Testing of Matrix Properties *

Eldar Fischer
NEC Research Institute and DIMACS†
fischer@research.nj.nec.com

Ilan Newman
Haifa University and NEC Research Institute‡
ilan@cs.haifa.ac.il

ABSTRACT

Combinatorial property testing deals with the following relaxation of decision problems: Given a fixed property P and an input f, distinguish between the case that f satisfies P, and the case that no input that differs from f in less than some fixed fraction of the places satisfies P. An (ϵ, q)-test for P is a randomized algorithm that queries at most q places of an input x and distinguishes with probability $2/3$ between the case that f has the property and the case that at least an ϵ-fraction of the places of f need to be changed in order for it to have the property.

Here we concentrate on labeled, d-dimensional grids, where the grid is viewed as a partially ordered set (poset) in the standard way (i.e as a product order of total orders). The main result here is an $(\epsilon, poly(1/\epsilon))$-test for every property of 0/1 labeled, d-dimensional grids that is characterized by a finite collection of forbidden induced posets. Such properties include the 'monotonicity' property and many other properties. A (less efficient) test for such properties with larger fixed size alphabets is also presented. Another result is a more efficient test than was previously known for a collection of bipartite graph properties.

Both collections above are variants of properties that are defined by certain first order formulae with no quantifier alternation over the syntax containing the grid order relations (and some additional relations for the bipartite graph properties). We also show that with one quantifier alternation, a certain property can be defined, for which no test with query complexity of $O(n^{1/10})$ exists. The above results identify new classes of efficiently testable properties.

Keywords

property testing, matrix properties

*A full version of this paper is available at
`http://www.cs.haifa.ac.il/ILAN/matrix-prop.ps`

†NEC, 4 Independence Way, Princeton NJ 08540, USA.

‡CS department, Haifa University, Haifa 31905, Israel

STOC'01, July 6-8, 2001, Hersonissos, Crete, Greece.

1. INTRODUCTION

Combinatorial property testing deals with the following relaxation of decision problems: Given a fixed property and an input f, one wants to decide whether f has the property or is 'far' from having the property. The general notion of property testing was first formulated by Rubinfeld and Sudan [15], who were motivated mainly by its connection to the study of program checking. The study of this notion for combinatorial objects, and mainly for labeled graphs, was introduced by Goldreich, Goldwasser and Ron [12]. A property is a collection of functions from a set (usually with some structure) to $\{0, 1\}$ (sometimes a larger range is considered). Being far is measured by the *hamming distance*, i.e. in how many places should the function be changed so as to have the property. An input function here is identified with a table of its 0/1 value for each point of the domain. A property is said to be (ϵ, q)-testable if there is a randomized algorithm that for every input function f queries the values of f on on at most q chosen points of the domain, and with probability $\frac{2}{3}$ distinguishes between the case that f has the property and the case that f is ϵn-far from having the property, where n is the size of the domain. When a property P is (ϵ, q)-testable with $q = q(\epsilon)$ (i.e. q is a function of ϵ only, and is independent of n) then we say that P is ϵ-testable; we say that P is testable if it is ϵ-testable for every $\epsilon > 0$.

Property testing has recently become quite an active research area, see e.g. [12, 13, 6, 4, 1, 2, 14, 9] for an incomplete list. Apart from its theoretical appeal, and the many questions it involves, it emerges in the context of PAC learning, program checking [10, 5, 15], probabilistically checkable proofs [3] and approximation algorithms [12].

One of the main tasks that emerged in the field, following [12], is to identify natural collections of properties that are efficiently testable (in terms of the number of queries). Goldreich et al. [12] studied some classes of properties (mainly graph properties) and identified many properties that are testable. Alon et al. [2] considered properties of functions $f : \{1, \ldots, n\} \longrightarrow \{0, 1\}$, namely, where each function is a binary string of length n. They suggested that properties that are defined by restricted logics might be testable. They proved that every regular language is testable, which is equivalent to saying that every property that is expressed by a certain second order monadic logic over ordered sequences is testable. Additional work in this line was done by [14], generalizing the above, and by [1, 9] on graph properties. In [8, 7, 6], the specific property of 'monotonicity' is studied.

Here we make another step in the direction established above: We present a logical model (including several variants) such that all properties that can be expressed in it are testable. Our structure is the d-dimensional grid $\{1, \ldots, n\}^d$, equipped with the natural product order. The language we use is that of first order formulae with the order relation. Our main result is that for every fixed d, every such formula using no quantifier alternation (i.e. using only one quantifier on a fixed sequence of points) is testable. Such properties have another equivalent formulation: Every such property is characterized by a finite collection of forbidden induced sub-posets. Such properties include 'monotonicity' (stating that there are no two points $x, y \;\; x \leq y$ that are labeled $1, 0$ respectively). The model also includes more complicated properties of chains, e.g: "all 2-dim, 0/1-labeled grids that when going towards the north-east direction in any possible way, contain no sequence of 4 points A,B,C,D labeled 0,1,0,1". More complicated examples are, e.g: "All 2-dim, 0/1-labeled grids that contain no four points A,B,C,D labeled by, say, 1,1,0,1 respectively and such that A is south-west (SW) of all, D is NE (north-east) of all, and neither B is SW of C, nor C is SW of B". We present an $(\epsilon, poly(1/\epsilon))$ 1-sided error test for such properties for every fixed dimension d. We also consider matrices over a larger (nonbinary) fixed alphabet (i.e. set of possible labels). For these we obtain a less efficient 1-sided test, which makes $exp(exp(poly(1/\epsilon)))$ many queries.

Another variant of this model dealing with forbidden matrices rather then posets leads to the following result: We show that every bipartite graph property that can be characterized by a finite collection of forbidden induced subgraphs is efficiently testable. The 2-sided error test that we construct has query complexity bounded by $exp(exp(poly(1/\epsilon)))$ (for 1-sided testing, the query complexity is triply exponential in $1/\epsilon$, and will be described only in the full version). Such bipartite graph properties were known to be testable by [1], but the previously known query complexity was $tower(tower(poly(1/\epsilon)))$, obtained by the bound on general graph properties using the regularity lemma. Our result here, first breaks the bound of using the regularity lemma for some collection of graph properties based on families of forbidden induced subgraphs[1]. Our main technical tool is a conditional version of the Regularity Lemma (Lemma 6.4) that may be interesting in its own right.

On the negative side, we show that there exists a poset (matrix) property expressed by using only one "$\forall\exists$" quantifier alternation which is not ϵ-testable for some ϵ. This puts a bound on the complexity of the relevant logical model that still guarantees testability. We also discuss several other variants of this model, some of which are efficiently testable, while in some, efficient testability remains open.

To put the results in perspective, we note that our model for the one dimensional case includes only regular properties, hence the result for this case is part of the results of [2]. A special instance of such properties (for all d) is the extensively studied property of 'monotonicity', namely the property: $\forall x_1, x_2 \left((x_1 \leq x_2) \rightarrow (M(x_2) \vee \neg M(x_1))\right)$, [7, 6] ('$M(x)$' means 'the value of the input matrix at x is 1'). On the other hand, one variant of the model (for $d = 2$) is tightly connected to bipartite graph properties. Hence, the model we suggest, apart from identifying a large class of efficiently testable properties, generalizes some previous results and bridges some others, allowing for a broader perspective.

The rest of the paper is organized as follows: Section 2 contains the basic model and preliminaries, in Section 3 we construct a simple to prove, 2-sided error, ϵ-test and then the 1-sided error test. Section 4 contains the test for properties of matrices over any fixed, non-binary alphabet. In Section 5 we construct a property that can be expressed as a formula with one quantifier alternation and which is not testable. In Section 6 we prove that bipartite graph properties that are characterized by a finite collection of forbidden induced subgraphs are efficiently testable. Finally, Section 7 poses some open problems.

[1]Bipartite graph properties that are characterized by forbidden subgraphs, rather then forbidden induced subgraphs, are trivially testable, see the end of Section 6. Unlike the case for properties defined by forbidden induced subgraphs, testing for partition-based graph properties, such as proper k-colorability, has polynomial bounds by [12].

2. PRELIMINARIES AND NOTATIONS

Let $[n] = \{0, \ldots, n-1\}$ with the natural order "$\leq$". The n-length d-dimensional grid is the poset (partially ordered set) $G(n, d) = [n]^d$ with the product order, namely $(\alpha_1, \ldots, \alpha_d) \leq (\beta_1, \ldots, \beta_d)$ if $\alpha_i \leq \beta_i$ for all $i = 1, \ldots, d$.

For $\alpha = (\alpha_1, \ldots, \alpha_d)$ and $\beta = (\beta_1, \ldots, \beta_d)$, if $\alpha_j = \beta_j$ for some $1 \leq j \leq d$, then we say that α and β *share a coordinate*.

A Boolean function $f : G(n, d) \longrightarrow \{0, 1\}$ is identified with a 0/1-labeling of the points of $G(n, d)$. Such a function will be called a 0/1 (n, d)-matrix, or just an (n, d)-matrix. Hence a property of functions on the structure $G(n, d)$ is just a set of (n, d)-matrices. For two (n, d)-matrices M and R their distance is $dist(M, R) = |\{x | \;\; M(x) \neq R(x)\}|$. For a matrix M and a property $\mathcal{P}$ we define $dist(M, \mathcal{P}) = min_{R \in \mathcal{P}} dist(M, R)$.

In the basic logical model that we study, the variables range over $G(n, d)$. The syntax includes the poset (binary) relation and the function unary relation $M(\cdot)$ ("being labeled 1"). Given a fixed set of variables, $x_1, \ldots, x_k$, a Boolean formula $\phi(x_1, \ldots, x_k)$ using the above relations specifies an allowed set of 0/1-labeled posets that are the truth assignment to it. The basic model of properties of (n, d)-matrices contains these properties that can be expressed as $\forall x_1, \ldots, x_k \phi(x_1, \ldots, x_k)$ where ϕ is a formula as above and k is a fixed constant (independent of n). For example, the following: $\forall x_1, x_2 \left((x_1 \leq x_2) \rightarrow (M(x_2) \vee \neg M(x_1))\right)$ is the well studied 'monotonicity' property. We call this model the $\forall$-poset model. Similarly the $\exists$-poset, the $\forall\exists$-poset models, etc. are defined.

Clearly if a property is expressible as an $\exists \phi(x_1, \ldots, x_k)$ formula then either every matrix is k-close to it, or the property is empty. Hence, such properties are trivially ϵ-testable. Things are starting to be interesting for $\forall$ properties. Our main result is that any such property is testable. We then show that there exists a $\forall\exists$-poset property that is not testable even for dimension $d = 2$.

We first note that testing a $\forall$-poset property is equivalent to testing that there is no forbidden fixed substructure: We say that a poset P is a *subgrid poset of dimension d* if there is a set of points in $G(n, d)$ (for some n) on which the induced order is isomorphic to P. We say that P is a 0/1-labeled poset if every point of P is labeled by 0 or 1. An (n, d)-matrix M contains a labeled poset P of size k if there are k points in M on which the order relation is isomorphic to P and the point labels are consistent with the labeling of P.

Definition 1. Let F be a set of 0/1-labeled posets. The following property of (n,d) matrices is defined: $\mathcal{M}_F = \{M | M$ contains no member of F as an induced labeled poset$\}$.

Observation 2.1. *Let $\mathcal{P}$ be a $\forall x_1,\ldots,x_k\phi(x_1,\ldots,x_k)$ type property of (n,d)-matrices. Then there is a set F of labeled k size posets, $|F| \leq 2^{k^2}$, such that $\mathcal{P} = \mathcal{M}_F$.*

Proof. We write $\exists x_1,\ldots,x_k\neg\phi(x_1,\ldots,x_k)$ for the negation of $\mathcal{P}$. In turn, we can write an equivalent DNF formula $\vee m_i(x_1,\ldots,x_k)$ for $\neg\phi$, where each $m_i(x_1,\ldots,x_k)$ represents a labeled poset F_i on at most k elements. Hence, a matrix satisfies $\mathcal{P}$ if and only if it has no F_i as a labeled-subgrid-poset. $\square$

Note that the above reduction is actually an equivalence.

3. ALGORITHMS FOR TESTING POSET PROPERTIES

We present here a 1-sided error test for $\mathcal{M}_F$ whose query complexity is polynomial in ϵ^{-1}. We start with some preliminaries, presented with a simple to prove, 2-sided error, algorithm that illustrates some of the basic ideas used throughout the following.

3.1 Preliminaries and a 2-sided error test

Let F be a finite collection of k-size posets. We present here a basic approach to matrix property testing, along with a 2-sided error ϵ-test for $\mathcal{M}_F$, whose number of queries is polynomial in ϵ^{-1} for any fixed dimension d.

Proposition 3.1. *For every fixed $\forall$ poset property of matrices, there exists a 2-sided ϵ-test which makes a number of queries which is polynomial in ϵ^{-1} and is independent of the size of the input.*

The proof of the proposition is implied by the algorithm below and the following Lemmata.

Let M be an (n,d)-matrix which we want to ϵ-test for $\mathcal{M}_F$. Let $m = (\frac{2^{k+1}k}{\epsilon^{k+1}})^{d-1}+1$. We divide M into m^d blocks of size $(n/m)^d$, by dividing $[n]$ into m, equal-size, intervals and taking Cartesian products. We query $\frac{8d\ln m}{\epsilon}$ uniformly random queries independently in each block of M. We tag each block as being 1, 0 or X according to the queries made: If all points that are queried in a block are labeled by '1' we tag it as '1', similarly if all points are labeled by '0' we tag it as '0'. Otherwise, we tag it as X. As a result we get an (m,d)-matrix, M_B, in which each entry represents a block of M and is labeled by 0,1 or X.

There are two major cases: If there are at least an $(\epsilon/2)$-fraction of the blocks that are tagged X we answer 'No'. We will prove in the sequel that in this case the algorithm is correct with probability 1.

The second case is when there are less than a fraction of $\epsilon/2$ of the blocks that are tagged X. In this case our intention is to check whether there is a matrix that is consistent with our knowledge of M as represented by M_B, and has the property. If we find such a matrix we answer 'Yes', if not we answer 'No'. Formally, Let M_Q be the following (n,d) matrix: For each 0-block of M, all corresponding entries of M_Q are '0', for each 1-block of M all corresponding entries in M_Q are '1'. The entries of M_Q that correspond to an X-block of M remain undefined. Now, each possibility of assigning 0/1 values to the undefined entries of M_Q and each possible choice of flipping the values in at most an $\epsilon/4$ fraction of the entries in every other block results in a 0/1-labeled matrix; we denote the set of all such matrices by $\mathcal{M}_{Q,\epsilon}$. We check if any of the members of $\mathcal{M}_{Q,\epsilon}$ has the property. If there is such a member, the algorithm answers 'Yes'. Otherwise, if every member of $\mathcal{M}_{Q,\epsilon}$ contains a member of F, the answer is 'No'. Note, this last phase of the algorithm involves no additional queries and is just a computation phase.

Clearly the query complexity of this algorithm is $poly(1/\epsilon)$ and is independent of n. We now show that it is correct with high probability. We first show that if the algorithm answers 'No' due to the fact that at least an $\epsilon/2$ fraction of the blocks are X-tagged, then it is correct with probability 1:

Claim 3.2. *If the fraction of X-blocks is at least $\epsilon/2$, then the queried locations in the X-blocks already contain a counter example to the property.*

Proof. Let M_B be the (m,d)-matrix defined above. We use the following d-dimensional Zarankiewicz [17], type theorem that finds a (k,d)-submatrix of a given label inside a large enough matrix with enough entries of this label.

Lemma 3.3 ([11]). *For $\delta < 1$ let M be an (m,d)-matrix in which at least δm^d of the entries are marked by 'X'. If $m > (\frac{k}{\delta^{k+1}})^{d-1}$ then there is a (k,d) submatrix all of whose entries are 'X'.*

Now, M_B satisfies Lemma 3.3 with $\delta = \frac{\epsilon}{2}$, thus there is a (k,d)-submatrix, W, of M_B which is all X. Our intent is to look back at the blocks of M that correspond to the entries of W. Each such block being tagged by X contains both a '0' and a '1' entry. Hence we want to argue that any labeled poset with k elements may be found within these blocks by choosing the right labeled entry. The only difficulty is that for two entries that share a coordinate in W and are comparable, the corresponding sampled entries in M might not be comparable. For this we use the following:

Lemma 3.4. *Any k-size d-dimensional grid poset can be embedded in $G(k,d)$ with no two points sharing a coordinate.*

Proof. It is enough to prove that P can be embedded in any d-dimensional grid with no two points sharing a coordinate, since we can then take the minimum subgrid containing it and it will clearly be a (k,d) grid.

Suppose that $f : P \rightarrow \{0,\ldots,m-1\}^d$ is any embedding of P into $G(m,d)$ for some m. Then we can think of f as a sequence of d functions $f_i : P \rightarrow \{0,\ldots,m-1\}$, $i = 1,\ldots,d$, so that $f_i(p)$ is the i-th coordinate of the point p in $G(m,d)$. We define a new embedding $f' : P \rightarrow G(m^{d+1},d)$ by the formula $f'(p) = m^d \cdot f(p) + \sum_{i=1}^{d} f_i(p) m^{i-1} \cdot (1,\ldots,1)$ (here $f(p)$ is taken as a d-dimensional row vector and $f(p')$ is defined as a linear combination of such vectors). It is not hard to see that this is also an embedding of the poset, and that no two members share a coordinate. $\square$

To end the proof of Claim 3.2, let P be an arbitrary member of F. By Lemma 3.4, P can be embedded in W with no two points sharing a coordinate. For each point of W to which a point of P was mapped, we choose an entry in the corresponding block of M that has the same value (such a point exists as the points of W correspond to X-tagged blocks). These entries are an embedding of P in M.

We now claim that with high probability, the tagging of the blocks of M essentially represents the true situation.

Definition 2. For a set Q of queries, we denote by G the following event: "Every 1-tagged block contains at most an $\epsilon/4$ fraction of '0' entries, and every 0-tagged block contains at most an $\epsilon/4$ fraction of '1' entries".

CLAIM 3.5. *With probability at least $1 - 1/m^d$ the event G happens.* □

The following now concludes the correctness proof:

CLAIM 3.6. *If there are less than an $\epsilon/2$ fraction of the blocks that are tagged by X , then the algorithm errs with probability bounded by $1/m^d$.*

PROOF. Assume that there are less than an $\epsilon/2$ fraction of the blocks that are tagged by X and that the event G has happened. If there exists a member of $\mathcal{M}_{Q,\epsilon}$ that has the property then we claim that M is close to having the property. Indeed, assuming that G happened, M is clearly at most $(\epsilon n^d/4)$-far from M_Q (disregarding the entries in the X-tagged blocks), while M_Q is at most $(3\epsilon n^d/4)$-far from any member of $\mathcal{M}_{Q,\epsilon}$ (counting *all* the entries in the X-tagged blocks), so M is at most (ϵn^d)-far from the member of $\mathcal{M}_{Q,\epsilon}$ having the property. Hence, in this case the algorithm may be wrong only if G did not happen, which occurs with a probability bounded by $1/m^d$.

If, on the other hand, no member of $\mathcal{M}_{Q,\epsilon}$ has the property, then clearly M does not have the property since given that G happened, M is in particular a member of $\mathcal{M}_{Q,\epsilon}$. Hence again the error probability is bounded by the probability that G does not happen. □

We note that the running time of the algorithm (as opposed to its query complexity) involves checking all possible $\mathcal{M}_{Q,\epsilon}$ resulting from M_Q. Although this seems to be exponential in n, it can be reduced considerably. However, since a better and faster algorithm is presented in the following, we omit the details.

The above scheme is used throughout this paper: We try to obtain an approximation of the input matrix which consists mostly of "simple" blocks. In the above case – blocks known to be almost monochromatic (all '0' or all '1').

3.2 More preliminaries – pseudomatrices and a Ramsey-like lemma

In the above algorithm use was made of the following fact: Given the (k, d) 'matrix of blocks' which are all tagged X, one could choose any entry (with the right label) from each block to find the counter-example. Such 'loose formations' of entries are formalized below.

Definition 3. A (k, d) *pseudogrid* is a subset $(x_i)_{i\in[k]^d}$ of the grid $G(n, d)$, such that if i and j do not share a coordinate, then $x_i \le x_j$ if and only if $i \le j$ (in the ordering of $G(k, d)$). Alternatively, $(x_i)_{i\in[k]^d}$ is a pseudogrid if and only if there exist $\{r_{a,l} | 1 \le a \le d, 0 \le l \le k\}$ (one may assume $r_{a,0} = 0$ and $r_{a,k} = n$) such that for every a and l the a'th coordinate of x_l is at least r_{a,l_a} but less than r_{a,l_a+1}. Namely, there exists a partition of $G(n, d)$ into k^d blocks such that $(x_i)_{i\in[k]^d}$ consists of exactly one member from each block. See Figure 1 for an example of a $(3, 2)$-pseudo grid.

A *pseudomatrix* in an (n, d) matrix A consists of a (k, d) pseudogrid in $G(n, d)$ and its labeling according to A. Given a (k, d) pseudomatrix in A defined by the pseudogrid $(x_i)_{i\in[k]^d}$, its *label matrix* is the (k, d) matrix B (which is not in itself a submatrix of A) consisting of the entries of the given pseudomatrix in the corresponding locations, that is, the one satisfying $B(i) = A(x_i)$ for all $i \in [k]^d$, see Figure 1.

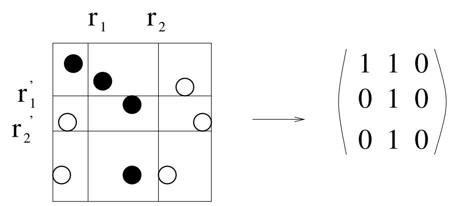

Figure 1: A $(3, 2)$-pseudogrid; bold dots for '1's, white ones for '0's. On the right is its label matrix.

Pseudomatrices will be used to show the existence of counter-examples within the queried values in 1-sided tests. Their main property is established in the following:

FACT 3.7. *Let A be a (k, d)-labeled pseudo matrix of M with a label matrix W. Assume that W contains a k size labeled poset F such that no two entries share a coordinate. Then M also contains F.* □

For 1-sided algorithms, to find a counter example we find a 'monochromatic' pseudomatrix in each block, using the following 'Ramsey-type' lemma.

LEMMA 3.8. *For every d and k there exists $c = c(d, k)$ such that for any $l \ge cm^c$, every 0/1-labeled (ml, d)-matrix which is partitioned into m^d blocks of size l^d, contains an (mk, d) pseudomatrix, so that its intersection with each block is a (k, d) pseudomatrix whose entries are all identical.*

In order to prove the lemma, we first need the following 1-dimensional version. In the sequel the locations in strings of length l are numbered by $0, \ldots, l-1$.

LEMMA 3.9. *For every k there exists $a = a(k)$ such that for any set of m binary strings of length $l \ge a \cdot m^a$, there exist $0 = r_0 < r_1 < \ldots < r_k = l$ such that each string contains a monochromatic substring (i.e. a substring all of whose entries are of identical values) with exactly one entry in the interval $[r_i, \ldots, r_{i+1} - 1]$ for every $i = 0, \ldots, k-1$.*

PROOF. It is enough to prove this for odd values of k. The proof is by induction. The basis $k = 1$ is trivial (it holds even for m strings of length 1).

Given that the theorem was proven for $k-2$, and $a(k-2)$ is known, we now prove it for k and find $a(k)$. We assume that none of the strings is monochromatic, as a monochromatic string will have a corresponding monochromatic substring for any $0 = r_0 < r_1 < \ldots < r_k = l$.

We first set the values of r_1 and r_{k-1} as follows: Let $l' = a(k-2)m^{a(k-2)}$, choose r_1 in the interval $[1, \ldots, l-1-l']$ uniformly at random, and set $r_{k-1} = r_1 + l'$.

For every fixed string s Let L_0 be the location of its first ('leftmost') '0' entry, L_1 the location of its first '1' entry, and R_0 and R_1 the locations of its last '0' and '1' entries respectively. We are interested in the event that none of $\{L_0, L_1, R_0, R_1\}$ fall into the interval $I = [r_1, \ldots, r_{k-1} - 1]$. The significance of this event is that if it occurs for a string s, then for any entry v in the interval $[r_1, ..., r_{k-1} - 1]$ there are two entries of s with an identical label as v, one in the interval $[0, ..., r_1 - 1]$, and one in the interval $[r_{k-1}, ..., l - 1]$.

Since one of L_0 and L_1 is equal to 0 and one of R_0 and R_1 is equal to $l - 1$, at most two of the four numbers above may fall into the interval I. Hence the probability that any of L_0, L_1, R_0 and R_1 is in the interval I is bounded by $2\frac{l'}{l-1-l'}$. It is now not hard to find $a = a(k)$ such that this probability is less than m^{-1} if $l \geq a \cdot m^a$. In particular, for such l there exist r_1 and a respective r_{k-1} for which the above event does not happen for any of the m strings.

We now consider for each string its substring of length $l' = a(k-2)m^{a(k-2)}$ consisting of the entries in the interval $[r_1, \ldots, r_{k-1} - 1]$. To these we apply the induction hypothesis, and find $r_2, \ldots, r_{k-2}$ so that each string contains a monochromatic substring with one entry in the interval $[r_i, \ldots, r_{i+1} - 1]$ for every $1 \leq i < k-1$. By the above discussion this substring can be extended to a monochromatic substring of the original string by finding identical labeled entries, one in the interval $[r_0, \ldots, r_1 - 1]$ and one in the interval $[r_{k-1}, \ldots, r_k - 1]$. $\square$

We note that there is an example that shows that a string length of $l = \frac{1}{4}m^2$ is insufficient for $k = 4$.

Proof of Lemma 3.8 (sketch). We shall use here the definition of pseudomatrices regarding the existence of a partition given by $\{r_{d',j} | 1 \leq d' \leq d, 0 \leq j \leq mk\}$ such that $(x_i)_{i\in[mk]^d}$ consists of one member from each block. In our case, the partition also has to satisfy $r_{d',k\cdot j} = l \cdot j$ for every $0 \leq j \leq m$ to ensure that the intersection of the (mk, d) pseudomatrix with each of the m^d blocks of A is a (k, d) pseudomatrix.

The proof is by induction on d. It is actually easier to prove the (seemingly stronger) statement that given p such matrices with $l \geq c(mp)^c$, there exists a corresponding pseudomatrix for each of them, all with the same partition $\{r_{d',j} | 1 \leq a \leq d, 0 \leq j \leq mk\}$ of $[n]^d$. The basis, $d = 1$, follows directly from Lemma 3.9, by cutting each string into m equal length substrings and applying Lemma 3.9 to these $p \cdot m$ new strings.

Suppose that we are given p matrices $A_0, \ldots, A_{p-1}$ over $[ml]^d$ for which we want to find the partition. Note that each matrix can actually be viewed as being a collection of m matrices where in the d-th dimension there is just one block. As we require that the partition will be the same for all matrices, these different m 'slices' will fit together to a partition of the original matrix.

The top level idea is the following: We look at the p matrices as $m \cdot p$ matrices that in the d dimension all contain just one block, and try to find the partition $\{r_{d,j}\}$ for the d-th dimension. To do this we just look at each l length 'string' along the d-th dimension and find the collection of $r_{d,j}$'s so that each string has a k-size monochromatic substring with exactly one point in each interval $[r_{d,j-1}, ..., r_{d,j} - 1]$. To do this we just use Lemma 3.9 (for suitable l and m); we do not actually find the partition for all $pl^{d-1}m^d$ strings of size l, but for a collection of $(pl')^{d-1}m^d$ strings, those for which every coordinate (between 1 and $d-1$) is in the range $[0, \ldots, l' - 1]$, modulo l. The choice of l' is described below.

Once this is done, we may replace each l-length string by the single value that is identical to the value of the monochromatic substring that is found. This reduces each of the $m \cdot p$, d-dimensional matrices to an $(l'm, d-1)$ matrix (after "deleting" the strings that were not considered for the partition above), and as explained, finding a common partition for these matrices will end the proof. However, by the induction hypothesis, a proper choice of l' ensures that such a partition for the $(d-1)$-dimensional case exists. Moreover, a closer examination of the arguments shows that as the chosen l' is polynomial in m and p, so is l. $\square$

3.3 A 1-sided test

Here we present the 1-sided test for $\mathcal{M}_F$. To avoid the dependence of a correctness of a 'No' answer on the event G happening, we will make additional queries in each block to ensure that if the approximated matrix is far from satisfying the property, then a counter-example to the property exists within the points already queried. In turn, this also avoids the necessity of checking all (n, d) matrices approximable by the queried values, as we need only to check the queries themselves. This makes also the running time of the algorithm polynomial in ϵ^{-1}.

Theorem 3.10. *For every fixed $\forall$ poset property of matrices, there exists a 1-sided error ϵ-test which makes $poly(\frac{1}{\epsilon})$ many queries, independently of the size of the input.*

For the sequel we need the following generalization of Lemma 3.4 to subblocks. Its proof is omitted.

Lemma 3.11. *Let $G(mk, d)$ and $G(ml, d)$ be considered as copies of $G(m, d)$ in which every entry is a (k, d) block and an (l, d) block respectively. Suppose that a k-size poset P embeds into $G(ml, d)$, then there is an embedding of P into $G(mk, d)$ so that the two embeddings map each point of P to the same respective block in $G(ml, d)$ and $G(mk, d)$, the mappings are isomorphic on corresponding blocks (in terms of the poset relation), and moreover no two points share a coordinate in $G(mk, d)$.*

Proof of Theorem 3.10. We present the 1-sided error test. We start exactly as in the algorithm of Subsection 3.1 by dividing M into m^d blocks of equal size, for $m = (\frac{2^{k+1}k}{\epsilon^{k+1}})^{d-1}+1$. We query $\frac{8d\ln m}{\epsilon}$ uniformly random queries independently in each block of M and tag each block as being 1, 0 or X according to the queries made: '1'/ '0' if all values that were queried inside the block are 1/0 respectively, and X otherwise. If there are at least an $\epsilon/2$ fraction of the blocks that are X-tagged then we answer 'No'.

If there are less than an $\epsilon/2$ fraction of the blocks that are X-tagged then we further divide each block into $l^d = (c \cdot m^c)^d$ subblocks (where $c = c(d, k)$ is the constant of Lemma 3.8) and query one arbitrary query in each subblock. We now set the outcome of the test as follows: We answer 'No' if there is a counter example among the queried points and answer 'Yes' otherwise.

Clearly the overall query complexity of this algorithm is polynomial in ϵ^{-1} (for a fixed d and k) and independent of n. We now prove its correctness.

Claim 3.12. *The algorithm is a 1-sided error algorithm with an error probability bounded by $1/m^d$.*

PROOF. To show correctness, let us analyze the various cases in which the algorithm may end. If after the first round of queries there are at least an $(\epsilon/2)$-fraction of the blocks that are X-tagged, then, exactly as in the proof of the 2-sided test, the algorithm answers 'No' and is correct.

Assume then that there are less than an $\epsilon/2$ fraction of the blocks that are X-tagged. We then have the second phase of queries in each subblock after which we re-tag each block according to the union of the old and new queries in it. Namely, some old 1-blocks and/or 0-blocks may become X-blocks. We may assume that there are less than an $\epsilon/2$ fraction of the blocks that are X, as otherwise we are back in the first case. Define now the (ml, d)-matrix, M_Q in which every entry corresponds to a subblock of M that is labeled by 0/1 as determined by the value of the query in this subblock. By Lemma 3.8 there is an (mk, d) pseudomatrix W, containing a monochromatic (k, d) pseudomatrix in each of the m^d blocks of M_Q. Note that in the natural correspondence between blocks of W and blocks of M, the label of every 0/1 block of M is identical to the label of the corresponding block of W (on X-blocks of M the label of the corresponding block of W may be either '0' or '1'). We claim that even when restricted to the points queried in the subblocks corresponding to the points of W, and have the decision of the algorithm according to whether they contain a counter example, it is correct (1-sided) with very high probability. Indeed, if a counter example is found among these points then certainly the algorithm rejects correctly.

Assume now that W does not contain a counter example. We claim that M is ϵn^d-close to have the property (with high probability). Indeed, assuming that the event G happened (as in Definition 2), we show how to obtain a matrix, M_W from M that has no counter example by changing at most ϵn^d entries of M: First we change the entries in each 0/1 blocks of M to have the label of their block. As in Section 3.1, we may assume that this will incur a change in at most $\frac{\epsilon}{4}n^d$ of the entries. We then change every entry in an X-block to have the label as in the corresponding intersection with W. As there are at most an $(\epsilon/2)$ fraction of X-blocks, this may incur in at most $\frac{\epsilon}{2}$ additional changed entries. Hence we get a matrix M_W, that is at most $\frac{3\epsilon}{4}n^d$-far from M. We claim that M_W has no counter example. Indeed, assume that M_W contains a counter example P, then by Lemma 3.11 (looking at M_W as an (ml, d)-matrix with $l = n/m$) there is a counter example in an (mk, d)-matrix in which its m^d-size blocks are labeled as the blocks of M_W. Moreover, no two points of this counter example share a coordinate, and so its existence in a label matrix of some pseudomatrix implies its existence in the pseudomatrix itself. Now the label matrix of W is such an (mk, d) matrix, and W in itself is a pseudomatrix contained in M_Q. Hence M_Q contains P with no two points sharing a coordinate. However, each entry of M_Q corresponds to a subblock of the actual input M that contains at least one queried point that is labeled by the same label as the corresponding entry of M_Q. Therefore we conclude that there is a counter example in the queried points of M in the subblocks that correspond to the points of W contrary to our assumption. □

The above completes the proof of Theorem 3.10.

Note, the algorithm is described as an adaptive one, however as queries at the second stage do not depend on the answers at the first stage, the algorithm is in fact non-adaptive.

4. TESTING OF NON-BINARY MATRICES

In this section we extend the result of the previous section to any fixed finite alphabet. The main result here is:

THEOREM 4.1. *Every $\forall$ poset property of matrices over a fixed finite alphabet can be 1-sided, ϵ-tested with query complexity and running time that are $exp(exp(poly(\epsilon^{-1})))$.*

4.1 Homogeneity in partitions and a strengthening of Lemma 3.8

The notion blocks being 'monochromatic' is not sufficient here. In particular, unlike in the 0/1 case, we cannot safely answer 'No' if many of the blocks are not 'monochromatic': It could be, for example, that a 0/1/2 matrix property is defined by an F all of whose members contain the label '2', while the partition of the input matrix into blocks may contain many blocks with both 0's and 1's.

In the new framework, we consider two partitions of a matrix at a time. We consider as before a partition P of M into m^d blocks, and a refinement achieved by repartitioning each block of the first partition into l^d subblocks, thereby obtaining a partition Q of M into $(ml)^d$ blocks. We label each block and each subblock with a subset of the alphabet Σ. In the construction of the tester this subset corresponds to the set of all labels found while querying entries from this block. Note that in particular a subblock is always labeled with a subset of the set used to label the whole block.

We will treat a block of P as 'featureless' if all its subblocks have the same label. However, in some cases it is only possible to find blocks which are close to satisfying this, which motivates the following definition.

Definition 4. A labeled set is called *ϵ-homogeneous* if all but at most an ϵ fraction of its labels are identical. For such a set we call this common label its *$(1-\epsilon)$-dominant label.*

Given a partition P of the matrix M and a refinement Q of P, and a block labeling as above, a block of P is called *ϵ-homogeneous* if its label is the $(1-\epsilon)$-dominant one for the set of the corresponding subblocks in Q with their labels.

Just as Lemma 3.8 was used in Section 3 to find a matrix which is both close to M and simple to check for $\mathcal{M}_F$, we shall use a similar lemma here. The pseudomatrix which we will find will be 'monochromatic' in each block, in the sense that all corresponding subblocks will have the same label (which is a subset of Σ). However, here we also need to ensure that for most blocks which are homogeneous enough, the subblocks will have the label of the block, so we will not need to modify many of the entries of M inside these blocks to arrive at the simplified matrix. We thus need the following strengthening of Lemma 3.8.

LEMMA 4.2. *For every h, d and k there exists $c = c(h, d, k)$ such that for any $l \geq cm^c$, every (ml, d)-matrix labeled with a set of h labels which is partitioned into m^d blocks of size l^d, contains an (mk, d) pseudomatrix, so that its intersection with each of these blocks is a (k, d) pseudomatrix whose entries are all identical, and moreover for all but at most an ϵ fraction of the $\frac{1}{c}\epsilon^c$-homogeneous blocks (for any ϵ), the common label of the intersection with them will be identical to the corresponding dominant label.*

The proof of this lemma follows closely the outline of the proof of Lemma 3.8, with additional care taken for the homogeneous blocks (the generalization to non-binary alphabets is not hard by itself). This proof is not presented here.

4.2 The test

Proof of Theorem 4.1 (sketch). Let F be a collection of forbidden Σ-labeled posets (for some fixed alphabet Σ), each of size at most k. In order to test for $\mathcal{M}_F$ we want to have two partitions of the matrix into blocks P, Q, where Q is a refinement of P with the following properties: Most blocks of P are δ-homogeneous with respect to the subblocks as defined by Q (for some small constant δ). In this case we will be able to conclude the test in a way similar to the 0/1 test. However, in order to arrive at such a situation, we cannot rely anymore on the assertion that if there are many non-homogeneous blocks then we have a counter example (as in the 0/1 case). Instead, we construct a sequence of partitions $P_0, \ldots, P_q$ of M, where each P_i is a refinement of the previous one. We will prove in the following that for such sequence, there are two members for which the above holds. The proof is by an iterative argument reminiscent of the proof of Szemerédi's Regularity Lemma. However, the dependency of the number of queries on ϵ will be only doubly exponential in a polynomial in ϵ, rather than a tower.

Formally, we choose $m_0 = 1$, and let $m_i = c \cdot m_{i-1}^{c+1}$, where $c = c(h, d, k)$ is the constant provided by Lemma 4.2 and h is set to $2^{|\Sigma|}$. Let P_0 have just one block and let $P_1, ..., P_q$ be a sequence of partitions, each P_i being a partition of M into m_i^d blocks which is a refinement of the previous one. We choose $q = \frac{|\Sigma| c(h,d,k)}{(\frac{1}{3}\epsilon)^{1+c(h,d,k)}}$, so $m_q = exp(exp(poly(1/\epsilon)))$.

The algorithm proceeds as follows. First, $\frac{6}{\epsilon}\ln((m_q)^d \cdot |\Sigma|)$ uniformly random queries are made independently in each block of P_q. If a counter example is found among the query points the algorithm rejects, otherwise it accepts.

It is clear from the formulation that the algorithm has a 1-sided error. We now show that it has bounded error for all inputs. Let us assume that there is no counter example within the set of the queried points. We will show that M is ϵn^d-close to having the property with high probability. To see this, we label each block of P_q with the set of all labels known to appear in it as a result of the queries made. We also label all the blocks of the other partitions (remember that P_q is a refinement of all of them) with the set of labels found while making queries within them. Denoting by G the event that no block of P_q (and hence no block of any P_j, $j < q$) contains more than a $\frac{1}{3}\epsilon$ fraction of entries of any label which did not appear in its queries, we note that with high probability G happens. The next step is to choose some $0 \leq p < q$, using the following claim.

Claim 4.3. *There exists $0 \leq p < q$ for which all but at most a $\frac{1}{3}\epsilon$ fraction of the blocks of P_p are $\frac{1}{c(h,d,k)}(\frac{1}{3}\epsilon)^{c(h,d,k)}$-homogeneous with respect to P_{p+1}.*

Proof. For every $\sigma \in \Sigma$, $0 \leq l < q$, and a set of blocks A of P_l, we denote by $\chi_{\sigma,l}(A)$ the fraction of blocks of P_l in A whose label contains the color σ. Let $\chi_l(A) = \Sigma_\sigma \chi_{\sigma,l}(A)$ and let $\psi_l = \chi_l(P_l)$. By a slight abuse of notation we let $\chi_{\sigma,l+1}(A)$ and $\chi_{l+1}(A)$ denote the respective quantities where the set of the subblocks of A from P_{l+1} is used. We claim that for any $\delta, \gamma > 0$, if P_l has more than a δ-fraction of blocks which are not γ-homogeneous, then $\psi_{l+1} \leq \psi_l - \delta \cdot \gamma$.

Indeed, we observe that for any $p' \leq l$ and any set of blocks A of p' $\chi_{p'}(A) \geq \chi_l(A)$. We also observe that $\psi_l = Pr(A)\chi_l(A) + (1 - Pr(A))\chi_l(P_l - A)$ (where $Pr(A)$ is the relative size of A to the whole matrix).

Now assume that P_l has more than a δ-fraction of blocks that are not γ-homogeneous, and let A be the set of all blocks of P_l that are not γ-homogeneous. Then $Pr(A) \geq \delta$. Also, by the above observation it is enough to show that $Pr(A)(\chi_l(A) - \chi_{l+1}(A)) \geq \delta \cdot \gamma$. By definition, for any $p \geq l$, $\chi_p(A) = \Sigma Pr(B)\chi_p(B))$ where B is a block in A and $Pr(B)$ is its relative size. However, for each block B, assuming that its label in P_l is $S, |S| = r \leq h$, then $\chi_l(B) = r$, while as B is not γ-homogeneous then $\chi_{l+1}(B) \leq Pr(S) \cdot r + (1 - Pr(S)) \cdot (r-1) \leq r - 1 + Pr(S) \leq r - \gamma$ (here $Pr(S)$ is the fraction of blocks of P_{l+1} in B that have label S). Hence $Pr(A)(\chi_l(A) - \chi_{l+1}(A)) \geq \delta \cdot \gamma$ as claimed.

To end the proof of the claim, note that since $\chi_0 \leq |\Sigma|$, if the claim does not hold for every p, it would imply that $\chi_q < 1$, which is a contradiction (every block label must contain at least one member of Σ). $\square$

We now return to the proof of the theorem: Choosing a p as in the formulation of the claim, we proceed as follows. We consider the (m_{p+1}, d) matrix consisting of the labels of the blocks of P_{p+1}, and its partition into $(m_p)^d$ blocks corresponding to the blocks of P_p. Over these we apply Lemma 4.2 to find a (km_p, d) pseudomatrix so that its intersection with every block from P_p is a monochromatic (k, d) pseudomatrix. By Lemma 4.2 for all but at most a $\frac{1}{3}\epsilon$ fraction of the homogeneous blocks in P_p, the restrictions of the pseudomatrix found to these blocks will have labels identical to the blocks. We let W be the label matrix corresponding to the above pseudomatrix.

We now consider the following (n, d) matrix M_W: For every entry of M whose label is in the set which is the common label of the intersection of the above pseudomatrix with the block from P_p, this will also be the entry of M_W in the respective location. Every other entry shall be replaced with an arbitrary member of the set labeling the intersection of the pseudomatrix with the entry's block. We then use M_W similarly to the 0/1 case to show that the algorithm is correct. We omit further details.

5. AN UNTESTABLE $\forall\exists$ PROPERTY

In this section we construct a $\forall\exists\phi(x_1, \ldots, x_k)$ property that is not ϵ-testable for some fixed ϵ. The construction is similar in spirit to the non-testable graph property constructed in [1]. In this abstract we construct such a property for a model which is stronger than the poset model. Based on this, a 0/1 poset-only $\forall\exists$ property which is not testable may be constructed. This latter example is deferred to the full version of this abstract. For the rest of this section we consider only 2-dimensional matrices, and use the notion of rows and columns in the usual matrix sense.

The property that we present uses the alphabet $\{0, 1, 2\}$: We say that a matrix satisfies the property 'permutation' if it is a row/column permutation of a symmetric matrix with all 2's on its primary diagonal, and no 2's anywhere else. It is not hard to see that this is equivalent to the matrix satisfying the following three conditions: (a) For every matrix entry other than '2' there is an entry on the same row and an entry on the same column which are both '2'. (b) For every '2' entry there is no other entry on the same row or column which is also '2'. (c) The matrix does not contain any row/column permutation of the matrix $\binom{2\ 0}{1\ 2}$.

Using the above equivalent formulation, the property 'permutation' can be expressed as a $\forall\exists\phi$ type property where ϕ

uses, apart from the order relation and the value relations ("the entry at location x is $0/1/2$"), the two additional relations: "x_1, x_2 are on the same row" and "x_1, x_2 are on the same column" (this model is discussed further in Section 6). The full version will contain a full proof of the proposition below, as well as the description of a non-testable poset-only $\forall\exists$ property of 0/1 matrices, derived from this one (through use of a suitable encoding).

PROPOSITION 5.1. *Every 0.1-test for property 'permutation' requires $\Omega(n^{1/10})$ even for two sided error.*

6. BIPARTITE GRAPH PROPERTIES

An important and natural variant of the poset-model is the model that allows to address properties that are characterized by a finite collection of forbidden submatrices. We restrict ourselves along the sequel to the 2-dimensional case. To put it in the framework of our study, we allow in our language, apart of the poset relation, the relations $row(x_1, x_2)$ which state that x_1 is on the same row as x_2, and similarly $col(x_1, x_2)$ for columns. The $\forall$-properties would then correspond to exactly the properties that are characterized by a finite collection of a forbidden submatrices (in a similar manner to Observation 2.1). We call this model the 'submatrix model'. It can be seen that the submatrix model lies between the $\forall$ and the $\forall\exists$ poset models.

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

Definition 5. For a finite collection F of 0/1 matrices, we denote by $\mathcal{S}_F$ all 0/1-matrices that do not contain any row and/or column permutation of a member of F.

OBSERVATION 6.1. *Every bipartite graph property (viewed as a collection of matrices) 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.* $\square$

Bipartite graph properties that are characterized by a finite set of forbidden induced subgraphs are testable: In [1], it was shown how to test $\forall$-graph properties for general graphs (not necessarily bipartite). However, the dependence on ϵ in these tests is $tower(tower(poly(1/\epsilon)))$, resulting from the use of the Regularity Lemma. Here we prove that such bipartite graph properties are testable (2-sided) with query complexity $exp(exp(poly(1/\epsilon)))$, hence, breaking the regularity lemma bounds in this case. Our main tool is a conditional (but efficient) version of the regularity lemma as stated in Lemma 6.4. The main result here is:

THEOREM 6.2. *Let F be a fixed collection of 2-dimensional matrices. Then $\mathcal{S}_F$ is $(\epsilon, exp(exp(poly(\frac{1}{\epsilon}))))$-testable, by a 2-sided error algorithm.*

Using a scheme involving an appropriate Ramsey-like lemma, it is also possible to arrive at a 1-sided test for these properties at the price of increasing the query complexity:

THEOREM 6.3. *Let F be a fixed collection of 2-dimensional matrices. Then $\mathcal{S}_F$ is $(\epsilon, exp(exp(exp(poly(\frac{1}{\epsilon})))))$-testable, by a one sided error algorithm.*

The proof of Theorem 6.3 is outside the scope of this abstract, and will appear in a future full version. To present the test proving Theorem 6.2, we will need some machinery:

Let M be a 0/1-labeled, $n \times m$ matrix. Let r be an integer, then an r-partition of M is a partition of the rows into $r' < r$ parts $\{R_1, \ldots, R_{r'}\}$ and a partition of the columns into $r'' < r$ parts $\{C_1, \ldots, C_{r''}\}$. Each submatrix of the form $R_i \times C_j$ will be called a block and its weight is defined by $\frac{1}{nm}|R_i||C_j|$. We also define similar weights for the R_i's and C_j's, e.g. $w(R_i) = \frac{1}{n}|R_i|$.

Definition 6. Let $\mathcal{P} = \{R_1, \ldots R_{r'}\} \times \{C_1, \ldots C_{r''}\}$ be an r-partition, and let $\delta > 0$. We say that $\mathcal{P}$ is a (δ, r)-*partition* if the total weight of the δ-homogeneous blocks is at least $1 - \delta$.

Unlike the poset model, in this model we cannot easily reject in the case that a given partition is not a (δ, r)-partition. However, it is still true that we can reject the input unless it admits *some* (δ, r)-partition, as the following lemma shows.

LEMMA 6.4. *Let k, δ be fixed, then for every $n \times n$, 0/1-matrix M, with n large enough, either M has a (δ, r)-partition for $r = r(\delta, k) \leq exp(exp(poly(1/\delta)))$, or for every 0/1-labeled $k \times k$ matrix F, a $g(\delta, k) > exp(-poly(1/\delta))$ fraction of the $k \times k$ submatrices of M are permutations of F.*

We note that Lemma 6.4 is a 'conditional' version of Szemerédi's Regularity Lemma, as a (δ, r)-partition is a regular partition in the sense of Szemerédi, of the corresponding bipartite graph. The improvement over using the Regularity Lemma is achieved because of this conditioning. The proof of the lemma is a direct consequence of Lemma 6.6 below.

Our next goal is to construct a test for such matrices. This test will be very similar to the test in Subsection 3.1, the main problem is to actually 'know' enough of the partition by sampling. For this we need some more machinery which is described in the following subsection.

6.1 Signatures and the general framework

Assume that M has a (δ, r)-partition. We have no hope to find it by $O(1)$ many queries as we cannot even sample a single point from each row. Hence, we need a 'high level feature' which motivates the following:

Let M be a matrix with a (δ, r) partition $\mathcal{P}$ defined by the row partition $\{R_1, \ldots, R_s\}$ and the column partition $\{C_1, \ldots, C_t\}$, $s, t \leq r$. $\mathcal{P}$ defines a $\{0/1/\text{X}\}$-labeled, $s \times t$ matrix P: The rows correspond to the sets $R_1, \ldots, R_s$, the columns to the sets $C_1, \ldots, C_t$, and the (i, j)-entry corresponds to the submatrix that is defined by R_i, C_j and is labeled by its $(1 - \delta)$-dominant label if it is δ-homogeneous, and otherwise by X. We say that P is the *pattern* of the partition $\mathcal{P}$. We also need information about the weights of R_i, C_i, $(i, j) \in [s] \times [t]$: Let M be an $n \times n$ matrix with a (δ, r)-partition $\mathcal{P}$ defined by the row partition $\{R_1, \ldots, R_s\}$ and the column partition $\{C_1, \ldots, C_t\}$. Then a γ-*signature* of $\mathcal{P}$ is an $s \times t$, $\{0/1/\text{X}\}$-labeled matrix P and two sequences $\{\alpha_i\}_1^s, \{\beta_i\}_1^t$, where P is a pattern of $\mathcal{P}$ and $|\frac{|R_i| \cdot |C_j|}{n^2} - \alpha_i \cdot \beta_j| \leq \frac{\gamma}{st}$ for every $(i, j) \in [s] \times [t]$.

Note that the signature of a partition is 'closed' under permutation of rows and columns; any row/column permutation of P with the respective permutations of $\{\alpha_i\}_1^s$ and $\{\beta_i\}_1^t$ is a signature of any row/column permutation of M.

The signature of a partition has sufficient properties for constructing a test, while it can be detected by sampling.

LEMMA 6.5. *Assume that an $n \times n$, 0/1 matrix M has a (δ, r)-partition with $r \geq \frac{100}{\delta^{10}}$, then by sampling $q = poly(r)$ many queries, a γ-signature of a $(12(\delta+\gamma)^{1/4}, r^4)$-partition can be found for any $\gamma \geq \frac{1}{r^{1/2}}$, with probability $1 - exp(-\gamma r)$.*

We note that a test for such partition can also be deduced from [12], with exponentially worse running time (and essentially the same type of query complexity). The proof of Lemma 6.5 will be presented in the full version of this abstract. We end this discussion by showing that it indeed implies a 2-sided error test.

PROOF OF THEOREM 6.2. Our 2-sided test follows the general idea of the test in Subsection 3.1. Blocks will now correspond to partition-blocks: Let $\delta_1 = (\frac{\epsilon}{2^7})^4$ and let $g = g(\delta_1, k)$, $r = r(\delta_1, k)$ of Lemma 6.4. For $\frac{1}{g^2}$ times, 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. If we find a counter example in the queried points we answer 'No'. Otherwise, by Lemma 6.4 we may assume that with probability $(1 - exp(-1/g))$ M has a (δ_1, r)-partition.

Let $\gamma = (\frac{\epsilon}{2^7})^4$ and let $\delta_2 = 12(\delta_1 + \gamma)^{1/4}$. Given that M has a (δ_1, r)-partition, using Lemma 6.5, we can find a γ-signature of a (δ_2, r^4)-partition by sampling $poly(r)$ queries. Let P and $\{\alpha_i\}_1^s, \{\beta_i\}_1^t$ be a signature of such partition. We form, as in Subsection 3.1, 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 each block of P, we set every entry of the corresponding block of M_Q to have the same label as in P. Now, exactly as in Subsection 3.1, each possibility of assigning 0/1 values to the X entries of M_Q and each possible choice of flipping the values in at most an $\epsilon/4$ fraction of the entries in every 0/1 block, results in a 0/1-labeled matrix; we denote the set of all such matrices by $\mathcal{M}_{Q,\epsilon}$. We check if any of the members of $\mathcal{M}_{Q,\epsilon}$ has the property. 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 the answer is 'No'. Note, this last phase of the algorithm involves no additional queries and is just a computation phase. Clearly the query complexity of the test is $poly(r, 1/\epsilon) = exp(exp(poly(1/\epsilon)))$.

To see that the algorithm is correct we first note that the algorithm can err only in the second phase. Next we claim that, with high probability, a row/column permutation of M is ϵn^2 close to each member of $\mathcal{M}_{Q,\epsilon}$. Indeed, assume that the signature that has been found is a γ signature of a (δ_2, r^4)-partition of M. Then, by a simple calculation, it follows that some row/column permutation of M is $(2\delta_2 + 2\gamma)n^2$-close to M_Q. Furthermore, M_Q is $\frac{\epsilon}{2}n^2$-close to any member of $\mathcal{M}_{Q,\epsilon}$, and hence M is $(\frac{\epsilon}{2} + 2\delta_2 + 2\gamma)$-close to any member of $\mathcal{M}_{Q,\epsilon}$. However, by our choice of $\delta_1, \delta_2, \gamma$, M is indeed ϵn^2-close to all members of $\mathcal{M}_{Q,\epsilon}$. The failure probability here is the sum of the failure of Lemma 6.4 and Lemma 6.5, which is exponentially small.

We conclude that, given that M is ϵn^2-close to all members of $\mathcal{M}_{Q,\epsilon}$, then if it has the property then certainly one member of $\mathcal{M}_{Q,\epsilon}$ will have the property (as M itself is such member). On the other hand if M is more than ϵn^2 far from having the property then certainly every member of $\mathcal{M}_{Q,\epsilon}$ cannot have the property. $\square$

Remark: In all the above we discussed forbidden *induced* subgraphs. Not having a forbidden subgraph (rather then induced subgraph) is a monotone decreasing property. In this case, the test is trivial, by density: For a large enough density, Lemma 3.3 asserts that the answer 'No' is correct (as the graph will have a large enough complete bipartite graph), while if the density is low the answer is trivially 'Yes', as the graph is close to the empty (edge-less) one.

6.2 Universal sets

Let M be a 0/1 matrix. Let $C = \{c_{i_1}, \ldots, c_{i_k}\}$ be an ordered set of columns of M and $\phi \in \{0,1\}^k$. Then ϕ defines the set of rows R_ϕ of M which consists of all rows whose intersection with the columns of C (in this order) is ϕ.

Definition 7. Let $0 < \alpha < 1$ and let M be a 0/1, $n \times m$ matrix. An ordered set of columns $C = \{c_{i_1}, \ldots, c_{i_k}\}$ is called (α, k)-universal if for every $\phi \in \{0,1\}^k$, $|R_\phi| \geq \alpha n$.

Lemma 6.4 follows directly from the following lemma.

LEMMA 6.6. *Let k be a fixed integer, $\delta > 0$, then for every $n \times m$, 0/1-matrix M, either M has a (δ, r)-partition for $r \leq exp(exp(poly(k, 1/\delta)))$ or at least $g(\delta, k)$-fraction of the ordered k-column sets are $(g(\delta, k), k)$-universal, where g is a function satisfying $g \geq exp(-poly(\delta, k)))$.*

Before we proof the lemma we start with an easier Claim:

CLAIM 6.7. *For every k,δ every 0/1 matrix either has a 2^{k+1}-partition for which the total weight of the δ-homogeneous blocks is at least $(1-\delta)\delta^{k-1}$, or at least a δ^k-fraction of its ordered sets of k-columns are δ^k-universal.*

PROOF. We prove the claim by induction on k. For $k = 1$, we may assume that M is not δ-homogeneous for otherwise it even has a $(\delta, 1)$-partition. Let C_M be the set of columns of M and let $C = \{c \in C_M | c \text{ is not } \delta\text{-homogeneous}\}$. If $|C| < \delta m$ then we have a partition of the columns into three parts: the set C, the set of columns that have at most a δ-fraction of 0's, and the set of columns with at most a δ-fraction of 1's. However, since in the above partition only C is not δ-homogeneous, its small size implies that this is a $(\delta, 3)$-partition. Hence, we now assume that $|C| \geq \delta m$. However, each $c \in C$ is $(\delta, 1)$-universal and therefore we have δm many $(\delta, 1)$-universal sets in M.

Assuming now that the claim has been proven for all $j < k$, we prove it for k: We may assume that M has a δ^{k-1}-fraction of δ^{k-1}-universal sequences, as otherwise, by the induction hypothesis M has a 2^k partition for which the total weight of the δ-homogeneous submatrices is at least $(1-\delta)\delta^{k-2}$, which is even stronger than what we need.

Let $C_{k-1} = \{c_1, \ldots, c_{k-1}\}$ be such a fixed sequence. C_{k-1} partitions the rows of M into 2^{k-1} sets: $\{R_\phi | \phi \in \{0,1\}^{k-1}\}$. We use this to define a partition $\{C\} \cup \{C_{\phi,0} | \phi \in \{0,1\}^{k-1}\} \cup \{C_{\phi,1} | \phi \in \{0,1\}^{k-1}\}$ of the columns into (up to) $2^k + 1$ sets: C is the set of columns whose intersection with every R_ϕ is not δ-homogeneous. For every other column, we pick arbitrarily one $\phi \in \{0,1\}^{k-1}$ such that the column is δ-homogeneous for R_ϕ, and put it in either $C_{\phi,0}$ or $C_{\phi,1}$ according its $(1-\delta)$-dominant label in the intersection.

By the definition of C, for any $c_k \in C$ the sequence $\{c_1, \ldots, c_k\}$ is $(\delta \cdot \delta^{k-1})$-universal. Hence, if C contains at least a δ fraction of the columns for all possible universal sequences C_{k-1} then we get at least a $\delta \cdot \delta^{k-1}$-fraction of the k ordered sets to be δ^k-universal sets, and we are done.

Let us assume therefore that C_{k-1} is a sequence for which $|C| < \delta \cdot m$. But then, each of the other sets of columns is δ-homogeneous for at least one ϕ, and that ϕ defines at least a δ^{k-1}-fraction of the rows. Thus, at least a δ^{k-1}-fraction of the total area in these columns is δ-homogeneous. As the columns not in C are at least a $1-\delta$ fraction of all columns, it follows that the total weight of the δ-homogeneous blocks in M is at least $(1-\delta)\delta^{k-1}$ as needed. $\square$

Proof idea of Lemma 6.6. Let $\delta_1 = \delta^2$. By Claim 6.7 we may assume that there is a 2^{k+1}-partition of M for which the total weight of the δ_1-homogeneous blocks is at least $(1-\delta_1)\delta_1^{k-1}$, as otherwise the claim already guarantees what we need. We amplify the fraction of area of δ_1-homogeneous submatrices by applying recursively the claim to each non δ_1-homogeneous submatrix, unless the fraction of k-sets that are δ^k-universal is large (in which case we are done).

If the process can be repeated $exp(poly(1/\epsilon))$ many times, then the total area of the δ_1-homogeneous blocks is at least $1-\delta_1$. However, this partition of the matrix is not a cross product of a row partition and a column partition. To make it a legitimate (δ, r)-partition, we end by taking the coarsest common refinement of the row partition in each block, and similarly for columns. $\square$

7. CONCLUDING REMARKS

We have seen that $\forall$-poset properties are testable, for $0/1$ matrices as well as matrices over any fixed finite alphabet, while some $\forall\exists$-poset properties are not testable. In addition we have seen that bipartite graph properties that are characterized by a finite set of forbidden induced subgraph are efficiently testable. This last model is a restricted case of the model of 'submatrix'. There are some major interesting problems that remain open regarding these models:

The situation with the interesting model of 'submatrix' is not yet understood. We showed that a relatively efficient test exists for the permutation invariant case using a 'conditional' Regularity Lemma (Lemma 6.4). While we can prove a conditional Regularity Lemma, similar to Lemma 6.4 for 2 dimensional matrices in general, we don't know yet how to construct a tester for this case. Another open question is to generalize the results for the binary case to any fixed, non binary alphabet. The most important question concerning this model is to obtain results for dimension higher than two, as this would related to testing of hypergraphs.

Back to the $\forall$-poset model: It is open whether more efficient test exist, especially in the case of non-binary alphabet. Another open problem is to better understand the model $\exists\forall$-poset. This latter model is related to some colorability problem in the spirit of [9], and currently the question whether properties in this model are testable is open.

Finally, other relations, apart of row, col (for 2-dim.) and the order may be used e.g, stating that two points are on the same row and on adjacent columns (and analogously for rows). The corresponding $\forall$ model also lies in between the $\forall$ and the $\forall\exists$ poset models. This generalizes the model 'submatrices' in the spirit of regular languages. We currently have no results for this model.

8. REFERENCES

[1] N. Alon, E. Fischer, M. Krivelevich and M. Szegedy, Efficient testing of large graphs. *Combinatorica* 20: 451–476, 2000.

[2] N. Alon, M. Krivelevich, I. Newman and M. Szegedy, Regular Languages are Testable with a Constant Number of Queries, In 40^{th} *IEEE FOCS Conference Proceedings*, pages 645–655, 1999.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy, Proof verification and the hardness of approximation problems, *JACM*, 45(3):501–555, 1998.

[4] T. Batu, R. Rubinfeld, P. White, Fast Approximate PCPs for Multidimensional Bin-Packing Problems, In *RANDOM-APPROX99 Conference Proceedings*, pages 245–256, 1999.

[5] M. Blum, M. Luby and R. Rubinfeld, Self-testing/correcting with applications to numerical problems. *JCSS*, 47:549–595, 1994.

[6] Y. Dodis, O. Goldreich, E. Lehman, S. Raskhodnikova, D. Ron and A. Samorodnitsky, Improved testing algorithms for monotonicity, In *3rd RANDOM Conference Proceedings*, pages 97–108, Springer-Verlag, Berkeley CA, 1999.

[7] F. Ergün, S. Kannan, S. R. Kumar, R. Rubinfeld and M. Viswanathan, Spot checkers, *JCSS*, 60(3):717–751, 2000.

[8] E. Fischer, On the strength of comparisons in property testing, manuscript.

[9] E. Fischer, Testing graphs for colorability properties, In 12^{th} *SODA Conference Proceedings*, pages 873–882, 2001.

[10] P. Gemmell, R. Lipton, R. Rubinfeld, M. Sudan and A. Wigderson, Self-testing/correcting for polynomials and for approximate functions, In 23^{th} *ACM STOC Conference Proceedings*, pages 32–42, 1991.

[11] T. Kövary, V.T. Sós, and P. Turán, On a problem of K. Zarankiewicz, *Colloq. Math.*, 3:50–57, 1954.

[12] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connections to learning and approximation. *JACM*, 45(4):653–750, 1998.

[13] O. Goldreich and D. Ron, Property testing in bounded degree graphs. In 29^{th} *ACM STOC Conference Proceedings*, pages 406–415, 1997.

[14] I. Newman, Testing of Function that have small width Branching Programs, In 41^{st} *IEEE FOCS Conference Proceedings*, pages 251– 258, 2000.

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

[16] A. C. Yao, Probabilistic computation, towards a unified measure of complexity. In 18^{th} *IEEE FOCS Conference Proceedings*, pages 222–227, 1977.

[17] K. Zarankiewicz, Problem P 101. *Colloq. Math.*, 2:116–131, 1951.