  also satisfies  $\mathcal{P}$ .*

*For bipartite graphs the definition of hereditary properties is analogous, only here an induced subgraph  $H$  of  $G$  also inherits the corresponding restriction of the bipartition of  $G$ .*

For hereditary (non-bipartite) graph properties, the following results of Goldreich, Trevisan and Alon show that the existence of any test implies the existence of a simple 1-sided test.

**Lemma 5.2 (Canonical testers, [14])** *For any property  $\mathcal{P}$  of graphs, if there exists an  $\epsilon$ -test for  $\mathcal{P}$  making  $q$  queries, then there exists such a test that acts by uniformly sampling  $\text{poly}(q)$  vertices of the input graph and basing the decision deterministically on (the isomorphism class of) the subgraph induced on these vertices.*

**Lemma 5.3 (N. Alon, presented in [14])** *If  $\mathcal{P}$  is an hereditary property of graphs, and for every  $\epsilon$  there exists an  $\epsilon$ -test for  $\mathcal{P}$  making  $q(\epsilon)$  queries (independently of the number of vertices  $n$ ), then for every  $\epsilon$  there exists a 1-sided  $\epsilon$ -test that works by uniformly sampling  $\text{poly}(q(\epsilon))$  vertices of the input graph and accepting if and only if the induced subgraph on these vertices itself satisfies  $\mathcal{P}$ .*

We should stress here that for the above lemma to work, the original test has to have the  $q(\epsilon)$  bound for *every*  $n$ , not just a large enough  $n$ . In our application, although the 2-sided test has an implicit lower bound on  $n$ , this bound is itself a polynomial in  $q(\epsilon)$ , and so the test can be converted to a test suitable for such a lemma with only a polynomial penalty in the number of queries (for an  $n$  which is too small, we can simply query the entire input).

For our purpose we need the corresponding versions of the above lemmas for bipartite graphs. We omit most of the proofs because they are mostly word-for-word identical to the proofs of the original lemmas.

**Lemma 5.4** *For any property  $\mathcal{P}$  of bipartite graphs, if there exists an  $\epsilon$ -test for  $\mathcal{P}$  making  $q$  queries, then there exists such a test that acts by uniformly sampling  $\text{poly}(q)$  vertices from every color class of the input graph and basing the decision deterministically on (the isomorphism class of) the subgraph induced on these vertices.*

**Proof:** Let  $G$  be the input graph with color classes  $U$  and  $V$ . We only prove here the existence of a test that makes its queries by uniformly sampling  $q$  vertices from  $U$  and  $q$  vertices from  $V$  and then checking the induced subgraph. The rest of the proof of the statement of the lemma (ensuring that the test makes a deterministic decision based on the isomorphism class, with another polynomial penalty in the number of queries), is word-for-word identical to the proof of Lemma 5.2 in [14, Section 4].

Suppose that  $\mathcal{A}$  is a testing algorithm for  $\mathcal{P}$  making  $q$  queries. We construct a new testing algorithm as follows: We construct sets  $\emptyset = U_0 \subset U_1 \subset \dots \subset U_q \subseteq U$  and  $\emptyset = V_0 \subset V_1 \subset \dots \subset V_q \subseteq V$ . We follow the course of  $\mathcal{A}$ : For every  $1 \leq i \leq q$ , toward the  $i$ 'th query of  $\mathcal{A}$  we make sure by induction that our set of queries already includes the queries that  $\mathcal{A}$  would have made before the  $i$ 'th query. The case  $i = 1$  is trivial.

Suppose now that the  $i$ 'th query of  $\mathcal{A}$  is the pair  $(u_i, v_i)$ , where  $u \in U$  and  $v \in V$ . We construct  $U_{i+1}$  and  $V_{i+1}$  from  $U_i$  and  $V_i$  as follows. If  $u_i \notin U_i$ , we set  $U_{i+1} = U_i \cup \{u_i\}$ . Otherwise, we set  $U_{i+1}$  to be the union of  $U_i$  with an arbitrary additional vertex from  $U$ . We perform the analogous operation for constructing  $V_{i+1}$  from  $V_i$  and  $v_i$ . We then make all possible queries between  $U_{i+1}$  and  $V_{i+1}$ .

It is clear that the induction condition now holds toward the  $i + 1$ 'th query of  $\mathcal{A}$ , as well as that after all  $q$  queries of  $\mathcal{A}$  were done, our query set includes all required queries for accepting or rejecting the input according to the algorithm  $\mathcal{A}$ .

Now if we randomly permute the sets  $U$  and  $V$  before the beginning of this algorithm, then it is not hard to see that  $U_q$  and  $V_q$  are in fact uniformly random subsets of  $U$  and  $V$  respectively (regardless of the original algorithm), as required.  $\blacksquare$

**Lemma 5.5** *If  $\mathcal{P}$  is an hereditary property of bipartite graphs, and for every  $\epsilon$  there exists an  $\epsilon$ -test for  $\mathcal{P}$  making  $q(\epsilon)$  queries (independently of the number of vertices  $n$ ), then for every  $\epsilon$  there exists a 1-sided  $\epsilon$ -test that works by uniformly sampling  $\text{poly}(q(\epsilon))$  vertices of every color class of the input graph, and accepting if and only if the induced subgraph on these vertices itself satisfies  $\mathcal{P}$ .*

**Proof:** The proof of this lemma from Lemma 5.4 is virtually word-for-word identical to the proof of Lemma 5.3 from Lemma 5.2 that is presented in [14, Appendix D]. ■

**Proof of Theorem 1.4:** From Theorem 1.3 we know that there exists a 2-sided  $\epsilon$ -test for  $\mathcal{S}_F$  making  $\text{poly}(1/\epsilon)$  queries (for a fixed  $F$ ) for every  $\epsilon$ . Lemma 5.5 then immediately implies the required result. ■

## 6 Open problems

### More general combinatorial structures

A long standing question in graph property testing is that of whether there exists a test for the property of a (general) graph being triangle-free, whose number of queries is less than a tower function in  $\epsilon$ . Noting the “conditional regularity” nature of Lemma 1.6 here, one would hope for an analogue that will work for triangles. However, formulating such an analogue is not as simple as it seems: For example, there could be a completely bipartite graph that has the tower lower bound of [11] with regards to having a regular partition. Hence, the only hope would be of finding a partition in which most of the non-regular pairs are somehow labeled as “irrelevant” for the existence of a triangle in the graph. This still remains open; we already know however by [1] that, unlike the case of bipartite graphs, a polynomial dependency is not possible for this case.

Another interesting open question would be to formulate a lemma in the spirit of Lemma 1.6 for higher dimensional matrices, that would in turn correspond to  $r$ -partite  $r$ -uniform hypergraphs. Here too there is probably no avoiding the existence of “irrelevant” portions for which there is no regularity. Take for example any three dimensional matrix which is constant along the last dimension; it does not contain, for example, the  $2 \times 2 \times 2$  matrix that is all zero apart from exactly one entry, while it may still not admit any relatively small regular partition.

### Matrices with row and column order

This direction seems at the moment more accessible than those outlined above. It would be interesting to test a matrix for the property of not containing a member of a forbidden family of submatrices, with the same row and column orders (i.e. containing only a row or column permutation of a forbidden matrix is now allowed). Lemma 1.6 holds also for this framework, so the missing part would be “untangling” the sets of rows and columns in the resulting partition, in order to prove from this partition that one need only consider a set of possible input matrices that can be calculated from a small sample (as in the proof of Theorem 1.3).

A move from 2-sided testing to 1-sided testing is also no longer guaranteed, as the tools from [14] no longer work when the row and column ordering has to be preserved. However, a Ramsey-like lemma that was used in the old version of the proofs as they appear in [9] could help here, at the cost of an additional exponent (the 1-sided test in [9] was triply exponential, where one additional

exponent was from the use of the Ramsey-like lemma, and two others were from using an older version of Lemma 1.6).

### Non-binary matrices

It would also be interesting to prove the result for matrices that are not binary. It is enough to look at matrices with a fixed finite alphabet, because one does not need to distinguish between the different labels that do not appear in the finite set of forbidden matrices  $F$ .

Again “full conditional regularity” cannot be guaranteed, but this problem seems accessible (though perhaps with a no longer polynomial dependence of the number of queries on  $\epsilon$ ). A possible course of attack could be first partitioning into blocks so that each of which contains less than the full set of labels, and then recursively classifying each block as either “repartitionable” or “homogeneous” in a way somewhat reminiscent of what was done (more easily) in [10, 9] for poset properties.

### References

- [1] N. Alon, Testing subgraphs in large graphs, *Random Structures and Algorithms* 21 (2002), 359–370.
- [2] N. Alon, S. Dar, M. Parnas and D. Ron, Testing of clustering, *SIAM J. of Computing* 16(3):393–417, 2003.
- [3] N. Alon, E. Fischer, M. Krivelevich and M. Szegedy, Efficient testing of large graphs. *Combinatorica* 20:451–476, 2000.
- [4] N. Alon and A. Shapira, Testing subgraphs in directed graphs, *JCSS*, 69(3):354–382, 2004.
- [5] N. Alon and J. H. Spencer, The probabilistic method (second edition), John Wiley, 2000.
- [6] R. Diestel, *Graph Theory* (second edition), Springer, 2000.
- [7] E. Fischer, Testing graphs for colorability properties, *Random Structures and Algorithms*, in press. A preliminary version appeared in 12<sup>th</sup> *SODA Conference Proceedings*, pages 873–882, 2001.
- [8] E. Fischer, The art of uninformed decisions: A primer to property testing, *BEATCS (Computationa Complexity Column)* 75:97–126, 2001.
- [9] E. Fischer and I. Newman, Testing of matrix properties, In 33<sup>rd</sup> *ACM STOC Conference Proceedings*, pages 286–295, 2001.
- [10] E. Fischer and I. Newman, Testing of matrix-poset properties, manuscript.
- [11] W. T. Gowers, Lower bounds of tower type for Szemerédi’s Uniformity Lemma, *Geometric and Functional Analysis*, 7(2):322–337, 1997.



- [12] T. Köváry, V.T. Sós, and P. Turán, On a problem of K. Zarankiewicz, *Colloq. Math.*, 3:50–57, 1954.
- [13] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connections to learning and approximation. *JACM*, 45(4):653–750, 1998.
- [14] O. Goldreich and L. Trevisan, Three theorems regarding testing graph properties, *Random Structures and Algorithms* 23(1):23–57, 2003.
- [15] D. Ron, Property testing (a tutorial), In: *Handbook of Randomized Computing* (S. Rajasekaran, P. M. Pardalos, J. H. Reif and J. D. P. Rolim eds), Kluwer Press, Vol. II pages 597–649, 2001.
- [16] N. Sauer, On the density of families of sets, *J. Combinatorial Theory, Ser. A*, 13:145–147, 1972.
- [17] E. Szemerédi, Regular partitions of graphs, In: *Proc. Colloque Inter. CNRS No. 260* (J. C. Bermond, J. C. Fournier, M. Las Vergnas and D. Sotteau eds.), pages 399–401, 1978.
- [18] S. Shelah, A combinatorial problem: Stability and order for models and theories in infinitary languages, *Pacific Journal of Mathematics*, 41:247–261, 1972.
- [19] K. Zarankiewicz, Problem P 101. *Colloq. Math.*, 2:116–131, 1951.