Limitation of the BLR testing in estimating nonlinearity (Extended Abstract)

Debajyoti Bera[1], Subhamoy Maitra[2], Dibyendu Roy[2], and Pantelimon Stănică[3]

[1] Indraprastha Institute of Information Technology, Delhi, India
[2] Indian Statistical Institute, Kolkata, India
[3] Naval Postgraduate School, Monterey, USA
dbera@iiitd.ac.in, subho@isical.ac.in, roydibyendu.rd@gmail.com, pstanica@nps.edu

Abstract. In this paper we concentrate on a limitation of the BLR (Blum-Luby-Rubinfeld) linearity test on a Boolean function f, which checks the weight of the function $F(x,y) = f(x)+f(y)+f(x+y)$ for many random inputs x, y, which should be 0 for a linear function. We point out that this test, which considers the weight of F and $G(x,y) = g(x) + g(y) + g(x+y)$, does not provide proper information on the nonlinearity ordering of two functions f, g. The problem remains even if we extend the algorithm to analyze the nonlinearities of F, G, too. In this direction, we provide examples of functions on any number of input variables, such that the BLR test (and certain extensions of it) fails to estimate the nonlinearity hierarchy property. We also consider a quantum algorithm to demonstrate the problem in maintaining such hierarchy.

Keywords: Boolean functions, BLR linearity testing, Nonlinearity.

1 Introduction

Nonlinearity is one of the most important properties for Boolean functions from the aspect of cryptology (since it provides resistance against linear attack), coding theory (covering radius of Reed-Muller codes), and combinatorics. To calculate the nonlinearity for an n-variable Boolean function, one requires $O(n2^n)$ time complexity using the fast Walsh transform. This complexity being exponential, there are efforts to estimate the nonlinearity of a Boolean function with fewer runs, both in classical and quantum domain. Despite having a lot of progress in deciding whether a Boolean function is linear or not by making a few queries to the function, to the best of our knowledge there has not been much headway into estimating the "nonlinearity" of the function. Nonlinearity of a function is defined roughly as the smallest distance of the function to any affine function. This paper provides evidence that the BLR linearity test (described below) and its extensions in classical and quantum domain fail to do so.

Our initial findings suggest that further techniques may be needed since existing (probabilistic) techniques in classical domains [1,4,7,10], do not really "preserve" the nonlinearity hierarchy. We measure the cost of estimation as the number of queries (called as query complexity) made to the Boolean function, say $f : \{0,1\}^n \to \{0,1\}$, that is given to us as a black-box and one can obtain $f(x)$ giving an input x. Since we can modify a linear function by changing the output at a single input, deterministic linearity detection requires querying f on all of the 2^n input points. Thus, any sub-exponential query algorithm, say $\mathcal{A}$, is bound to make an error.

For the BLR algorithm, $wt(F(x,y))$ plays an important role, where $F(x,y) = f(x) + f(y) + f(x+y)$. For all the known tests [1,4,7,10], $wt(F) = 0$ whenever $nl(f) = 0$, i.e., f is linear[4]. The difficulty of using these tests for nonlinearity estimation is that $wt(F)$ is *not monotonically increasing* with $nl(f)$. One of the objectives of this paper is to highlight that there exists a pair of functions, say f_1 and f_2 for every input-length (greater than or equal to 4) for which $nl(f_1) > nl(f_2)$ but $wt(F_1) < wt(F_2)$ and $nl(F_1) < nl(F_2)$, for the classical BLR test and a quantum linearity test as well. Classical linearity tests are mostly variations and extensions of BLR, so we explain our observation with respect to BLR. Quantum linearity testing algorithms, on the other hand, are mostly based on estimating Walsh coefficients being done using the single-query Deutsch-Jozsa algorithm and we use such a quantum algorithm to highlight the difficulty mentioned above.

2 Background

A Boolean function on n variables is a map from the n-dimensional vector space of all binary tuples $\mathbb{F}_2^n = \{0,1\}^n$ into the two-element field $\mathbb{F}_2 = \{0,1\}$. We will denote the set of n-variable Boolean functions as $\mathcal{B}_n$, with $|\mathcal{B}_n| = 2^{2^n}$.

An n-variable Boolean function f can be considered to be a multivariate polynomial over $\mathbb{F}_2$. More precisely, f can be written as $f(x_1,\ldots,x_n) = a_0 + \sum_{1\le i\le n} a_i x_i + \sum_{1\le i<j\le n} a_{ij} x_i x_j + \cdots + a_{12\ldots n} x_1 x_2 \ldots x_n$, where the coefficients are in $\{0,1\}$. This representation is called the *algebraic normal form* (ANF) of f. The number of variables in the highest order product term with nonzero coefficient is called the *algebraic degree*, or simply the degree of f and denoted by $deg(f)$.

Functions of degree at most one are called *affine* functions. An affine function with constant term equal to zero is called a *linear* function. The set of all n-variable affine functions is denoted by $\mathcal{A}_n$. That is, the set of affine functions contains all the linear functions and their complements.

For $x = (x_1,\ldots,x_n)$ and $\omega = (\omega_1,\ldots,\omega_n)$ both in $\mathbb{F}_2^n$, we define the inner product by $x \cdot \omega = x_1\omega_1 + \cdots + x_n\omega_n$. A Boolean function $\ell(x)$ is affine if $\ell(x) = \omega \cdot x + \omega_0$ for some fixed $\omega \in \mathbb{F}_2^n, \omega_0 \in \mathbb{F}_2$ (if $\omega_0 = 0$, then it is linear).

[4] Without loss of generality, throughout the paper, we consider the functions where $f(0,0,\ldots,0) = 0$. The analysis may be changed accordingly when $f(0,0,\ldots,0) = 1$.

For $f \in \mathcal{B}_n$, the *Walsh transform* of $f(x)$ is an integer valued function over $\mathbb{F}_2^n$, defined as $\widehat{f}(\omega) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+x\cdot\omega}$.

The fastest known classical algorithm to calculate all the Walsh spectrum values of $f \in \mathcal{B}_n$, i.e., $\widehat{f}(\omega)$ at each of the 2^n points ω, is of $O(n2^n)$ time complexity. Calculation of the Walsh spectrum value at a specific point requires $O(2^n)$ time in classical domain. The nonlinearity of an n-variable function f is $nl(f) = \min_{g \in \mathcal{A}_n} (d(f,g))$, i.e., the minimum distance from the set of all n-variable affine functions. In terms of Walsh spectrum, the nonlinearity of f is given by $nl(f) = 2^{n-1} - \frac{1}{2} \max_{\omega \in \mathbb{F}_2^n} |\widehat{f}(\omega)|$.

The (Hamming) weight wt of a vector is the number of 1's in the vector. The weight of a Boolean functions is the weight of its truth table (output values). If $wt(f) = 2^{n-1}$, then f is called a balanced function. In terms of the Walsh coefficients, $f \in \mathcal{B}_n$ is balanced if and only if $\widehat{f}(0,\ldots,0) = 0$. The distance between two vectors is the weight of their sum, that is, $d(u,v) = wt(u+v)$.

3 Linearity testing

Testing whether a Boolean function (given as an oracle) is affine or not is an important question in the field of computational complexity [4,3]. For further results in this area of property testing, one may refer to [2,8,9].

Definition 1. *Given two n-variable Boolean functions f and g, we say that f, g are ϵ-far if $\frac{|\{x \in \mathbb{F}_2^n : f(x) \neq g(x)\}|}{2^n} = \frac{d(f,g)}{2^n} > \epsilon$, Further, an n-variable Boolean function f will be called ϵ-far from a subset S of n-variable Boolean functions if f is ϵ-far from all the functions $g \in S$. Naturally, the definition of ϵ-close is just the opposite, i.e., f, g are ϵ-close if $\frac{|\{x \in \mathbb{F}_2^n : f(x) \neq g(x)\}|}{2^n} = \frac{d(f,g)}{2^n} \leq \epsilon$.*

The classical probabilistic test for linearity is well known as the BLR (Blum-Luby-Rubinfeld) test [4] that exploits the property of a linear function $\ell(x+y) = \ell(x) + \ell(y)$, $\forall x, y \in \mathbb{F}_2^n$, with $\ell(0) = 0$. When the testing fails, we can conclude (with some probability) that the function in consideration is not linear.

Below, we provide the detail description of the BLR linearity testing. The prime goal of this test is to test the linearity of a function without considering all possible inputs of the Boolean function. If the function $f \in \mathcal{B}_n$ is approximately a linear function, then $f(x+y) = f(x) + f(y)$ for many random $x, y \in \mathbb{F}_2^n$. To be more precise, let $F : \mathbb{F}_2^{2n} \to \mathbb{F}_2$ be such that $F(x,y) = f(x) + f(y) + f(x+y)$, where $x, y \in \mathbb{F}_2^n$. If $f(x+y) = f(x) + f(y)$ for many random x, y, then $F(x,y)$ will be zero for randomly many inputs of F. Hence, $wt(F)$ should reflect the linearity of f.

The BLR algorithm follows a few simple steps and returns whether the function is approximately linear or not:

1. *Select random x, y.*
2. *Check the satisfiability of $f(x+y) = f(x) + f(y)$.*

3. *If item 2 is satisfied for many randomly chosen x, y, then the function f is probably linear.*

As the algorithm is based on random sampling, it is a probabilistic test. We assume that p is the probability that BLR is successful in the linearity testing of f. Further, one is interested in checking the relation between this probability p and function f. This relation can be obtained from the existing proof of the BLR linearity test. The result states that, if the function is ϵ-close (then $p = 1 - \epsilon$), then $\max_{a \in \mathbb{F}_2^n} \widehat{f}(a) \geq 1 - 2\epsilon$.

4 Limitation of BLR test in nonlinearity hierarchy

It is natural to wonder whether one can get some estimates on the nonlinearity of a function f given the weight (or even the nonlinearity of the corresponding function F), by running the BLR test (or modifications of it) and attempt to provide a range $[nl(f) - a, nl(f) + b]$ in which the nonlinearity of f lies with certain probability. It is the purpose of this section to give further proof that this is not quite realizable with just the BLR test (or slight modifications of it).

It is clear that the BLR test is a probabilistic algorithm and errors may occur. However, we like to point out a major problem here. Even in case we consider all the inputs, the weight $wt(F)$, where $F = f(x) + f(y) + f(x + y)$ does not provide a correct picture. Consider two 4-variable functions f_1, f_2 such that $3 = nl(f_1) < nl(f_2) = 5$ (we computed and there are 560 choices for f_1 and 448 choices for f_2). We chose randomly two such functions with truth tables $f_1 = 0000000100011000$, $f_2 = 0000000100010111$. However, $114 = wt(F_1) > wt(F_2) = 90$ and $98 = nl(F_1) > nl(F_2) = 90$. Thus, the BLR algorithm, by itself, cannot take care of the actual distance even when calculated over all the inputs.

One may wonder if by taking all Walsh transforms values of F, perhaps we can provide some improvement in the testing. Calculating the Walsh transform value of F at a point $\omega = (\alpha, \beta)$ amounts to calculating the weight of the function $F_{\alpha,\beta}(x, y) = F(x, y) + L_{\alpha,\beta}(x, y) = f(x) + f(y) + f(x + y) + L_{\alpha,\beta}(x, y)$, where $L_{\alpha,\beta}(x, y) = (\alpha, \beta) \cdot (x, y) = \alpha \cdot x + \beta \cdot y$, for some fixed α, β (when there is no danger of confusion, and α, β are fixed, we shall use L in lieu of $L_{\alpha,\beta}$). We can use $F_{\alpha,\beta}$ in the BLR linearity testing with the same probability of success. The process is described below:

Extended (α, β)-BLR linearity testing:

1. *Choose random x, y.*
2. *Query $F_{\alpha,\beta}(x, y) = f(x) + f(y) + f(x + y) + L(x, y)$, for many randomly chosen x, y.*
3. *If $F_{\alpha,\beta}(x, y) = 0$ for many randomly chosen x, y, then we say that f is approximately linear.*

We assume that the above is successful with probability $p = Pr[f(x)+f(y)+L(x, y) = f(x+y)]$. If the function f is ϵ-close to L, then we will have $1-\epsilon = p =$

$Pr[\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y) = \mathbf{f}(x+y)]$, where $\mathbf{f}(x) = (-1)^{f(x)}$ and $\mathbf{L}(x,y) = (-1)^{L(x,y)}$.

Theorem 2. *For $f \in \mathcal{B}_n$ and $\alpha, \beta \in \mathbb{F}_2^n$ (writing L for $L_{\alpha,\beta}$), we have*

$$Pr[f(x)+f(y)+L(x,y) = f(x+y)] = E\left[\frac{1}{2} + \frac{1}{2}\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y)\mathbf{f}(x+y)\right]$$
$$= \sum_{c\in\mathbb{F}_2^n} \widehat{f}(c)\widehat{f}(c+\alpha)\widehat{f}(c+\beta) \leq \max_{c\in\mathbb{F}_2^n} \widehat{f}(c).$$

Proof. First, $\frac{1}{2} + \frac{1}{2}\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y)\mathbf{f}(x+y) = 1$, if $\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y) = \mathbf{f}(x+y)$, otherwise it is 0. Now, $p = Pr[f(x)+f(y)+L(x,y) = f(x+y)]$ becomes

$$p = E\left[\frac{1}{2} + \frac{1}{2}\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y)\mathbf{f}(x+y)\right] = \frac{1}{2} + \frac{1}{2}E\left[\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y)\mathbf{f}(x+y)\right]. \quad (1)$$

Next, we compute

$$E\left[\mathbf{f}(x)\mathbf{f}(y)\mathbf{L}(x,y)\mathbf{f}(x+y)\right] = E\left[\sum_{a,b,c\in\mathbb{F}_2^n} \widehat{f}(a)\widehat{f}(b)\widehat{f}(c)\lambda_a(x)\lambda_b(y)\lambda_c(x+y)\lambda_\alpha(x)\lambda_\beta(y)\right]$$
$$= \sum_{a,b,c\in\mathbb{F}_2^n} \widehat{f}(a)\widehat{f}(b)\widehat{f}(c)E\left[\lambda_{a+c+\alpha}(x)\right]E\left[\lambda_{b+c+\beta}(y)\right]$$
$$= \sum_{c\in\mathbb{F}_2^n} \widehat{f}(c)\widehat{f}(c+\alpha)\widehat{f}(c+\beta) \leq \left|\sum_{c\in\mathbb{F}_2^n} \widehat{f}(c)\widehat{f}(c+\alpha)\widehat{f}(c+\beta)\right|$$
$$\leq \left(\sum_{c\in\mathbb{F}_2^n}\left(\widehat{f}^2(c)\widehat{f}^2(c+\alpha)\right)\right)^{\frac{1}{2}}\left(\sum_{c\in\mathbb{F}_2^n}\left(\widehat{f}^2(c+\beta)\right)\right)^{\frac{1}{2}} \text{ (by Cauchy–Schwarz's inequality)}$$
$$\leq \max_{c\in\mathbb{F}_2^n}\widehat{f}(c)\left(\sum_{c\in\mathbb{F}_2^n}\left(\widehat{f}^2(c+\alpha)\right)\right)^{\frac{1}{2}}\left(\sum_{c\in\mathbb{F}_2^n}\left(\widehat{f}^2(c+\beta)\right)\right)^{\frac{1}{2}} = \max_{c\in\mathbb{F}_2^n}\widehat{f}(c),$$

since $E[\lambda_{a+c+\alpha}(x)] = 1$, if $a = c+\alpha$, and 0, otherwise, as well as, $E[\lambda_{b+c+\beta}(y)] = 1$, if $b = c+\beta$, and 0, otherwise. □

Corollary 3. *Let $f \in \mathcal{B}_n, \alpha, \beta \in \mathbb{F}_2^n$, $L(x,y) = (\alpha,\beta)\cdot(x,y)$, and $Pr[f(x)+f(y)+L(x,y) = f(x+y)] = 1-\epsilon$. Then, we have $\frac{1}{2} + \frac{1}{2}\max_{c\in\mathbb{F}_2^n}\widehat{f}(c) \geq 1-\epsilon$, that is, $\max_{c\in\mathbb{F}_2^n}\widehat{f}(c) \geq 1-2\epsilon$, similarly as in the BLR linearity test.*

Computational data revealed (as we pointed out in the beginning of Section 4) that the BLR and the Extended (α,β)-BLR test provide incorrect estimates on the nonlinearity even if we consider all possible inputs on the function f.

4.1 Limitation of BLR test on Boolean functions involving higher number of variables

In this section we prove the existence of functions in $n \geq 4$ number of variables on which BLR test does not preserve nonlinearity hierarchy. We start with a result that gives the convolution of the functions involved in $F_{\alpha,\beta}$, which will be useful to us to achieve our goal.

Theorem 4. *Let $f, F_{\alpha,\beta}$ be Boolean functions in n, respectively, $2n$ variables such that $F_{\alpha,\beta}(x,y) = f(x) + f(y) + f(x+y) + \alpha x + \beta y$, where α, β are random but fixed n-length vectors. Then $\widehat{F}_{\alpha,\beta}(x,y) = 2^{-n} \sum_{v \in \mathbb{F}_2^n} \widehat{f}(v)\widehat{f}(\alpha+x+v)\widehat{f}(\beta+y+v)$.*

Proof. For $x, y \in \mathbb{F}_2^n \times \mathbb{F}_2^n$, denote $f_1(x,y) = f(x), f_2(x,y) = f(y), f_3(x,y) = f(x+y)$, $f_4(x,y) = \alpha x + \beta y$. We shall be using throughout the following relation from [6, Thm. 2.17] for a Boolean sum $g+h$: $\widehat{h+g}(x) = 2^{-n} \sum_{u \in \mathbb{F}_2^n} \widehat{g}(u)\widehat{h}(u+x)$, as well as the fact that $\widehat{f}(x,y) = \widehat{g}(x)\widehat{h}(y)$, if $f(x,y) = g(x) + h(y)$. For $F := F_{\alpha,\beta}$, this becomes

$$\begin{aligned}\widehat{F}(x,y) &= 2^{-2n} \sum_{(u_1,u_2) \in \mathbb{F}_2^{2n}} \widehat{f_1+f_2}(u_1,u_2)\widehat{f_3+f_4}(x+u_1, y+u_2) \\ &= 2^{-4n} \sum_{(u_1,u_2),(v_1,v_2) \in \mathbb{F}_2^{2n}} \widehat{f}(u_1)\widehat{f}(u_2)\widehat{f}_3(v_1,v_2)\widehat{f}_4(x+u_1+v_1, y+u_2+v_2).\end{aligned}$$

Now, using [6, Lemma 2.9] and the Kronecker function $\delta_0(w) = 1$, respectively 0, if $w = 0$, respectively, $w \neq 0$, we compute

$$\begin{aligned}\widehat{f}_3(v_1,v_2) &= \sum_{(w_1,w_2) \in \mathbb{F}_2^n} (-1)^{f(w_1+w_2)+v_1w_1+v_2w_2} \\ &= \sum_{(w_1,z) \in \mathbb{F}_2^n} (-1)^{f(z)+v_1w_1+v_2w_1+v_2z} \\ &= \sum_{z \in \mathbb{F}_2^n} (-1)^{f(z)+v_2z} \sum_{w_1 \in \mathbb{F}_2^n} (-1)^{w_1(v_1+v_2)} = 2^n \widehat{f}(v_2)\delta_0(v_1+v_2),\end{aligned}$$

so, $\widehat{f}_3(v_1,v_2) = 0$ unless $v_1 = v_2$, when $\widehat{f}_3(v_1,v_2) = 2^n\widehat{f}(v_2)$. Further, under $v_1 = v_2$, a similar analysis will render

$$\widehat{f}_4(x+u_1+v_1, y+u_2+v_1) = 2^{2n}\delta_0(\alpha+x+u_1+v_1)\delta_0(\beta+y+u_2+v_1).$$

Thus, the expression is 0, unless $u_1 = \alpha + x + v_1, u_2 = \beta + y + v_1$. Putting all these together, we obtain $\widehat{F}(x,y) = 2^{-n} \sum_{v_1 \in \mathbb{F}_2^n} \widehat{f}(v_1)\widehat{f}(\alpha+x+v_1)\widehat{f}(\beta+y+v_1)$, and the theorem is shown. □

The following lemma is known and easy to show.

Lemma 5. *The Walsh transform of the concatenation $f' = f\|\bar{f}$, that is, $f'(u, u_n) = f(u)+u_n$, where $u = (u_0, \ldots, u_{n-1})$ is given by $\widehat{f'}(u, u_n) = \widehat{f}(u)\left(1 + (-1)^{u_n+1}\right)$.*

For easy referral, we define the following properties on two Boolean functions f_1, f_2 in n variables and their corresponding $2n$-variables BLR relevant functions F_1, F_2 with $F_i(x, y) = f_i(x) + f_i(y) + f_i(x + y)$:

$$(\mathcal{P}_1): \quad nl(f_1) < nl(f_2) \text{ and } wt(F_1) > wt(F_2);$$
$$(\mathcal{P}_2): \quad nl(f_1) < nl(f_2) \text{ and } nl(F_1) > nl(F_2).$$

Our proof below considers only the two properties above, but it is easy to construct examples for any number of variables where the inequalities are reversed (hence our claim that the weight/nonlinearities hierarchy is not preserved) by starting the inductive procedure with a pair of functions satisfying whatever inequality we wish. For example, taking $f_1 = 0000000000001110$, $f_2 = 0000000000001111$, then $nl(f_1) = 3 < nl(f_2) = 4$ and $nl(F_1) = 90 < nl(F_2) = 96$, we then apply the method in the proof below.

Theorem 6. *For any value of n, there exist two functions f_1 and f_2 on which the BLR test do not preserve the weight and nonlinearity hierarchy, namely the properties $\mathcal{P}_1$ and $\mathcal{P}_2$ hold.*

Proof. From the discussion of Section 4 we know that there exist two functions $f_1^{(4)}$, $f_2^{(4)}$ on 4 variables satisfying $\mathcal{P}_1$. We further consider $F_1^{(4,4)}(x, y) = f_1^{(4)}(x) + f_1^{(4)}(y) + f_1^{(4)}(x + y)$ and $F_2^{(4,4)}(x, y) = f_2^{(4)}(x) + f_2^{(4)}(y) + f_2^{(4)}(x + y)$. As discussed, $nl(f_1^{(4)}) < nl(f_2^{(4)})$ but $wt(F_1^{(4,4)}) > wt(F_2^{(4,4)})$. We use an inductive method to show the existence of such functions involving n variables. We define two functions $f_1^{(5)}$, $f_2^{(5)}$ involving 5 variables, such that $f_1^{(5)}(x_0, \ldots, x_4) = f_1^{(4)}(x_0, \ldots, x_3) + x_4$ and $f_2^{(5)}(x_0, \ldots, x_4) = f_2^{(4)}(x_0, \ldots, x_3) + x_4$. It can be observed that $nl(f_1^{(5)})$ and $nl(f_2^{(5)})$ will follow the same relation, i.e., $nl(f_1^{(5)}) < nl(f_2^{(5)})$. We further note that

$$\begin{aligned}
&F_1^{(5,5)}(x_0, \ldots, x_4, y_0, \ldots, y_4) \\
&= f_1^{(5)}(x_0, \ldots, x_4) + f_1^{(5)}(y_0, \ldots, y_4) + f_1^{(5)}(x_0 + y_0, \ldots, x_4 + y_4) \\
&= f_1^{(4)}(x_0, \ldots, x_3) + x_4 + f_1^{(4)}(y_0, \ldots, y_3) + y_4 + f_1^{(4)}(x_0 + y_0, \ldots, x_3 + y_3) + x_4 + y_4 \\
&= F_1^{(4,4)}(x_0, \ldots, x_3, y_0, \ldots, y_3).
\end{aligned}$$

Similarly $F_2^{(5,5)}(x_0, \ldots, x_4, y_0, \ldots, y_4) = F_2^{(4,4)}(x_0, \ldots, x_3, y_0, \ldots, y_3)$. Hence the weight of $F_1^{(5,5)}$ and $F_2^{(5,5)}$ will also satisfy $wt(F_1^{(5,5)}) > wt(F_2^{(5,5)})$. So from our two functions involving 4 variables (see Section 4) we can construct two new functions involving 5 variables on which the BLR test satisfies $\mathcal{P}_1$.

In general, if we have two functions $f_1^{(n)}$ and $f_2^{(n)}$ on n number of variables which follow the same relation as in Section 4 $\left(nl(f_1^{(n)}) < nl(f_2^{(n)})\right.$ but $\left.wt(F_1^{(n,n)}) > wt(F_2^{(n,n)})\right)$, we can construct two functions $f_1^{(n+1)}$, $f_2^{(n+1)}$ on $n+1$ number of variables on which BLR test fails to predict the correct nonlinearity

relation, that is $\mathcal{P}_1$ is satisfied. Here $f_1^{(n+1)}(x_0,\ldots,x_n) = f_1^{(n)}(x_0,\ldots,x_{n-1}) + x_n$ and $f_2^{(n+1)}(x_0,\ldots,x_n) = f_2^{(n)}(x_0,\ldots,x_{n-1}) + x_n$. Hence the first claim of our theorem follows.

Next, for the same concatenation construction generating $f_1^{(n+1)}, f_2^{(n+1)}$ in $n+1$ variables from $f_1^{(n)}, f_2^{(n)}$, and computing the Walsh coefficients of the corresponding $F_1^{(n+1)}, F_2^{(n+1)}$, by using the convolution Theorem 4 and Lemma 5 we obtain (we let $x = (x_0,\ldots,x_{n-1})$, $y = (y_0,\ldots,y_{n-1})$, $v = (v_0,\ldots,v_{n-1})$)

$$\begin{aligned}\widehat{F}_i(x,x_n,y,y_n) &= 2^{-(n+1)} \sum_{(v,v_n)\in\mathbb{F}_2^{n+1}} \widehat{f}_i(v)\left(1+(-1)^{v_n+1}\right) \\ &\qquad \cdot \widehat{f}_i(x+v)\left(1+(-1)^{x_n+v_n+1}\right)\widehat{f}_i(y+v)\left(1+(-1)^{y_n+v_n+1}\right) \\ &= 2^{-n}\sum_{v\in\mathbb{F}_2^n} \widehat{f}(v)\widehat{f}_i(x+v)\left(1+(-1)^{x_n}\right)\widehat{f}_i(y+v)\left(1+(-1)^{y_n}\right).\end{aligned}$$

If $x_n = 1$, or $y_n = 1$, then $\widehat{F}_i(x,x_n,y,y_n) = 0$, otherwise $\widehat{F}_i(x,0,y,0) = 4\times 2^{-n}\sum_{v\in\mathbb{F}_2^n}\widehat{f}_i(v)\widehat{f}_i(x+v)\widehat{f}_i(y+v) = 4\widehat{F}_i(x,y)$, showing that $\max_{(x,x_n,y,y_n)\in\mathbb{F}_2^{n+2}} |\widehat{F}_i(x,x_n,y,y_n)| = 4\max_{(x,y)\in\mathbb{F}_2^n}|\widehat{F}_i(x,y)|$, which will imply $\mathcal{P}_2$. □

5 Nonlinearity hierarchy and limitations of a quantum linearity test

Most quantum algorithms operating on Boolean functions are adaptations of the famous Deutsch-Jozsa (equivalently, 1-level Bernstein-Vazirani) quantum algorithm, that we refer to as DJ [11]. This is due to the amazing property of the DJ algorithm which, when given an input f, the distribution of its observed outputs is exactly (and somewhat magically) $\widehat{f}^2(w)/2^{2n}$ for all $w\in\mathbb{F}_2^n$, that also after making only one quantum query to f.

All the currently known quantum tests for linearity [8,5] are also based upon the DJ algorithm. We show below how to extend the Deutsch-Jozsa circuit to obtain a linearity-testing circuit using 2 queries that we call as $DJLIN$. We use the fact that f is nonlinear if and only if there are two points $w\neq y\in\mathbb{F}_2^n$ such that $|\widehat{f}(w)|>0$ and $|\widehat{f}(y)|>0$. Our linearity-testing circuit is slightly simpler compared to the earlier solutions [8,5] but uses the same underlying idea.

To model the black-box access to f, it is customary in quantum algorithms to use an unitary operator of the form $U_f|x\rangle = (-1)^{f(x)}|x\rangle$ for making queries [5]. For n-bit f, DJ can be implemented as $H^n\cdot U_f\cdot H^n$ applied to an n-qubit register that is initialized to $|0^n\rangle$. It is easy to show that the output state becomes $\sum_{w\in\mathbb{F}_2^n}\frac{\widehat{f}(w)}{2^n}|w\rangle$ after only one query to U_f.

The circuit for $DJLIN$ starts with two n-bit and one single bit registers initialized to $|0^n\rangle|0^n\rangle|0\rangle$ and then applies DJ twice, independently, on the first two

[5] One can also use an operator with the transformation $|x\rangle|b\rangle\mapsto|x\rangle|x\rangle|b\oplus f(x)\rangle$. Both the operators can be easily converted to the other by using Hadamard gates.

registers. We would get $\frac{1}{2^{2n}} \sum_x \sum_y \widehat{f}(x)\widehat{f}(y)|x\rangle|y\rangle|0\rangle$. Now apply the following operation on all three registers: $|a,b\rangle|c\rangle \mapsto |a,b\rangle|c \oplus NEQ(a,b)\rangle$; here, $NEQ(a,b)$ is a Boolean function that outputs 1 whenever $a \neq b$ and the final operation can be easily implemented by using, say, CNOT and Toffoli gates.

The output state of the $DJLIN$ circuit can be written as

$$\frac{1}{2^{2n}} \sum_x \widehat{f}^2(x)|x,x\rangle|0\rangle + \frac{1}{2^{2n}} \sum_{x \neq y} \widehat{f}(x)\widehat{f}(y)|x,y\rangle|1\rangle.$$

Our quantum algorithm will measure the third register and outputs "linear" if it observes it in $|0\rangle$. We now discuss the suitability of $DJLIN$ for linearity testing.

Lemma 7. *$DJLIN$ uses two queries and* always *outputs "linear" if the input f is linear. On the other hand, if f is nonlinear, then the probability of incorrectly outputing "linear" is $\frac{1}{2^{4n}} \sum_{w \in \mathbb{F}_2^n} \widehat{f}^4(w)$ which can be at most $\frac{1}{2^{2n}} \max_{w \in \mathbb{F}_2^n} \widehat{f}^2(w)$.*

Proof. The number of queries is clearly 2. For the next claim, recall that for a linear f, $\widehat{f}(x)$ is non-zero (actually, $\pm 2^n$) for exactly one point, say w. In that case, the output state is going to be $|w,w\rangle|0\rangle$ and so, $DJLIN$ will always output "linear".

For the final claim, take any nonlinear f and let $\tau = \frac{1}{2^n} \max_{w \in \mathbb{F}_2^n} |\widehat{f}(w)|$. The probability that $|0\rangle$ is observed at the end is $\frac{1}{2^{4n}} \sum_{w \in \mathbb{F}_2^n} \widehat{f}^4(w) \leq \frac{1}{2^{2n}} \sum_w \tau^2 \widehat{f}^2(w)$ (since $\frac{1}{2^n}|\widehat{f}(w)| \leq \tau$). Applying Parseval's equality, we get the upper bound of τ^2 on the error probability. □

We will use $p(f)$ to denote the probability that our algorithm outputs an incorrect answer for input f; based on the above lemma, $p(f) = 0$ if f is linear and $p(f) = \frac{1}{2^{4n}} \sum_x \widehat{f}^4(x)$ if f is nonlinear. Our quantum linearity testing algorithm is better compared to the classical BLR algorithm since, not only the former uses two queries against three queries by the latter, but also the error probability of the latter was shown to be upper bounded by $\frac{1}{2} + \frac{1}{2}\tau^2$ by Blum et al. [4] and Aumann et al. [1]. However, the error probability of our algorithm is at most τ^2 which is always less than that of the BLR test.

Having shown a quantum linearity testing algorithm that is comparable, if not better, to the classical BLR algorithm, we next elaborate on the observation that linearity is not preserved even by this quantum algorithm. For this, we construct a fairly simple counter-example on functions with n number of variables for any $n \geq 4$. Consider two n-bit Boolean functions, f_1 such that $\widehat{f}_1(00^{n-1}) = \widehat{f}_1(10^{n-1}) = \frac{1}{\sqrt{2}}2^n$ (therefore, $\widehat{f}(w) = 0$ for all other $w \in \mathbb{F}_2^n$) and f_2 such that $\widehat{f}_2(a0^{n-2}) = \frac{1}{2}2^n$ for $a \in \{00, 01, 10, 11\}$ (therefore, $\widehat{f}(w) = 0$ for all other $w \in \mathbb{F}_2^n$). It is easy to show that $nl(f_1) = (1 - \frac{1}{\sqrt{2}})2^{n-1}$ and $nl(f_2) = \frac{2^{n-1}}{2}$; therefore, $nl(f_1) < nl(f_2)$. However, using Lemma 7, $p(f_1) = \frac{1}{2}$ whereas $p(f_2) = \frac{1}{4}$, i.e., $p(f_1) > p(f_2)$ demonstrating the violation of nonlinearity hierarchy even for quantum tests.

The earlier quantum algorithms [8,5] use different approaches to reduce the probability of error of $DJLIN$ to get a better trade-off in the number of calls to

f and the error compared to classical algorithms, but because they all follow the Lemma 7 at their core, they suffer from the same limitation of not preserving nonlinearity.

6 Conclusion

In this paper we concentrate on the BLR (Blum-Luby-Rubinfeld) linearity test in classical and quantum domains and its limitations in revealing any information on the nonlinearity or weight of the involved function. We ultimately show both in the classical and quantum domain the existence of a pair of functions, say f_1 and f_2 for every input-length $n \geq 4$, for which $nl(f_1) > nl(f_2)$ but $wt(F_1) < wt(F_2)$, and/or $nl(F_1) < nl(F_2)$, for the currently known classical and quantum linearity tests, where $F_i(x, y) = f_i(x) + f_i(y) + f_i(x + y)$ is the relevant BLR function, corresponding to f_i, $i = 1, 2$.

Acknowledgements. The authors would like to thank the reviewers for their excellent comments, which significantly improved the editorial and technical quality of the paper.

References

1. Y. Aumann, J. Håstad, M. Rabin, M. Sudan. *Linear consistency testing.* J. Comput. Syst. Sci. 62 (2001), 589–607.
2. M. Bellare, D. Coppersmith, J. Håstad, M. Kiwi and M. Sudan. *Linearity testing over characteristic two.* IEEE Trans. Inform. Theory 42 (1996), 1781–1795.
3. E. Bernstein and U. Vazirani. *Quantum complexity theory.* Proceedings of the 25th Annual ACM Symposium on Theory of Computing, (ACM Press, New York, 1993), pp. 11–20.
4. M. Blum, M. Luby and R. Rubinfeld. *Self-Testing/Correcting with Applications to Numerical Problems.* J. Comput. Syst. Sci. 47:3 (1993), 549–595.
5. K. Chakraborty and S. Maitra. *Improved quantum test for linearity of a Boolean function.* arXiv:1306.6195 [quant-ph], 2013.
6. T. W. Cusick, P. Stănică. Cryptographic Boolean Functions and Applications (Ed. 2). Academic Press, San Diego, CA, 2017.
7. J. Håstad and A. Wigderson. *Simple analysis of graph tests for linearity and PCP.* Random Structures and Algorithms 22:2 (2003), 244–254.
8. M. Hillery and E. Andersson. *Quantum tests for the linearity and permutation invariance of Boolean functions.* Review A 84, 062329 (2011), 1–7.
9. T. Kaufman, S. Litsyn and N. Xie. Breaking the ϵ-soundness bound of the linearity test over $\mathbb{F}_2$. SIAM Journal of Computing 39:5 (2010), 1988–2003.
10. A. Samorodnitsky and L. Trevisan. *A PCP characterization of NP with optimal amortized query complexity.* In Proceedings of the 32nd ACM symposium on Theory of Computation, pp. 191–199, 2000.
11. D. Deutsch and R. Jozsa. *Rapid solutions of problems by quantum computation.* In Proc. Royal Society of London A. 439 (1992), 553–558.