Functions that have read-twice constant width branching programs are not necessarily testable

Eldar Fischer
Department of Computer Science
The Technion
Haifa, Israel
Email: eldar@cs.technion.ac.il

Ilan Newman
Department of Computer Science
University of Haifa
Haifa, Israel
and: NEC Research Institute
Email: ilan@cs.haifa.ac.il

Abstract

We construct a property on 0/1-strings that has a representation by a collection of width 3, read-twice oblivious branching programs, but for which any 2-sided ϵ-testing algorithm must make at least $\Omega(n^{1/10})$ many queries for some fixed small enough ϵ. This shows that Newman's result [15] cannot be generalized to read-k-times functions for $k > 1$.

1 Introduction

Combinatorial property testing deals with the following relaxation of decision problems: Given a fixed property and an input x, one wants to decide whether x has the property or is 'far' from having the property. The general notion of property testing was first formulated by Rubinfeld and Sudan [17] and first studied for combinatorial objects by Goldreich, Goldwasser and Ron [13]. A property in this respect is a collection of 0/1 strings, and being far is measured by the *hamming distance*, namely, in how many places should an input string x be changed so as to have the property. A property is said to be (ϵ, q)-testable if there is a randomized algorithm that for every input x of size n queries the values of x on on at most q chosen points of the domain, and with probability at least $\frac{2}{3}$ distinguishes between the case that x has the property and the case that x is ϵn-far from having the property. 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. [13, 14, 7, 5, 2, 3, 15, 9, 8] 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 [12, 6, 17], probabilistically checkable proofs [4], and approximation algorithms [13]. For recent surveys on property testing see [10, 16].

One of the main tasks that emerged in the field, following [13], is to identify natural collections of properties that are efficiently testable (in terms of the number of queries). Goldreich et al. [13] studied some classes of properties (mainly graph properties) and identified many properties that are testable. Newman [15], following Alon et al. [3], proved that any property that can be computed by an oblivious read-once constant-width branching program is testable (and with a 1-sided error). Here we prove that this cannot be generalized to read-twice branching programs, namely, we exhibit a property that can be computed by a width 3 read-twice oblivious branching program, but for which any 2-sided $3.5 \cdot 10^{-8}$-test requires at least $\Omega(n^{1/10})$ many queries.

The property that we construct can be represented by a CNF formula with a constant clause size and such that each variable appears in at most a constant number of clauses. Hence we also prove that such properties, in particular properties with constant size 0-witnesses, are not efficiently testable in general (this also follows from the result of [11] regarding hard to test monotonicity properties, but there it does not have the extra condition of each variable appearing in a bounded number of clauses and the bound for adaptive algorithms is currently much lower).

2 Preliminaries and notation

We identify properties with the collection of their characteristic Boolean functions. Namely, a property $\mathcal{P} \subseteq \{0,1\}^*$ is identified with $\{f_n : \{0,1\}^n \longrightarrow \{0,1\}\}$ such that $f_n(x) = 1$ if and only if $x \in \mathcal{P}$.

An oblivious leveled branching program (BP) here is a directed graph, B, in which the nodes are partitioned into levels $L_1, \ldots, L_{m+1}$. There are two special nodes; a 'start' node belonging to L_1 and an 'accept' node belonging to L_{m+1}. All edges are between nodes of consecutive levels, and each node apart from those on L_{m+1} has exactly two out-going edges to the next level, one of which is labeled by '0' and the other by '1'. In addition, every level L_i apart from L_{m+1} is associated with a member l_i of $\{1, \ldots, n\}$, that refers to a location of a bit in the input. An input $x \in \{0,1\}^n$ naturally defines a path in B: It contains one node from each level, starting with the start-node at L_1; given the choice of a node from a level L_i, $1 \leq i \leq m$, its outgoing edge which is labeled by x_{l_i} is followed to select the node at L_{i+1}. A leveled BP defines naturally a Boolean function $g : \{0,1\}^n \longrightarrow \{0,1\}$ in the following way: $g(x) = 1$ if the path that x defines reaches *accept*, otherwise $g(x) = 0$. The *width* of a leveled BP is the size of its largest level.

A *read-k-times* BP for a function $g : \{0,1\}^n \longrightarrow \{0,1\}$ is an oblivious leveled branching program computing g, in which no member of $\{1, \ldots, n\}$ is associated with more than k of the levels $L_1, \ldots, L_m$.

In the following we denote by F_2 the field of two elements. We denote by $\oplus$ the addition in F_2. We omit all floor and ceiling signs whenever the implicit assumption that a quantity is integer makes no essential difference. We make no attempt to optimize the coefficients involved, just the function types (e.g. polynomial versus logarithmic).

3 The main result

Theorem 3.1 *There exists a property $\mathcal{P}$ that has a width 3, read-twice, oblivious branching program which is not ϵ-testable for $\epsilon < 3.5 \cdot 10^{-8}$ even with a 2-sided error.*

The proof of Theorem 3.1 follows from the discussion and the proof of Theorem 3.3 below.

Our aim is to define a function that has a read-twice constant width BP but is not efficiently testable. We first define the following structure. Let $V = [n] \times [m]$. We think of V as the disjoint union of m disjoint vertex sets of size n, $V_1, \ldots, V_m$. Let $\mathcal{G} = \{G_i = (V_i, V_{i+1}, E_i)\}_{i=1}^{m-1}$ be a sequence of bipartite graphs. We define a Boolean circuit as follows. A Boolean variable X_v is associated with each $v \in V_1$. Each $v \in V_i$, $i > 1$, is associated with a parity function whose inputs are the vertices of V_{i-1} adjacent to v according to G_{i-1}. In particular, this circuit assigns to each v a linear function (over F_2) over the formal Boolean variables $\{X_u | u \in V_1\}$.

Given $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$, the circuit $C_\mathcal{G}$ is associated with a property $\mathcal{P}_\mathcal{G} \subseteq \{0,1\}^V$ in the following way: Suppose a Boolean value x_v is assigned for each node v. This assignment is said to have $\mathcal{P}_\mathcal{G}$ if for every $i = 2, \ldots, m$ and every $v \in V_i$, $x_v = \bigoplus_{(u,v) \in E_{i-1}} x_u$. Namely, the property is satisfied if the assignments to the nodes represent a consistent computation of $C_\mathcal{G}$, where $\{x_u | u \in V_1\}$ represent the values of the original formal variables $\{X_u | u \in V_1\}$.

$\mathcal{P}_\mathcal{G}$ in general does not have a read-twice BP. However, it is associated with the semantically close property $\mathcal{P}'_\mathcal{G}$ that is constructed by adding a new Boolean value $x_{(u,v)}$ for each edge $(u,v) \in G_i, i = 1, \ldots, m-1$. We think of the edge values as carrying the Boolean values between the nodes of the circuit. Namely, an assignment has property $\mathcal{P}'_\mathcal{G}$ if for every node $v \in V_i, x_v = x_{(v,w)}$ for every edge $(v,w) \in E_i$, and $x_v = \bigoplus_{(u,v) \in E_{i-1}} x_{(u,v)}$. Letting $E = \cup E_i$ we note that $\mathcal{P}'_\mathcal{G} \subseteq \{0,1\}^{V \cup E}$.

For $v \in V_{i+1}$ we let $N(v) = \{u | \ (u,v) \in E_i\}$, namely, $N(v)$ is the set of 'inputs' to the gate v in the description of the circuit above. Also let $\Delta(\mathcal{G}) = max_{v \in V} |N(v)|$.

Claim 3.2 *The following hold with regards to $\mathcal{P}'_\mathcal{G}$.*

1. *For every $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$ as above, the property $\mathcal{P}'_\mathcal{G}$ has a width 3, oblivious deterministic read-twice BP.*

2. *$\mathcal{P}'_\mathcal{G}$ has a CNF in which every clause is of size at most $\Delta + 1$, and each variable appears in less than $2^{\Delta+2}$ clauses.*

3. *If $\Delta(\mathcal{G})$ is bounded by a constant Δ (independent of n), then testing $\mathcal{P}_\mathcal{G}$ can be reduced to testing $\mathcal{P}'_\mathcal{G}$: For every assignment for $\mathcal{P}_\mathcal{G}$, $x \in \{0,1\}^V$, let $y(x) \in \{0,1\}^{V \cup E}$ be an assignment for $\mathcal{P}'_\mathcal{G}$ that is defined by $y_v = x_v$ for every $v \in V$ and $y_{(u,v)} = x_u$ for every $(u,v) \in E$.*

   If $x \in \mathcal{P}_\mathcal{G}$ then $y(x) \in \mathcal{P}'_\mathcal{G}$, and if $dist(x, \mathcal{P}_\mathcal{G}) \geq \epsilon nm$ then $dist(y(x), \mathcal{P}'_\mathcal{G}) \geq \epsilon nm$. Note that as the number of variables for $\mathcal{P}'_\mathcal{G}$ is bounded by $nm + \Delta n(m-1)$, any ϵ-test for $\mathcal{P}'_\mathcal{G}$ yields a $(\Delta+1)\epsilon$-test for $\mathcal{P}_\mathcal{G}$ with the same number of queries. In particular, to prove that $\mathcal{P}'_\mathcal{G}$ is not ϵ-testable, it is enough to prove that $\mathcal{P}_\mathcal{G}$ is not $\frac{\epsilon}{\Delta+1}$-testable.

Proof: For the first item express the property $\mathcal{P}'_\mathcal{G}$ as all assignments satisfying that for every node $v \in V_i$, we have $x_v = x_{(v,w)}$ for every $(v,w) \in E_i$, and $x_v = \bigoplus_{(u,v) \in E_{i-1}} x_{(u,v)}$. Then, reading the node values level by level and the edge values adjacent to each node, the first condition (for every fixed v) can be expressed trivially by a width 3 oblivious read once branching program (that reads x_v and then compares it to all the appropriate $x_{(v,w)}$). The second condition, for every fixed v, can also be expressed by a width 3 oblivious read once branching program. Taking the conjunction of all these conditions for all possible v results in an oblivious width 3 read-twice branching program, as each node value x_v may be read only by its comparison program and its parity verification program (once by each program), and each edge value $x_{v,w}$ may be read only by the comparison program of v and the parity verification program of w.

The second item also follows directly from the explanation above, by expanding each condition defined by a parity function of some variables to its CNF form. The third item was already proven in the statement of the claim. ∎

The properties $\mathcal{P}_\mathcal{G}$ and $\mathcal{P}'_\mathcal{G}$ depend on the exact choice of the collection $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$. The theorem below asserts that there exists $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$ for which $\mathcal{P}_\mathcal{G}$ and $\mathcal{P}'_\mathcal{G}$ are not testable. In Section 4 and Section 5 we provide a proof for the following theorem.

Theorem 3.3 *For $m = \frac{1}{2}n^{1/4}$ there exists a family of graphs $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$ with $\Delta(\mathcal{G}) \leq 7$, for which any algorithm that $5 \cdot 10^{-7}$-tests the property $\mathcal{P}_\mathcal{G}$ as defined above requires $\Omega(n^{1/8})$ queries.*

Note, the total input size (number of variables) in both $\mathcal{P}_\mathcal{G}$ and $\mathcal{P}'_\mathcal{G}$ is $\theta(nm) = \theta(n^{5/4})$.

4 Proof of Theorem 3.3

For $\mathcal{P}_\mathcal{G}$ (and hence $\mathcal{P}'_\mathcal{G}$) not to be efficiently testable, we need a family $G_1, \ldots, G_{m-1}$ of bipartite graphs that have an 'expansion like' property with $\Delta = O(1)$. We will first explicitly state the property needed, and prove that it implies the non-testability result. In Section 5 we will prove that bipartite graphs with this property exist.

Let $G(U, V, E)$ be a bipartite graph, and let $V' \subseteq V$. We denote by $O_G(V')$ the set of vertices $u \in N(V')$ for which $|\{(u, v)|v \in V'\}|$ is odd. Namely, $O_G(V')$ are those vertices in U that have an odd degree in the induced subgraph $G(U, V', E \cap (U \times V'))$. Similarly, for $U' \subseteq U$, $O_G(U')$ is defined.

Definition 4.1 *Let $G(U, V, E)$ be a bipartite graph, and let $\mathcal{F}$ be a family of nonempty sets, each a subset of either V or U. G is said to be $(\alpha, \mathcal{F})$-odd-expanding if for every $V' \in \mathcal{F}$, $|O_G(V')| \geq \min((1 + \alpha)|V'|, \alpha n)$.*

A comment is due – unlike the case with the usual notion of expanders, it is easy to show that there exist no "parity expander" that works for all non-empty subsets of the vertices. To see this, consider vertex subsets as vectors in $(F_2)^n$, and note that going from a vertex subset to the set of its odd neighbors involves a linear transformation over this space; simple linear algebra concludes this argument.

In the rest of the sequel let $V_1, \ldots, V_m$ be disjoint sets of size n. For every $1 \leq i < m$, we let G_i be a bipartite graph with a vertex set $V_i \cup V_{i+1}$. We now define the notion of (parity) propagations. Given a vertex $u \in V_i$ and $v \in V_j$ for some $1 \leq i \leq j \leq m$, a *propagation path* from u to v is a sequence of vertices $v_k \in V_k$ for $i \leq k \leq j$, such that $v_i = u$, $v_j = v$ and for every $i \leq k < j$ the pair (v_k, v_{k+1}) is an edge of G_k. For $1 \leq i \leq m$ and a set S of vertices from $V_1 \cup \ldots \cup V_i$, we define the *propagation* of S to V_i as the set of vertices of V_i that have an *odd* number of distinct propagation paths from members of S. Similarly, for a set $T \subset V_i \cup \ldots \cup V_m$ we define the *backpropagation* of T to V_i as the set of vertices of V_i from which there is an odd number of propagation paths to members of T.

Some intuition about propagations can be given by the following: Consider a set $U \subset V_1$. Its propagation into V_1 is U itself, while its propagation into V_2 is just $O_{G_1}(U)$. Similarly, its propagation into V_{i+1} can be calculated recursively as $O_{G_i}(U_i)$, where U_i denotes the propagation of U into V_i.

More about the meaning of the propagations and backpropagations is given by the following observations. To state them, we consider a member of $\mathcal{P}_\mathcal{G}$, given by a value x_v for every $v \in V$.

Observation 4.2 *For every i let $U_i = \{v \in V_i | x_v = 1\}$. Then U_i is exactly the propagation of U_1 into V_i for every i.*

Proof: By a simple induction using the connection of propagations to the notion of $O_{G_i}(U_i)$. ∎

Observation 4.3 *Let $v \in V$, and let U be the backpropagation of $\{v\}$ into V_1. Then $x_v = \bigoplus_{u \in U} x_u$.*

Proof: By induction on the i such that $v \in V_i$ (using also the fact that the backpropagation of a set S into V_i equals the backpropagation of its backpropagation to V_j for any $i < j$ such that $S \subset V_j \cup \ldots \cup V_m$). ∎

The last observation is easy to expand to (note that the following also holds for an S that has vertices from more than one level):

Observation 4.4 *Let $S \subset V$, and let U be the backpropagation of S into V_1. Then $\bigoplus_{v \in S} x_v = \bigoplus_{u \in U} x_u$.* ∎

For the rest of the sequel let $(4.5^{2.5} \cdot e^{12.5})^{(2/3)} < 51000 \leq C$. Fix $q = 0.01\sqrt{m} = 0.01n^{1/8}$. Let $\mathcal{F} \subseteq 2^V$ for some set V of size n. We denote by $\mathcal{F}^{\leq q}$ the family of all subsets $S \subset V$ for which there exists $T \in \mathcal{F}$ such that $|(S - T) \cup (T - S)| \leq q$. Namely, $\mathcal{F}^{\leq q}$ contains all sets that can be obtained from sets of $\mathcal{F}$ by adding or deleting at most q elements.

Our aim now is to define families $\mathcal{F}_i$ for which we want G_i to be odd-expanding.

Definition 4.5 *Suppose that $\mathcal{G} = G_1, \ldots, G_{m-1}$ is the family of bipartite graphs $G_i = (V_i, V_{i+1}, E_i)$.*

1. *Let X_m be the family of all nonempty subsets of V_m of size at most q. The sets X_i, $1 \leq i \leq m - 1$, are defined inductively (X_i from X_{i+1}) by: Let $Y_i = \{O_{G_i}(V) | V \in X_{i+1}\}$ and let $X_i = Y_i^{\leq q} - \{\emptyset\}$.*

2. *Let Z_1 be the family of all nonempty subsets of V_1 of size at most $n/(10C)$. Each Z_i, $i \geq 2$, is defined inductively to contain all nonempty subsets of V_i of size at most $n/(10C)$, and in addition all nonempty subsets that are the propagations of the members of Z_{i-1} into V_i.*

3. *Finally we set $\mathcal{F}_i = X_{i+1} \cup Z_i$ for every $1 \leq i \leq m-1$.*

Note that by our choice of parameters $|X_m| \leq n^q$ and hence $|X_i| \leq n^{q(m+1-i)} \leq n^{qm} \leq 2^{\sqrt{n}}$ for each i. Also, $|Z_i| \leq \binom{n}{(n/(10C))} + |Z_{i-1}|$ and hence $|Z_i| < m\binom{n}{(n/(10C))} = o(e^{n/C})$ for every i, so $|\mathcal{F}_i| < e^{n/C}$ for every i.

In the following discussion, we shall restrict ourselves to an instance of $\mathcal{G}$ where each G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding. The existence of such a setup is proven in Section 5.

Proposition 4.6 *Let $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$ where $G_i = (V_i, V_{i+1}, E_i)$, $i = 1, \ldots, m-1$, and assume that G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding. Then any $(\frac{1}{33C} - o(1))$-testing algorithm for $\mathcal{P}_{\mathcal{G}}$ requires at least $\Omega(n^{1/8})$ many queries (even for a 2-sided error).*

The proof of Proposition 4.6 shall be done using Yao's method [18]. This means that we shall define a distribution P on inputs that satisfy $\mathcal{P}_{\mathcal{G}}$, a distribution N on inputs that are far from satisfying it, and prove that these cannot be distinguished from each other by any deterministic algorithm making $o(n^{1/8})$ queries with an error probability at most $\frac{1}{3}$. This implies that there exists a distribution over the inputs for which every such algorithm errs with probability at least $\frac{1}{6}$ (one takes with probability $\frac{1}{2}$ a random input according to P, and with probability $\frac{1}{2}$ an input according to N), and thus there exists no randomized algorithm making $o(n^{1/8})$ queries that errs with probability less than $\frac{1}{6}$ on all inputs (showing that this implies the same also for algorithms with an error probability bounded by $\frac{1}{3}$ is trivial).

We now define the two distributions, P and N. We first select a uniformly random integer l such that $\frac{1}{3}m \leq l \leq \frac{2}{3}m$. We define P to consist of choosing a random member of $\mathcal{P}_{\mathcal{G}}$ – we choose independently uniformly and randomly the values of $\{x_u | u \in V_1\}$ and then extend them to an assignment to all variables as per the calculation depicted by $\mathcal{G}$. Although l was not used in choosing the input according to P, we shall use it in the analysis that compares P to N – the latter uses the random choice of l in choosing its input.

For defining N, we choose two inputs $\mathcal{A}$ and $\mathcal{A}'$ that satisfy the property. $\mathcal{A}$ is chosen according to P. To choose $\mathcal{A}'$ we take an arbitrary member z of $\{x_u | u \in V_1\}$, and invert its value with respect to its value in $\mathcal{A}$, keeping all other values of $\{x_u | u \in V_1\}$. The rest of $\mathcal{A}'$ is again the extension of these values. To choose an input $\mathcal{B}$ according to N, we first choose l, $\mathcal{A}$ and $\mathcal{A}'$ as above, and let $\mathcal{B}$ be identical to $\mathcal{A}$ for any value x_v, $v \in V_1 \cup \ldots \cup V_l$, and identical to $\mathcal{A}'$ for any value x_w, $w \in V_{l+1} \cup \ldots \cup V_m$. We next prove that an input chosen according to N is indeed far from satisfying $\mathcal{P}_{\mathcal{G}}$.

Claim 4.7 *Any input chosen according to N is $\frac{nm(1-o(1))}{33C}$-far from satisfying $P_{\mathcal{G}}$.*

Proof: Let B be chosen as above by choosing l, z, $\mathcal{A}$ and $\mathcal{A}'$. We first note that $\mathcal{A}'$ differs from $\mathcal{A}$ in all vertices that are in the propagation of z to any V_i. Since we assume that each G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding this means that for each $i \geq \log_{1+C}(n/C)$, the propagation set of z is of size at least n/C. Hence for every $i \geq \Theta(\log n)$, $\mathcal{A}$ differs from $\mathcal{A}'$ in at least n/C places.

We then note that $\mathcal{B}$ is at least $\frac{mn(1-o(1))}{3C}$-far from both $\mathcal{A}$ and $\mathcal{A}'$. This is true as $\mathcal{B}$ is identical to $\mathcal{A}$ for all V_i, $i \leq l$ and hence different from $\mathcal{A}'$ on a number of locations which, by the previous paragraph and the choice of l, totals at least $\frac{m-o(m)}{3} \cdot \frac{n}{C}$. A similar argument goes for the distance to $\mathcal{A}$.

Finally, let $\mathcal{C}$ be the closest input to $\mathcal{B}$ such that $\mathcal{C}$ satisfies $\mathcal{P}_{\mathcal{G}}$. Assume first that $\mathcal{C}$ is identical to $\mathcal{A}$ on V_i for some $i \leq l$. Then $\mathcal{C}$ must be identical to $\mathcal{A}$ on each $j \geq i$. This implies that it differs from $\mathcal{A}'$, and hence from $\mathcal{B}$, in at least n/C places for each V_i, $i > l$, as was shown by the first paragraph. This totals to a distance of at least $\frac{mn(1-o(1))}{3C}$.

Assume then that there is no $i \leq l$ for which $\mathcal{C}$ is identical to $\mathcal{A}$. Let i be the smallest for which $\mathcal{C}$ differs from $\mathcal{A}$ in at most $n/(10C)$ places in V_i. If $i \geq \frac{10m}{33}$ then clearly $\mathcal{C}$ differs from $\mathcal{A}$, and hence from $\mathcal{B}$, in at least $n/(10C)$ places for each $j \leq i$. This would total a distance between $\mathcal{C}$ and $\mathcal{B}$ of at least $\frac{n}{10C} \cdot \frac{10m}{33} = \frac{nm}{33C}$ which proves the claim. Assume then that $i < \frac{10m}{33}$ and let S_i be the set of places in V_i where $\mathcal{C}$ differs from $\mathcal{A}$. Then $\mathcal{C}$ differs from $\mathcal{A}$ in the propagation set of S_i into V_j for each $j \geq i$. Note also that $S_i \in \mathcal{F}_i$, and that its propagation set into V_j is in $\mathcal{F}_j$ for every $j \geq i$. Hence, by the odd-expansion assumption for each G_j, this propagation set contains at least n/C elements for each $j \geq i + \Theta(\log n)$ and hence $\mathcal{C}$ differs from $\mathcal{A}$, and hence also from $\mathcal{B}$, in at least $\frac{n}{C} \cdot (\frac{m}{3} - \frac{10m(1-o(1))}{33}) = \frac{n}{C}\frac{m-o(m)}{33}$ places, which completes the proof of the claim. ■

Let $q = o(\sqrt{m})$. We now analyze a fixed set of q queries; let us denote the set of queries by $S = \{x_1, \ldots, x_q\}$, and the corresponding levels by $x_1 \in V_{i_1}, \ldots, x_q \in V_{i_q}$. Let D be the event that $|i_j - l| > \sqrt{m}$ for every j; clearly with probability $1 - o(1)$ this event occurs, and we shall prove in the following that conditioned on D, the restrictions of P and N on $x_1, \ldots, x_q$ yield the same distribution.

For every variable y, let U_y be its propagation to V_1. Let v_y be the characteristic vector of U_y, considered as a member of $(F_2)^n$. In order to understand the restriction of P to a set of queries S, we use the following lemma.

Lemma 4.8 *For any set of variables T, and any set $T' \subset T$ such that $\{v_y | y \in T'\}$ is a maximal linearly independent subset of the set of vectors $\{v_y | y \in T\}$, the following occurs.*

The restriction of the distribution P to T' is the uniform distribution on a set of $|T'|$ independent binary variables, and the values chosen for T' completely determine the values chosen for all of T.

To prove the above we need the following well known (folklore) tool.

Lemma 4.9 *Suppose that $x_1, \dots, x_k$ are random variables with the range $\{0, 1\}$, and suppose that for every $\emptyset \neq S \subset [1, \dots, k]$, the probability for $\bigoplus_{i \in S} x_i = 1$ is exactly $\frac{1}{2}$. Then the variables $x_1, \dots, x_k$ are all uniform and independent.* ∎

Proof of Lemma 4.8: Fix a set $T' \subseteq T$ such that $\{v_y | y \in T'\}$ is a maximal linearly independent set of vectors among $\{v_y | y \in T\}$. By Observation 4.4 the value of every y is completely determined by the values of the variables in U_y, and is in fact their parity. For a set $T'' \subseteq T'$ the probability that $\bigoplus_{y \in T''} y = 1$ is $\frac{1}{2}$ because the backpropagation of T'' to the first layer is nonempty. This means that the distribution over T' is uniform and independent, by Lemma 4.9.

On the other hand, because of the maximality of T', for every $y' \in T - T'$ there exists $T'' \subset T'$ such that $v_{y'} = \bigoplus_{y \in T''} v_y$, so the value of y' is determined by the values of the variables in T'. ∎

Using the set of queries S we now define S_u as the set of all queries that belong to levels above V_l (l was chosen while choosing an input according to P or N), and S_d as the set of all queries belonging to V_l or below. We also define S'_u as a maximal subset of S_u for which $\{v_y | y \in S'_u\}$ is independent, and S'_d as a maximal subset of S_d for which $\{v_y | y \in S'_d\}$ is independent.

The following observation will be used in proving that conditioned on D happening (which occurs with high probability), both P and N induce the same distribution on S, assuming that each G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding.

Observation 4.10 *Given that D happens, for every nonempty subset S''_u of S'_u and nonempty subset S''_d of S'_d, the backpropagation of $S''_u \cup S''_d$ to V_1 is of size at least 2.*

Proof: Since $|S| \leq q$, it follows that the backpropagation of $S''_u \cap (\cup_{j \geq i} V_j)$ into V_i is in $\mathcal{F}_i$ for every i for which $S''_u \cap (\cup_{j \geq i} V_j) \neq \emptyset$ (the backpropagation is nonempty by the choice of S'_u). Hence by the fact that G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding it follows that the backpropagation of S''_u to V_l has size $\Theta(n)$. In turn this implies that the backpropagation of $(S''_u \cup S''_d) \cap (\cup_{j \geq i} V_j)$ into i for every $i \leq l$ is of size at least 2 (and in fact $\Theta(n)$); in particular this holds for $i = 1$. ∎

And now for the final claims.

Claim 4.11 *Given that D happened, the distribution induced by P on $S'_u \cup S'_d$ is the uniform independent one, and the values of the members of $S - S'_u \cup S'_d$ are determined by those of $S'_u \cup S'_d$.*

Proof: Observation 4.10 shows that the vectors corresponding to all backpropagations to V_1 of the members of $S'_u \cup S'_d$ are independent, and this set is clearly maximal with respect to this property (remember that S'_u and S'_d were chosen to have this independence property, and be maximal with respect to it). The claim then follows from Lemma 4.8. ∎

Claim 4.12 *Given that D happened, the distribution induced by N on $S'_u \cup S'_d$ is the uniform independent one, and the values of the members of $S - S'_u \cup S'_d$ are determined by those of $S'_u \cup S'_d$ in the same manner as with the distribution P.*

Proof: Because of Observation 4.10, for every nonempty $S''_u \subseteq S'_u$ and $S''_d \subseteq S'_d$, the parity of the values of $S''_u \cup S''_d$ is 1 with probability exactly $\frac{1}{2}$ (remember that the backpropagation to V_1 includes at least one more variable apart from the one that was inverted to construct $\mathcal{A}'$ from $\mathcal{A}$). This, together with the probabilities for the parity of the values of S''_u by themselves and those of S''_d by themselves (implied by the choice of S'_u and S'_d), implies that the induced distribution on $S'_u \cup S'_d$ is uniform and independent by Lemma 4.9.

The values of the members of $S - S'_u \cup S'_d$ have the same dependencies that they have under the distribution P, because none of these dependencies involves members of both S_u and S_d (again by Observation 4.10). ∎

Proof of Proposition 4.6: We let $\mathcal{G}$, q, P and N be as before, and prove that even an adaptive algorithm cannot distinguish between an input chosen according to P and an input chosen according to N.

Let $Q = \{v_1, \dots, v_q\} \subseteq V$ be a set of vertices in the corresponding levels $v_1 \in V_{i_1}, \dots, v_q \in V_{i_q}$. We denote by $D(Q)$ the event that $|i_j - l| > \sqrt{m}$ for every j.

Now let $\mathcal{T}$ be an adaptive algorithm that queries at most q values $x_v, v \in V$. Every leaf in the decision tree that represent $\mathcal{T}$ is labeled by either 'accept' or 'reject'. Let L be the set of all leaves that are labeled by 'accept'. We may assume that $\text{Prob}_P(L) \geq \frac{2}{3}$, as otherwise the algorithm will not correctly identify many inputs chosen according to P. We shall prove that this necessarily implies that $\text{Prob}_N(L) > \frac{1}{3}$, which implies that in this case the algorithm does not correctly identifies many inputs chosen according to N.

Each leaf $z \in L$ is associated with the set of vertices $Q(z)$ that were queried along the way to z and with a set of answers $ans(z) : Q(z) \longrightarrow \{0,1\}$ to those queries. Clearly, $Q(z)$ and $ans(z)$ together uniquely define z. Moreover, the algorithm will reach z if and only if the variables corresponding to the nodes of $Q(z)$ are respectively assigned $ans(z)$, regardless of the assignments made to other variables that are queried in other branches of the decision tree of $\mathcal{T}$.

Let $Var(z)$ be the set of variables associated with the vertices in $Q(z)$ and let $V'(z) \subset Var(z)$ be such that $\{v_y | y \in V'\}$ is a maximal linearly independent subset of the set of vectors $\{v_y | y \in Var(z)\}$. Lemma 4.8 asserts that for positive inputs the answers on $V'(z)$ uniquely determine the answers on $Q(z) - V'(z)$. We may assume that for every $z \in L$ the answers to $Q(z) - V'(z)$ are consistent with the answers on $V'(z)$, as otherwise the leaf z may be changed to 'reject' without reducing the sucess probability of $\mathcal{T}$ under either P or N. Let $dim(z)$ denote the size of $V'(z)$. This implies the following.

Claim 4.13 *For every $z \in L$, $\text{Prob}_P(z) = 2^{-dim(z)}$.*

Proof: Immediate from Lemma 4.8. ∎

Using it we prove the following claim. Note that together with the assumption that $\text{Prob}_P(L) \geq \frac{2}{3}$ it implies that $\text{Prob}_N(L) > \frac{1}{3}$, thus completing the proof of Proposition 4.6.

Claim 4.14 *Every $z \in L$ satisfies that $Prob_N(z) \geq (1-o(1))2^{-dim(z)} = (1-o(1))Prob_P(z)$.*

Proof: Note that $\text{Prob}_N(z) \geq \text{Prob}_N(z \wedge D(Q(z))) = \text{Prob}_N(D(Q(z)))\text{Prob}_N(z|D(Q(z))) \geq (1-o(1))2^{-dim(z)}$. The last inequality follows from $\text{Prob}_N(z|D(Q(z))) = 2^{-dim(z)}$, which follows from Claim 4.12. ∎

5 The existence of an odd-expanding circuit

We now prove that there exists $\mathcal{G} = \{G_1, \ldots, G_{m-1}\}$ such that every G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding for $\mathcal{F}_i$ as per Definition 4.5, which concludes the proof of Theorem 3.3 (by satisfying the conditions of Proposition 4.6) and hence of the lower bound. First we need the following lemma.

Lemma 5.1 *Let $(4.5^{2.5} \cdot e^{12.5})^{(2/3)} < 51000 \leq C$, let $|U| = |V| = n$, and suppose that $\mathcal{F}$ is a family of at most $e^{n/C}$ nonempty subsets of U. Then $G(U \cup V, E)$, where E is the union of 7 random perfect matchings between U and V, is $(\frac{1}{C}, \mathcal{F})$-odd-expanding with probability at least $(1 - n^{-3/2})$.*

Note: G may have parallel edges with constant probability.

Proof of Lemma 5.1: To prove the odd-expansion properties with regards to a fixed family $\mathcal{F}$ we analyze the odd-expansion of subsets $S \subseteq U$ according to their sizes. We first show that with very high probability G will be odd-expanding for all small enough sets, using the standard expansion property of random graphs. Only for larger sets we need the bound on the size of $\mathcal{F}$.

For a set S of size k, $k \leq n/C$, let us look at the 7 random sets that are the images of S under the 7 random matchings. Each such image $M_i(S)$ is a random set of size k and is independent of the others. We first want to bound the probability that the union of these sets is small: Let S be fixed and let $T \subseteq V$ be of size $4.5k$. Using the inequalities $(\frac{n}{k})^k \leq \binom{n}{k} \leq (\frac{ne}{k})^k$ we get:

$$Prob[\cup_{i=1}^{7} M_i \subseteq T] \leq \frac{\binom{4.5k}{k}^7}{\binom{n}{k}^7} \leq \frac{(4.5e)^{7k}}{(\frac{n}{k})^{7k}} = (\frac{4.5ek}{n})^{7k}.$$

Hence, by summing for all possible $k \leq n/C$ and for all possible sets S and T we get:

$$Prob\left[\exists S, |S| = k \leq n/C, |\cup_{i=1}^{7} M_i| \leq 4.5k\right] \leq \Sigma_{k=1}^{n/C} \binom{n}{k} \binom{n}{4.5k} (\frac{4.5ek}{n})^{7k} \leq \Sigma_{k=1}^{n/C} (\frac{ne}{k})^k (\frac{ne}{4.5k})^{4.5k} (\frac{4.5ek}{n})^{7k} = \Sigma_{k=1}^{n/C} (\frac{4.5^{2.5} \cdot e^{12.5} \cdot k^{1.5}}{n^{1.5}})^k \leq \frac{4.5^{2.5} \cdot e^{12.5}}{n^{1.5}} + \Sigma_{k=2}^{\frac{n}{2eC}} (\frac{4.5^{2.5} \cdot e^{12.5} \cdot k^{1.5}}{n^{1.5}})^k + \Sigma_{k=\frac{n}{2eC}}^{n/C} (\frac{4.5^{2.5} \cdot e^{12.5} \cdot k^{1.5}}{n^{1.5}})^k \leq \frac{4.5^{2.5} \cdot e^{12.5}}{n^{1.5}} + \Sigma_{k=2}^{\frac{n}{2eC}} \frac{4.5^{2.5} \cdot e^{12.5} \cdot 8}{n^{1.5}} 2^{-k+2} + n(\frac{4.5^{2.5} \cdot e^{12.5}}{C^{1.5}})^{n/2eC} = O(n^{-3/2}).$$

The last bound on the summands is by our choice of C. It guarantees that the first series is dominated by a geometric series while in the second series each summand is at most $(\frac{4.5^{2.5} \cdot e^{12.5}}{C^{1.5}})^{n/2eC}$, which is exponentially small in n.

Now note (by looking at the degrees of vertices in the graph induced by $S \cup N(S)$) that if $|N(S)| \geq 4.5|S|$ then $|O_G(S)| \geq 2|S|$, which is more than what we need.

Now let $S \subseteq V$ be of size k, $n/C \leq k \leq n/2$. Fix $i \in \{1, \ldots, 7\}$ and let $T_j = V - M_j$ for $j \neq i$. Let $P_i = M_i \cap (\cap_{j \neq i} T_j)$, namely P_i contains all elements that are in the ith image of S but not in any $M_j, j \neq i$. Certainly $\cup_i P_i \subseteq O_G(S)$.

To estimate the size of $\cup_i P_i$, we simulate the process of choosing a random matching of S by marking each element of V independently, with probability k/n. We then repeat it independently 7 times. Finally, we condition on the event that M_i, the set marked in the i'th round, is of size exactly k for every i, and pick a random 1-1 mapping between S and each of the 7 marked sets. Let B_i be the event that $|M_i| = k$. Then $Prob(B_i) \geq \Omega(1/\sqrt{n})$ (as $|M_i|$ is binomially distributed, and then using the Chebyshev inequality [1]). Note also that in the random model where we do not condition on the events $B_r, r = 1, \ldots, 7$, the expected size of $\cup P_i$ is $7k \cdot (1 - \frac{k}{n})^6 \geq \frac{6.9n}{C}$ by our choice of k. Also, as this size is a sum of independent random Boolean variables we may apply a Chernoff bound on the probability of the deviation from the expectation being large. Hence we get:

$Prob\left[|\cup_i P_i| \leq \frac{n}{C}| \wedge_{r=1}^{7} B_r\right] \leq$
$Prob[|\cup_i P_i| \leq \frac{n}{C}] \cdot (Prob[\wedge_{r=1}^{7} B_r])^{-1} \leq$
$exp(-\frac{2.48n}{C}) \cdot n^{3.5} \leq \frac{1}{n^2} exp(-2.4n/C)$.

Hence with probability of at least $1 - \frac{1}{n^2} \cdot e^{-\frac{2.4n}{C}}$ the union of the P_i's has size at least $\frac{n}{C}$, which implies that $|O_G(S)| \geq \frac{n}{C}$ as required. Thus for any fixed family $\mathcal{F}$ of at most $e^{2.4n/C}$ sets of sizes of at most $n/2$ we have that G is $(1/C, \mathcal{F})$-odd-expanding with probability at least $1 - n^{-2}$.

Finally, for larger sets: Let $S \subseteq V$ with $|S| = k \geq n/2$. Let T be the set of all elements of V that are the images of all M_i, namely $T = \cap M_i(S)$. Certainly $T \subseteq O_G(S)$. A similar analysis to the above shows that the expected value of $|T|$ is $n(\frac{k}{n})^7 \geq \frac{n}{128}$. Hence with probability at most $exp(-\frac{9}{16} \cdot \frac{n}{2 \cdot 128}) \leq exp(-n/C)$ we may get $|T| \leq n/C$. Again this makes sure that with probability $1 - n^{-2}$ the graph G is $(1/C, \mathcal{F})$-odd-expanding for every fixed $\mathcal{F}$ containing at most $2^{n/C}$ sets of sizes above $n/2$ each. This, together with the previous cases, completes the proof of the lemma (the argument for subsets of V is identical to that for subsets of U). ∎

Lemma 5.2 *Suppose that $\mathcal{G} = G_1, \ldots, G_{m-1}$ is a family of graphs in which each G_i is constructed randomly and independently as was done in Lemma 5.1. Suppose also that for every i, $\mathcal{F}_i$ is defined as in Definition 4.5. Then with a positive probability every G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding.*

Proof: For every $1 \leq i \leq m$ let H_i be the event that G_i is $(\frac{1}{C}, \mathcal{F}_i)$-odd-expanding. Note that $\mathcal{F}_i$ is dependent only on $\{G_j | j \neq i\}$ and not on G_i itself, which is chosen independently of them. Also note that $|\mathcal{F}_i| \leq e^{n/C}$. Thus Lemma 5.1 bounds from below the probability of H_i happening by $1 - n^{-3/2}$. In particular, with positive probability all the events H_i occur. ∎

References

[1] N. Alon and J. H. Spencer, The probabilistic method, *John Wiley & Sons, Inc.* 1991.

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

[3] 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.

[4] 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.

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

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

[7] 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.

[8] E. Fischer, On the strength of comparisons in property testing, manuscript (available as ECCC TR00-083).

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

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

[11] E. Fischer, E. Lehman, I. Newman, S. Raskhodnikova, R. Rubinfeld and A. Samorodnitsky, Monotonicity testing over general poset domains, manuscript.

[12] 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.

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

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

[16] 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, 2001.

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

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