Quantum Information Processing (2020) 19:311
https://doi.org/10.1007/s11128-020-02817-z

A quantum algorithm to estimate the Gowers U_2 norm and linearity testing of Boolean functions

C. A. Jothishwaran[1] · Anton Tkachenko[3] · Sugata Gangopadhyay[2] · Constanza Riera[3] · Pantelimon Stănică[4]

Received: 11 May 2020 / Accepted: 12 August 2020 / Published online: 24 August 2020

Abstract
We propose a quantum algorithm to estimate the Gowers U_2 norm of a Boolean function, and extend it into a second algorithm to distinguish between linear Boolean functions and Boolean functions that are ϵ-far from the set of linear Boolean functions, which seems to perform better than the classical BLR algorithm. Finally, we outline an algorithm to estimate Gowers U_3 norms of Boolean functions.

Keywords Boolean functions · Fourier spectrum · Gowers uniformity norms · Quantum algorithms

✉ Sugata Gangopadhyay
sugata.gangopadhyay@cs.iitr.ac.in

C. A. Jothishwaran
jothi@ph.iitr.ac.in

Anton Tkachenko
Anton.Tkachenko@hvl.no

Constanza Riera
csr@hvl.no

Pantelimon Stănică
pstanica@nps.edu

1 Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, Roorkee 247667, India

2 Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Roorkee 247667, India

3 Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, 5020 Bergen, Norway

4 Department of Applied Mathematics, Naval Postgraduate School, Monterey, CA 93943-5216, USA

1 Introduction

Learning a Boolean function from its values resulting from querying it, and possibly storing the query results in a list, is a central problem of both computational learning theory and cryptography. The form of access to a Boolean function that allows us to query it and always obtain the correct result is called an oracle access to a Boolean function. Computational learning theory introduced by Valiant [17] involves the approximation of a Boolean function f given its oracle access. Linial et al. [12] were among the first researchers to consider the Fourier approach to learning. From their research, we know that if the Fourier coefficients of a Boolean function are concentrated on a small collection of domain points, then knowing them leads to identifying linear functions that are highly correlated to the original Boolean function. This spectral analysis of Boolean functions is also relevant to cryptography, where we are interested in constructing functions that are difficult to learn. Goldreich and Levin [7] proposed a learning algorithm that finds large Fourier coefficients with high probability, given oracle access to a Boolean function. Thus, Goldreich-Levin Theorem is significant for both cryptography and computational learning theory. The linearity testing algorithm proposed by Blum et al. [2], now called the BLR Test, is another cornerstone achievement of computational learning theory. The proof of the probability bounds achieved by the BLR test depends on the (Fourier) spectral analysis of Boolean functions. For an introduction to the analysis of Boolean functions and applications to diverse areas of computer science, we refer to O'Donnell [15].

The problem of estimating properties of Boolean functions by using quantum algorithms has been studied from the eighties onward. One of the first quantum algorithms is the Deutsch–Jozsa algorithm [6]. The Deutsch–Jozsa algorithm distinguishes between constant and balanced Boolean functions using a single query by looking at a particular Fourier coefficient. The importance of the approach introduced by Deutsch and Jozsa [6] is evident from several investigations during the last two decades in essentially the same direction. We refer to the results of Maitra and Mukhopadhyay [13], Xie et al. [18], and Bera et al. [1] to demonstrate that fact. These investigations stem from a desire to understand whether the availability of quantum computers lets us learn the properties of Boolean functions more efficiently, or discover unknown weaknesses of Boolean functions used in designing cryptosystems.

We have undertaken our present study against this backdrop. We choose to estimate Gowers uniformity norms [8] since the Gowers uniformity norm of dimension d of a function f tells us the extent of correlation of f to the polynomial phase functions of degree up to $d-1$. In this paper, we consider the Gowers uniformity norm U_2 of dimension 2 for Boolean functions, and find a quantum estimate of its upper bound. We also propose a linearity test of Boolean functions based on the same quantum algorithm, which seems to perform better than the classical BLR algorithm. Finally, we propose an algorithm to estimate the U_3 norm.

1.1 Boolean functions

We denote the ring of integers, the set of positive integers, and the fields of real numbers and complex numbers by $\mathbb{Z}$, $\mathbb{Z}^+$, $\mathbb{R}$, and $\mathbb{C}$, respectively. For any $n \in \mathbb{Z}^+$, the set $[n] = \{i \in \mathbb{Z}^+ : 1 \leq i \leq n\}$, and $\mathbb{F}_2^n = \{x = (x_1, \ldots, x_n) : x_i \in \mathbb{F}_2, \text{ for all } i \in [n]\}$ where $\mathbb{F}_2$ is the prime field of characteristic 2. Addition in each of the above algebraic systems is denoted by '+'. An n-variable Boolean function F is a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. The set of all such functions is denoted by $\mathfrak{B}_n$. Each function $F \in \mathfrak{B}_n$ has its character form $f : \mathbb{F}_2^n \to \mathbb{R}$ defined by $f(x) = (-1)^{F(x)}$, for all $x \in \mathbb{F}_2^n$. In this article, abusing notation, we refer to the character form f as Boolean functions and go to the extent of writing $f \in \mathfrak{B}_n$, whenever $F \in \mathfrak{B}_n$, if there is no danger of confusion. For any $x, y \in \mathbb{F}_2^n$, the inner product $x \cdot y = \sum_{i\in[n]} x_i y_i$ where the sum is over $\mathbb{F}_2$. The (Hamming) weight of a vector $u = (u_1, \ldots, u_n) \in \mathbb{F}_2^n$ is $\mathrm{wt}(u) = \sum_{i\in[n]} u_i$, where the sum is over $\mathbb{Z}$. The weight of a Boolean function $F \in \mathfrak{B}_n$, or equivalently $f \in \mathfrak{B}_n$ is the cardinality $\mathrm{wt}(F) = |\{x \in \mathbb{F}_2^n : F(x) \neq 0\}|$, or equivalently $\mathrm{wt}(f) = |\{x \in \mathbb{F}_2^n : f(x) \neq 1\}|$. The Hamming distance between $F, G \in \mathfrak{B}_n$, or equivalently, between $f, g \in \mathfrak{B}_n$ is $d_H(F, G) = |\{x \in \mathbb{F}_2^n : F(x) \neq G(x)\}|$, or, $d_H(f, g) = |\{x \in \mathbb{F}_2^n : f(x) \neq g(x)\}|$. Any Boolean function $F \in \mathfrak{B}_n$ can be expressed as a polynomial, called the algebraic normal form (ANF),

$$F(x_1, \ldots, x_n) = \sum_{u\in\mathbb{F}_2^n} \lambda_u x^u \text{ where } \lambda_u \in \mathbb{F}_2, \text{ and } x^u = \prod_{i\in[n]} x_i^{u_i}. \tag{1}$$

The algebraic degree of a Boolean function $\deg(F) = \max\{\mathrm{wt}(u) : \lambda_u \neq 0\}$. A Boolean function with algebraic degree at most 1 is said to be an affine function. An affine function in $\mathfrak{B}_n$ is of the form $\varphi(x) = u \cdot x + \varepsilon$ for some $u \in \mathbb{F}_2^n$ and $\varepsilon \in \mathbb{F}_2$. An affine function with $\varepsilon = 0$ is said to be a linear function. We denote the set of all n-variable affine functions by $\mathfrak{A}_n$, and the set of all n-variable linear functions by $\mathfrak{L}_n$.

The Fourier series expansion of $f \in \mathfrak{B}_n$ is

$$f(x) = \sum_{u\in\mathbb{F}_2^n} \widehat{f}(u)(-1)^{u\cdot x}. \tag{2}$$

The coefficients $\widehat{f}(u)$ are said to be the Fourier coefficients of f. The transformation $f \mapsto \widehat{f}$ is the Fourier transformation of f. It is known that

$$\sum_{x\in F_2^n} (-1)^{v\cdot x} = \begin{cases} 0 & \text{if } v \neq 0 \\ 2^n & \text{if } v = 0. \end{cases} \tag{3}$$

Equations (2) and (3) yield

$$\begin{aligned} \sum_{x\in\mathbb{F}_2^n} f(x)(-1)^{u\cdot x} &= \sum_{x\in\mathbb{F}_2^n} \sum_{v\in\mathbb{F}_2^n} \widehat{f}(v)(-1)^{(u+v)\cdot x} \\ &= \sum_{v\in\mathbb{F}_2^n} \widehat{f}(v) \sum_{x\in\mathbb{F}_2^n} (-1)^{(u+v)\cdot x} = 2^n \widehat{f}(u), \end{aligned} \tag{4}$$

that is, $\widehat{f}(u) = 2^{-n} \sum_{x \in \mathbb{F}_2^n} f(x)(-1)^{u \cdot x}$. The sum

$$\begin{aligned} \sum_{x \in \mathbb{F}_2^n} f(x)^2 &= \sum_{x \in \mathbb{F}_2^n} \sum_{u \in \mathbb{F}_2^n} \widehat{f}(u)(-1)^{u \cdot x} \sum_{v \in \mathbb{F}_2^n} \widehat{f}(v)(-1)^{v \cdot x} \\ &= \sum_{u \in \mathbb{F}_2^n} \sum_{v \in \mathbb{F}_2^n} \widehat{f}(u)\widehat{f}(v) \sum_{x \in \mathbb{F}_2^n} (-1)^{(u+v) \cdot x} = 2^n \sum_{u \in \mathbb{F}_2^n} \widehat{f}(u)^2. \end{aligned}$$

The identity $\sum_{x \in \mathbb{F}_2^n} \widehat{f}(x)^2 = 2^{-n} \sum_{x \in \mathbb{F}_2^n} f(x)^2$, is known as the *Plancherel's identity*. This is true for $f : \mathbb{F}_2^n \to \mathbb{R}$. If $f \in \mathfrak{B}_n$, we have the *Parseval's identity* $\sum_{u \in \mathbb{F}_2^n} \widehat{f}(u)^2 = 1$. For $f, g \in \mathfrak{B}_n$ the convolution product, $f * g$ is defined as

$$(f * g)(x) = 2^{-n} \sum_{y \in \mathbb{F}_2^n} f(y)g(x + y) = 2^{-n} \sum_{y \in \mathbb{F}_2^n} f(x + y)g(y). \tag{5}$$

Using (4) on (5)

$$\begin{aligned} \widehat{f * g}(u) &= 2^{-n} \sum_{x \in \mathbb{F}_2^n} (f * g)(x)(-1)^{u \cdot x} = 2^{-2n} \sum_{x \in \mathbb{F}_2^n} \sum_{y \in \mathbb{F}_2^n} f(y)g(x + y)(-1)^{u \cdot x} \\ &= 2^{-2n} \sum_{x \in \mathbb{F}_2^n} \sum_{y \in \mathbb{F}_2^n} f(y)(-1)^{u \cdot y} g(x + y)(-1)^{u \cdot (x+y)} \\ &= \left(2^{-n} \sum_{y \in \mathbb{F}_2^n} f(y)(-1)^{u \cdot y} \right) \left(2^{-n} \sum_{x \in \mathbb{F}_2^n} g(y)(-1)^{u \cdot x} \right) = \widehat{f}(u)\widehat{g}(u). \end{aligned} \tag{6}$$

For each $x \in \mathbb{F}_2^n$, $(f * f)(x) = 2^{-n} \sum_{y \in \mathbb{F}_2^n} f(y)f(x + y)$ is said to be the *autocorrelation* of f at x, and $\widehat{f * f}(x) = \widehat{f}(x)^2$.

The derivative of $f \in \mathfrak{B}_n$ at $c \in \mathbb{F}_2^n$ is the function

$$\Delta_c f(x) = f(x)f(x + c), \quad \text{for all} \quad x \in \mathbb{F}_2^n. \tag{7}$$

We write

$$\Delta_{x^{(1)},\ldots,x^{(k)}} f(x) = \prod_{S \subseteq [k]} f\left(x + \sum_{i \in S} x^{(i)} \right), \tag{8}$$

where $x^{(i)} \in \mathbb{F}_2^n$, for all $i \in [k]$, and some $k \in \mathbb{Z}^+$. In Eq. (7), we have defined derivatives of a Boolean function when the codomain of the function is $\{1, -1\}$. In that case, the resulting derivative turns out to be function from $\mathbb{F}_2^n$ to $\{1, -1\}$. The derivative of a Boolean function $F : \mathbb{F}_2^n \to \mathbb{F}_2$, at a point $a \in \mathbb{F}_2^n$ is

$$\Delta_a F(x) = F(x) + F(x + a), \quad \text{for all} \quad x \in \mathbb{F}_2^n. \tag{9}$$

For any $a, b \in \mathbb{F}_2^n$

$$\Delta_{a,b}F(x) = F(x) + F(x+b) + F(x+a) + F(x+a+b). \tag{10}$$

For $a, b, c \in \mathbb{F}_2^n$,

$$\begin{aligned}\Delta_{a,b,c}F(x) =& F(x) + F(x+c) + F(x+b) + F(x+b+c) + F(x+a) \\ &+ F(x+a+c) + F(x+a+b) + F(x+a+b+c).\end{aligned} \tag{11}$$

In general, for $x^{(1)}, \ldots, x^{(k)} \in \mathbb{F}_2^n$,

$$\Delta_{x^{(1)},\ldots,x^{(k)}}F(x) = \sum_{S\subseteq[k]} F\left(x + \sum_{i\in S} x^{(i)}\right). \tag{12}$$

1.2 Gowers uniformity norms

Gowers [8] introduced (now, called Gowers) uniformity norms in his work on Szmerédi's theorem. In their full generality, Gowers uniformity norms operate over functions from finite sets to the field of complex numbers. For an introductory reading on the topic, we refer to the Ph.D. thesis of Chen [4]. The Gowers U_k norm of $f \in \mathfrak{B}_n$, denoted by $\|f\|_{U_k}$, is defined as

$$\|f\|_{U_k} = \left(2^{-(k+1)n} \sum_{x,x^{(1)},\ldots,x^{(k)}\in\mathbb{F}_2^n} \prod_{S\subseteq[k]} f\left(x + \sum_{i\in S} x^{(i)}\right)\right)^{2^{-k}}. \tag{13}$$

The Gowers U_2 norm is

$$\begin{aligned}\|f\|_{U_2} &= \left(2^{-3n} \sum_{x\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{b\in\mathbb{F}_2^n} f(x)f(x+a)f(x+b)f(x+a+b)\right)^{2^{-2}} \\ &= \left(2^{-3n} \sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n} f(x)f(x+a) \sum_{b\in\mathbb{F}_2^n} f(x+b)f(x+a+b)\right)^{2^{-2}} \\ &= \left(2^{-3n} \sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n} f(x)f(x+a) \sum_{y\in\mathbb{F}_2^n} f(y)f(y+a)\right)^{2^{-2}} \\ &= \left(2^{-n} \sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n} \widehat{f}(x)^2(-1)^{x\cdot a} \sum_{y\in\mathbb{F}_2^n} \widehat{f}(y)^2(-1)^{y\cdot a}\right)^{2^{-2}}\end{aligned}$$

$$= \left(2^{-n} \sum_{x\in\mathbb{F}_2^n} \sum_{y\in\mathbb{F}_2^n} \widehat{f}(x)^2 \widehat{f}(y)^2 \sum_{a\in\mathbb{F}_2^n} (-1)^{(x+y)\cdot a}\right)^{2^{-2}} = \left(\sum_{x\in\mathbb{F}_2^n} \widehat{f}(x)^4\right)^{2^{-2}}.$$

The Gowers U_3 norm is

$$\begin{aligned}\|f\|_{U_3} = \Big(2^{-4n} \sum_{x\in\mathbb{F}_2^n} \sum_{a\in\mathbb{F}_2^n} \sum_{b\in\mathbb{F}_2^n} \sum_{c\in\mathbb{F}_2^n} & f(x) f(x+a) f(x+b) f(x+a+b) \\ & f(x+c) f(x+a+c) f(x+b+c) f(x+a+b+c)\Big)^{2^{-3}}.\end{aligned} \tag{14}$$

Substituting the derivative in (14)

$$\begin{aligned}\|f\|_{U_3} &= \Big(2^{-4n} \sum_{c\in\mathbb{F}_2^n} \sum_{x,a,b\in\mathbb{F}_2^n} \Delta_c f(x) \Delta_c f(x+a) \Delta_c f(x+b) \Delta_c f(x+a+b)\Big)^{2^{-3}} \\ &= \Big(2^{-n} \sum_{c\in\mathbb{F}_2^n} \|\Delta_c f(x)\|_{U_2}^2\Big)^{2^{-3}}.\end{aligned} \tag{15}$$

In general, the Gowers U_k norm of $f \in \mathfrak{B}_n$ is

$$\begin{aligned}\|f\|_{U_k} &= \left(2^{-(k+1)n} \sum_{x,x^{(1)},\ldots,x^{(k)}\in\mathbb{F}_2^n} \prod_{S\subseteq[k]\setminus[2]} \prod_{T\subseteq[2]} f\left(x + \sum_{i\in S} x^{(i)} + \sum_{j\in T} x^{(j)}\right)\right)^{2^{-k}} \\ &= \left(2^{-(k+1)n} \sum_{x,x^{(1)},\ldots,x^{(k)}\in\mathbb{F}_2^n} \prod_{T\subseteq[2]} \prod_{S\subseteq[k]\setminus[2]} f\left(x + \sum_{i\in S} x^{(i)} + \sum_{j\in T} x^{(j)}\right)\right)^{2^{-k}} \\ &= \left(2^{-(k+1)n} \sum_{x^{(3)},\ldots,x^{(k)}\in\mathbb{F}_2^n} \sum_{x^{(1)},x^{(2)}\in\mathbb{F}_2^n} \prod_{T\subseteq[2]} \Delta_{x^{(3)},\ldots,x^{(k)}} f\left(x + \sum_{j\in T} x^{(j)}\right)\right)^{2^{-k}} \\ &= \left(2^{-(k-2)n} \sum_{x^{(3)},\ldots,x^{(k)}\in\mathbb{F}_2^n} \|\Delta_{x^{(3)},\ldots,x^{(k)}} f(x)\|_{U_2}^2\right)^{2^{-k}}.\end{aligned} \tag{16}$$

Equation (16) shows the relation between the Gowers U_k norm and the U_2 norms of the $(k-2)$th derivatives of f. The time complexity of computing the Gowers U_2 norm of a Boolean function $f \in \mathfrak{B}_n$ is $O(n2^{2n})$. Arguing in the same way, the time complexity of computing Gowers U_k norm is $O(n2^{kn})$.

In this paper, we propose a quantum algorithm to estimate an upper bound of Gowers U_2 norm and based upon that, we find a quantum counterpart of the BLR linearity testing [2] that tends to perform better than the classical version, assuming the availability of a quantum computer with a sufficient number of qubits. The complexities

of the quantum algorithms are independent of the number of variables n, of course, again with the strong assumption of the availability of a fairly large quantum computer.

1.3 Gowers uniformity norms and approximation of Boolean functions by low degree Boolean functions

In this section, we discuss the connection between the Gowers uniformity norms and the approximation of Boolean functions by low degree Boolean functions. The nonlinearity, denoted by $nl(f)$, of a Boolean function $f \in \mathfrak{B}_n$ is the minimum Hamming distance from f to all affine functions in $\mathfrak{A}_n$. That is

$$nl(f) = \min\{d_H(f, \varphi) : \varphi \in \mathfrak{A}_n\}. \tag{17}$$

The rth-order nonlinearity of a Boolean function f, denoted by $nl_r(f)$, is the minimum Hamming distance from f to the functions having algebraic degree less than or equal to r. The first-order nonlinearity $nl_1(f) = nl(f)$. It is well known that (cf. [5])

$$nl(f) = 2^{n-1}\Big(1 - \max_{x\in\mathbb{F}_2^n} |\widehat{f}(x)|\Big). \tag{18}$$

Carlet [3] obtained lower bounds of rth-order nonlinearity of Boolean functions by using nonlinearities of their higher-order derivatives. This establishes a relationship between the rth-order nonlinearities of Boolean functions and Fourier coefficients of their derivatives. Gowers uniformity norms involve Fourier coefficients of higher-order derivatives (16), and serve the same purpose as evident from the following theorem.

Theorem 1 ([4], Fact 2.2.1) *Let $k \in \mathbb{Z}^+$, $\epsilon > 0$. Let $P : \mathbb{F}_2^n \to \mathbb{F}_2$ be a polynomial of degree at most k, and $f : \mathbb{F}_2^n \to \mathbb{R}$. Suppose $|2^{-n} \sum_{x\in\mathbb{F}_2^n} f(x)(-1)^{P(x)}| \geq \epsilon$. Then $\|f\|_{U_{k+1}} \geq \epsilon$.*

For $k = 1$, informally, this means that if, for some f, the norm $\|f\|_{U_2}$ is small then its Fourier coefficients are small, and therefore f has high nonlinearity. On the other hand,

$$\begin{aligned}
\|f\|_{U_2}^4 &= \sum_{x\in\mathbb{F}_2^n} \widehat{f}(x)^4 \\
&\leq \max_{x\in\mathbb{F}_2^n} |\widehat{f}(x)|^2 \sum_{x\in\mathbb{F}_2^n} \widehat{f}(x)^2 \\
&= \max_{x\in\mathbb{F}_2^n} |\widehat{f}(x)|^2 \text{ (applying Parseval identity)} \\
&= (1 - 2^{1-n} nl(f))^2 \text{(using 18)}.
\end{aligned} \tag{19}$$

Equation (19) tells us that if a Boolean function has high nonlinearity then its U_2 norm is small, and if U_2 norm is large, then the nonlinearity is small.

The second-order nonlinearity of a Boolean function is the minimum of the distances of that function from the quadratic Boolean function (i.e., the Boolean functions with algebraic degree at most 2). By Theorem 1, for all polynomials $P : \mathbb{F}_2^n \to \mathbb{F}_2$

of degree at most 2, if $\|f\|_{U_3} < \epsilon$, then $|2^{-n} \sum_{x \in \mathbb{F}_2^n} f(x)(-1)^{P(x)}| < \epsilon$. Therefore, the second-order nonlinearity of such functions ought to be high. Green and Tao [9] proved that just as for U_2, if a Boolean function has high second-order nonlinearity, then its U_3 norm is low. They also proved that such an implication is not valid for U_k norms for $k \geq 4$.

The discussion in this section points to the fact that Gowers U_2 and U_3 norms have the promise of being good indicators for the first- and second-order nonlinearities of a Boolean function. Determination of these nonlinearities has complexities that scale exponentially with the number of input variables of Boolean functions. In the following section, we propose a quantum algorithm to estimate an upper bound of the Gowers U_2 norm that is probabilistic in nature, the probability converges as $e^{-2m^2t^2}$ where m is the number of trials and t is a positive error margin.

1.4 Quantum information: definitions and notation

In this section, we will introduce some notation that we use throughout the paper. For an introduction to quantum computing, we refer to Rieffel and Polak [16], or Nielsen and Chuang [14].

A *qubit* or *qu-bit* can be described by a vector $|\psi\rangle = (a, b)^{\mathrm{T}} \in \mathbb{C}^2$, where 'T' indicates the transpose, $|a|^2$ is the probability of observing the value 0 when we measure the qubit, and $|b|^2$ is the probability of observing 1. If both a and b are nonzero, the qubit has both the value 0 and 1 at the same time, and we call this a *superposition*. Once we have measured the qubit, however, the superposition collapses, and we are left with a classical state that is either 0 or 1 with certainty. A state of n qubits is represented by a normalized complex vector with 2^n elements. We define $\langle\psi|$ as the conjugate transpose of $|\psi\rangle$. This notation is known as the bra-ket notation. We denote the standard basis (column) vectors as $|0\rangle$ and $|1\rangle$, and then $|\psi\rangle = (a, b)^{\mathrm{T}} = a|0\rangle + b|1\rangle$.

In the following, we will use the conventional notation $|a\rangle|b\rangle := |a\rangle \otimes |b\rangle$, or $|ab\rangle := |a\rangle \otimes |b\rangle$. A state on n qubits can be represented as a $\mathbb{C}$-linear combination of the vectors of the standard basis $|\psi\rangle = \sum_{x \in \mathbb{F}_2^n} a_x |x\rangle$, where $a_x \in \mathbb{C}$, $\forall x \in \mathbb{F}_2^n$, and $\sum_{x \in \mathbb{F}_2^n} |a_x|^2 = 1$; the set of vectors $|x\rangle$ forms a basis for the n qubit states and is referred to as the computational basis. Let $|0_n\rangle$ be the quantum state associated with the zero vector in $\mathbb{F}_2^n$. The vectors $|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$ and $|-\rangle = \frac{|0\rangle-|1\rangle}{\sqrt{2}}$ define the Hadamard basis for single qubit states. Any Boolean function $F \in \mathfrak{B}_n$ can be implemented as a *bit oracle implementation* U_F, so that:

$$|x\rangle|\varepsilon\rangle \xrightarrow{U_F} |x\rangle|\varepsilon + F(x)\rangle \tag{20}$$

Here, $x \in \mathbb{F}_2^n$ and $\varepsilon \in \mathbb{F}_2$. This implies that U_F acts on the last qubit of the $n+1$ qubit state $|x\rangle|\varepsilon\rangle$ depending on the value of $F(x)$, i.e., $|\varepsilon\rangle$ is "flipped" if $F(x) = 1$ and is left unchanged, otherwise. The bit oracle U_F maps vectors belonging to the computational basis of the $n+1$ qubit states to other vectors in the same basis. U_F is therefore a unitary transformation. The n qubit state vector $|x\rangle$ is the input to the oracle and the state $|\varepsilon\rangle$ is the target qubit .

If the target qubit for U_F is $|-\rangle$, then $|x\rangle|-\rangle \xrightarrow{U_F} (-1)^{F(x)}|x\rangle|-\rangle$. We write $|x\rangle \xrightarrow{U_F} (-1)^{F(x)}|x\rangle$ with the understanding that there is an additional target qubit in the $|-\rangle$ state that remains unchanged and refer to this as the *phase oracle implementation* of the function F. Suppose that a computational basis state is of the form $|x^{(1)}\|x^{(2)}\|\ldots\|x^{(m)}\rangle$ where for any two vectors $x \in \mathbb{F}_2^r$ and $y \in \mathbb{F}_2^s$, the concatenation $x\|y = (x_1, \ldots, x_r, y_1, \ldots, y_s)$ is a vector in $\mathbb{F}_2^{r+s}$. It is reasonable to write $|x^{(1)}\|x^{(2)}\|\ldots\|x^{(m)}\rangle = |x^{(1)}\rangle|x^{(2)}\rangle\ldots|x^{(m)}\rangle$. The vector $x^{(i)} \in \mathbb{F}_2^{r_i}$, for some $r_i \in \mathbb{Z}^+$ is said to be the content of the ith register. If $x^{(i)}, x^{(j)} \in \mathbb{F}_2^r$, for some $r \in \mathbb{Z}^+$, we define MCNOT_i^j as

$$|x^{(1)}\rangle\ldots|x^{(i)}\rangle\ldots|x^{(j)}\rangle\ldots|x^{(m)}\rangle \xrightarrow{\mathrm{MCNOT}_i^j} |x^{(1)}\rangle\ldots|x^{(i)}+x^{(j)}\rangle\ldots|x^{(j)}\rangle\ldots|x^{(m)}\rangle.$$

We can realize the transformation induced by MCNOT_i^j by using r conventional CNOT gates between the ith and j^{th} registers with the qubits in the jth register acting as the control qubits.

Let $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ be the 2×2 identity matrix, and $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ be the 2×2 Hadamard matrix. The tensor product of matrices is denoted by $\otimes$. The matrix H_n is recursively defined as:

$$\begin{aligned} H_2 &= H \otimes H, \\ H_n &= H_{n-1} \otimes H, \quad \text{for all} \quad n \geq 3. \end{aligned} \tag{21}$$

Note that, for $x \in \mathbb{F}_2^n$, $H_n|x\rangle = 2^{\frac{-n}{2}} \sum_{x' \in \mathbb{F}_2^n} (-1)^{x \cdot x'}|x'\rangle$. In the next section, we propose an algorithm to compute Gowers U_2 norm of Boolean functions. Our approach resembles that employed by Bera et al. [1] to estimate the autocorrelation spectra of Boolean functions.

2 A quantum algorithm to estimate Gowers uniformity norms

The Gowers U_2 norm of a Boolean function, $F \in \mathfrak{B}_n$, may be estimated by starting with the initial quantum state $|0_n\rangle|0_n\rangle|0_n\rangle$, and apply the following transformations:

$$\begin{aligned} |0_n\rangle|0_n\rangle|0_n\rangle &\xrightarrow{H_n^{\otimes 3}} 2^{-\frac{3n}{2}} \sum_{b \in \mathbb{F}_2^n} \sum_{a \in \mathbb{F}_2^n} \sum_{x \in \mathbb{F}_2^n} |x\rangle|a\rangle|b\rangle \\ &\xrightarrow{U_F \otimes I \otimes I} 2^{-\frac{3n}{2}} \sum_{b \in \mathbb{F}_2^n} \sum_{a \in \mathbb{F}_2^n} \sum_{x \in \mathbb{F}_2^n} (-1)^{F(x)}|x\rangle|a\rangle|b\rangle \\ &\xrightarrow{\mathrm{MCNOT}_1^2} 2^{-\frac{3n}{2}} \sum_{b \in \mathbb{F}_2^n} \sum_{a \in \mathbb{F}_2^n} \sum_{x \in \mathbb{F}_2^n} (-1)^{F(x)}|x+a\rangle|a\rangle|b\rangle \end{aligned}$$

$$
\begin{aligned}
&\xrightarrow{U_F\otimes I\otimes I} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{F(x)+F(x+a)}|x+a\rangle|a\rangle|b\rangle\\
&\xrightarrow{\text{MCNOT}_1^2} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{F(x)+F(x+a)}|x\rangle|a\rangle|b\rangle\\
&\xrightarrow{\text{MCNOT}_1^3} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{F(x)+F(x+a)}|x+b\rangle|a\rangle|b\rangle\\
&\xrightarrow{U_F\otimes I\otimes I} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{F(x)+F(x+a)+F(x+b)}|x+b\rangle|a\rangle|b\rangle\\
&\xrightarrow{\text{MCNOT}_1^2} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{F(x)+F(x+a)+F(x+b)}|x+a+b\rangle|a\rangle|b\rangle\\
&\xrightarrow{U_F\otimes I\otimes I} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{\Delta_{a,b}F(x)}|x+a+b\rangle|a\rangle|b\rangle\\
&\xrightarrow{\text{MCNOT}_1^2} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{\Delta_{a,b}F(x)}|x+b\rangle|a\rangle|b\rangle\\
&\xrightarrow{\text{MCNOT}_1^3} 2^{-\frac{3n}{2}} \sum_{b\in\mathbb{F}_2^n}\sum_{a\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{\Delta_{a,b}F(x)}|x\rangle|a\rangle|b\rangle\\
&\xrightarrow{H_n^{\otimes 3}} \sum_{a',b',x'\in\mathbb{F}_2^n}(2^{-3n}\sum_{x,a,b\in\mathbb{F}_2^n}(-1)^{\Delta_{a,b}F(x)+a\cdot a'+b\cdot b'+x\cdot x'})|x'\rangle|a'\rangle|b'\rangle.
\end{aligned}
\tag{22}
$$

It should be remembered that in addition to the three n-qubit registers used, there is an additional target qubit that is in the $|-\rangle$ state required for the phase oracle implementation of F defined in Sect. 1.4, owing to this fact the qubit has been dropped from the sequence of operations, for brevity.

The above sequence of operations excluding both the $H_n^{\otimes 3}$ is summarized as

$$
|x,a,b\rangle \xrightarrow{\mathfrak{D}_F} (-1)^{\Delta_{a,b}F(x)}|x,a,b\rangle. \tag{23}
$$

The complete algorithm is therefore the application of the unitary operation $H_n^{\otimes 3}\circ \mathfrak{D}_F \circ H_n^{\otimes 3}$ to the $|0_n\rangle|0_n\rangle|0_n\rangle$ input state.

The probability that a measurement on the output state of the algorithm yields the result $|0_n\rangle|0_n\rangle|0_n\rangle$ is given by

$$
\Pr[x'=a'=b'=0_n] = \left(2^{-3n}\sum_{x,a,b\in\mathbb{F}_2^n}(-1)^{\Delta_{a,b}F(x)}\right)^2
$$

and since $f(x) = (-1)^{F(x)}$, using (10)

$$\Pr[x' = a' = b' = 0_n] = (\|f\|_{U_2}^4)^2 = \|f\|_{U_2}^8. \tag{24}$$

2.1 Estimation of the upper bound of Gowers U_2 norm

Let the final output state at the end of the transformation described in (22) be

$$|\Psi\rangle = \sum_{a',b',x' \in \mathbb{F}_2^n} C(x', a', b')|x'a'b'\rangle. \tag{25}$$

The probability amplitude of the state $|x'a'b'\rangle$ is

$$C(x', a', b') = 2^{-3n} \sum_{x,a,b \in \mathbb{F}_2^n} (-1)^{\Delta_{a,b}F(x) + a\cdot a' + b\cdot b' + x\cdot x'}. \tag{26}$$

The outcome of a measurement, with respect to the computational basis, performed on the output state is a $3n$ bit string $(x'\|a'\|b')$, where $x', a', b' \in \mathbb{F}_2^n$, and the probability of measuring said string is $|C(x', a', b')|^2$. Therefore, the eighth power of the Gowers U_2 norm is given by $|C(0_n, 0_n, 0_n)|^2$. The next theorem outlines a strategy to determine a probabilistic upper bound of the Gowers U_2 norm.

Theorem 2 *We assume that the measurements are done with respect to the computational basis. Suppose that Y is a random variable defined on the set of all possible measurement outcomes on the quantum state $H_n^{\otimes 3} \circ \mathfrak{D}_F \circ H_n^{\otimes 3}|0_n\rangle|0_n\rangle|0_n\rangle$ as*

$$Y(x', a', b') = 2^{-3n}(x'\|a'\|b')_{10},$$

where $(x'\|a'\|b')_{10}$ is the decimal value of the concatenated $3n$ bit string. Following the usual convention, we write Y instead of $Y(x', a', b')$. Suppose that $(Y_1, \ldots, Y_m)$ be a random sample such that each Y_i is independent and identically distributed as Y. Let $\overline{Y} = \frac{1}{m}\sum_{i\in[m]} Y_i$. Then,

$$\Pr\left[\|f\|_{U_2} \leq (1 + t - \overline{Y})^{1/2^3}\right] \geq 1 - \exp(-2m^2t^2),$$

for any positive real number t.

Proof Let the expectation of Y, $E[Y] = \mu$. Let $\Pr[Y = 0] = \|f\|_{U_2}^8 = p$, so $\Pr[Y \neq 0] = 1 - p$. The range of the random variable Y has 2^{3n} distinct values in the interval $[0, 1]$ including 0. Let us denote them by $y_0, y_1, \ldots, y_{2^{3n}-1}$, where $y_j = 2^{-3n} j$. The expectation of Y is

$$\begin{aligned}
\mu = E[Y] &= y_0 \Pr[Y = 0] + y_1 \Pr[Y = y_1] + \cdots + y_{2^{3n}-1} \Pr[Y = y_{2^{3n}-1}] \\
&= y_1 \Pr[Y = y_1] + y_2 \Pr[Y = y_2] + \cdots + y_{2^{3n}-1} \Pr[Y = y_{2^{3n}-1}] \\
&< \Pr[Y = y_1] + \Pr[Y = y_2] + \cdots + \Pr[Y = y_{2^{3n}-1}] \\
&= \Pr[Y \neq 0] = 1 - p.
\end{aligned} \tag{27}$$

Suppose that $(Y_1, \ldots, Y_m)$ be a random sample of size m. The sample mean is $\overline{Y} = \frac{1}{m}\sum_{i\in[m]} Y_i$. By the Hoeffding inequality [11]

$$\Pr\left[\overline{Y} \geq \mu + t\right] \leq \exp(-2m^2t^2). \tag{28}$$

where t is any positive real number. Using Eqs. (27) and (28),

$$\begin{aligned} \Pr\left[1 - p > \mu \geq \overline{Y} - t\right] &\geq 1 - \exp(-2m^2t^2), \\ \text{which implies, } \Pr\left[p < 1 + t - \overline{Y}\right] &\geq 1 - \exp(-2m^2t^2), \\ \text{that is, } \Pr\left[\|f\|_{U_2} < (1 + t - \overline{Y})^{1/2^3}\right] &\geq 1 - \exp(-2m^2t^2). \end{aligned} \tag{29}$$

The theorem is shown. □

The last line of (29) tells us that if we measure m times and compute $\overline{Y}$, then the probability that $\|f\|_{U_2}$ is bounded above by $(1 + t - \overline{Y})^{1/2^3}$ is $1 - \exp(-2m^2t^2)$. Therefore, with an appropriate choice of m and t, we can estimate an upper bound of the Gowers U_2 norm of f with a very high probability.

2.2 Discussion of the algorithm and its complexity

The time complexity of the algorithm depends on the number of elementary gate operations that are applied. The unitary transformation applied is $H_n^{\otimes 3} \circ \mathfrak{D}_F \circ H_n^{\otimes 3}$ on the initial state $|0_n\rangle|0_n\rangle|0_n\rangle$. As defined in (23), $\mathfrak{D}_F$ comprised of the unitary operation MCNOT_1^3 is applied twice, and the operations MCNOT_1^2 and $U_F \otimes I \otimes I$ are applied four times each.

The oracle operation U_F is considered to be a given, and therefore, the gates used in the implementation of U_F are not explicitly considered and the oracle is treated as a single block operation. The action of $U_F \otimes I \otimes I$ is therefore treated as a constant time operation. The MCNOT gates are each implemented by n elementary CNOT gates, where $F \in \mathfrak{B}_n$. The Hadamard transformations applied before and after $\mathfrak{D}_F$, each require $3n$ single qubit Hadamard gates. The entire algorithm, excluding U_F, consists only of CNOT and Hadamard gates. The quantum algorithm for estimating the Gowers U_2 norm of a Boolean Function $F \in \mathfrak{B}_n$ requires $6n$ Hadamard gates and $6n$ CNOT gates and is therefore a linear time algorithm.

The quantum algorithm described thus far is a probabilistic algorithm. The quantity that is directly estimated is $p = \Pr[Y = 0]$ as defined in the proof of Theorem 2. The relation between the sample mean of the outcomes obtained by running the algorithm a number of times and the value of p is given by (29). The measured quantity $\overline{Y}$ provides an upper bound for p which in turn gives an upper bound for the Gowers U_2 norm.

As shown in (29), the probability that the sample mean $\overline{Y}$ defines an upper bound for p converges to unity as exponential of the square of the number of times the algorithm is run (or queried). If $\Pr\left[p < 1 + t - \overline{Y}\right] \geq 1 - \delta$, where $0 < \delta < 1$, the query

complexity of the algorithm may be calculated as follows:

$$\begin{aligned} \exp(-2m^2t^2) &= \delta \\ 2m^2t^2 &= \log\left(\frac{1}{\delta}\right) \\ m &= \left[\frac{1}{2t^2}\log\left(\frac{1}{\delta}\right)\right]^{1/2} \end{aligned} \tag{30}$$

It should be noted here that the query complexity of this algorithm is a function of the values of δ and t and is independent of the size of the Boolean function n.

2.3 Linear approximation employing the Gowers U_2 norm

We start by defining distance between Boolean functions in terms of probabilities.

Definition 3 For any two functions $f, g \in \mathfrak{B}_n$,

$$\text{dist}(f, g) = \Pr_{\mathbf{x}\sim\mathbb{F}_2^n}[f(\mathbf{x}) \neq g(\mathbf{x})] = \frac{d_H(f, g)}{2^n}$$

where $\mathbf{x}$ is a random variable uniformly distributed over $\mathbb{F}_2^n$.

The function f is said to be ϵ-close to g if $\text{dist}(f, g) \leq \epsilon$, and ϵ-far from g if $\text{dist}(f, g) > \epsilon$. We will now design Algorithm 1 to determine whether a function is linear or ϵ-far from linear; we refer to Hillery and Anderson [10, Section III] for a discussion on such tests.

Algorithm 1 Linearity checking with the Gowers U_2 norm.

Input: Quantum implementation of $f \in \mathfrak{B}_n$.

1: Initial state: $2^{-\frac{3n}{2}} \sum_{a,b,x\in\mathbb{F}_2^n} |x, a, b\rangle$.

2: Perform the following sequence of transformations:

$$\begin{aligned} & 2^{-\frac{3n}{2}} \sum_{a,b,x\in\mathbb{F}_2^n} |x, a, b\rangle \\ \xrightarrow{\mathfrak{D}_F} & 2^{-\frac{3n}{2}} \sum_{a,b,x\in\mathbb{F}_2^n} (-1)^{\Delta_{a,b}F(x)} |x, a, b\rangle \\ \xrightarrow{H_n^{\otimes 3}} & \sum_{a',b',x'\in\mathbb{F}_2^n} (2^{-3n} \sum_{x,a,b\in\mathbb{F}_2^n} (-1)^{\Delta_{a,b}F(x)+a\cdot a'+b\cdot b'+x\cdot x'}) |x', a', b'\rangle. \end{aligned}$$

3: Measure the output state with respect to the computational basis.

4: If the measurement result is $|0_n, 0_n, 0_n\rangle$ then "ACCEPT" (the function is linear).

5: Else "REJECT".

Theorem 4 *If f is a linear function then the output is "ACCEPT" with probability* 1. *If f is ϵ-far from linear functions, then probability of "REJECT" is greater than* $1 - \exp(-8\epsilon)$.

Proof If f is a linear functions, then the output is "ACCEPT" with certainty. This directly follows from the definition of Gowers U_2 norm. If f is ϵ-far from linear functions, then

$$\|f\|_{U_2}^{2^3} \leq \left(1 - 2\frac{nl(f)}{2^n}\right)^4 \leq (1-2\epsilon)^4.$$

This means that the probability that the output is "ACCEPT" is less than or equal to $(1-2\epsilon)^4$; therefore the probability of "REJECT" is greater than $1-(1-2\epsilon)^4 \approx 1-\exp(-8\epsilon)$. □

The result concerning the BLR test is:

Theorem 5 [15, Theorem 1.30] *Suppose the BLR Test accepts $F : \mathbb{F}_2^n \to \mathbb{F}_2$ with probability $1-\epsilon$. Then, f is ϵ-close to being linear.*

By the BLR test, if a function is ϵ-far from the linear functions, and it is promised that we have such functions and linear functions only, then given a function from the latter class, the probability that the algorithm will REJECT is greater than ϵ.

Remark 6 The algorithm presented here has been implemented in the IBM quantum machine (https://www.ibm.com/quantum-computing/) for some small examples and has given the expected output of probabilities.

Acknowledgements Research of C. A. Jothishwaran and Sugata Gangopadhyay is a part of the project "Design and Development of Quantum Computing Toolkit and Capacity Building" sponsored by the Ministry of Electronics and Information Technology (MeitY) of the Government of India.

Appendix: Generalization to higher Gowers norms

The same technique can be used for other Gowers norms. For instance, we can apply the unitary transformation $H_n^{\otimes 4} \circ \mathfrak{D}_F^3$ to the state $2^{-2n} \sum_{x\in\mathbb{F}_2^n} \sum_{a\in\mathbb{F}_2^n} \sum_{b\in\mathbb{F}_2^n} \sum_{c\in\mathbb{F}_2^n} |x\rangle|a\rangle|b\rangle|c\rangle$, where, with notation $M_i^j = \text{MCNOT}_i^j$ and $U_F^3 = U_F \otimes I \otimes I \otimes I$,

$$\begin{aligned}\mathfrak{D}_F^3 = {} & M_1^3 \circ U_F^3 \circ M_1^3 \circ M_1^4 \circ U_F^3 \circ M_1^3 \circ U_F^3 \circ M_1^2 \circ U_F^3 \circ M_1^4 \\ & \circ U_F^3 \circ M_1^3 \circ U_F^3 \circ M_1^2 \circ U_F^3.\end{aligned}$$

We obtain thus the state $\sum_{a',b',b',x'\in\mathbb{F}_2^n} 2^{-4n} \sum_{a,b,c,x\in\mathbb{F}_2^n} (-1)^{\Delta_{a,b,c}F(x)} |x,a,b,c\rangle$. Then, $\Pr[x'=a'=b'=c'=0_n] = \left(2^{-4n} \sum_{a,b,c,x\in\mathbb{F}_2^n} (-1)^{\Delta_{a,b,c}F(x)}\right)^2$, and, using (11), $\Pr[x'=a'=b'=c'=0_n] = \left(\|f\|_{U_3}^8\right)^2 = \|f\|_{U_3}^{16}$.

References

1. Bera, D., Maitra, S., Tharrmashastha, S.: Efficient quantum algorithms related to autocorrelation spectrum. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology—INDOCRYPT 2019—20th International Conference on Cryptology in India, Hyderabad, India, December 15–18, 2019, LNCS 11898, pp. 415–432. Springer (2019). https://doi.org/10.1007/978-3-030-35423-7_21
2. Blum, M., Luby, M., Rubinfeld, R.: Self-testing/correcting with applications to numerical problems. J. Comput. Syst. Sci. **47**(3), 549–595 (1993)
3. Carlet, C.: Recursive lower bounds on the nonlinearity profile of Boolean functions and their applications. IEEE Trans. Inf. Theory **54**(3), 1262–1272 (2008)
4. Chen, V.Y.: The Gowers' norm in the testing of Boolean functions. Ph.D. thesis, Massachusetts Institute of Technology (2009)
5. Cusick, T., Stănică, P.: Cryptographic Boolean Functions and Applications, 2nd edn. Elsevier, Amsterdam (2017)
6. Deutsch, D., Jozsa, R.: Rapid solution of problems by quantum computation. Proc. R. Soc. Lond. Ser. A **439**, 553–558 (1992)
7. Goldreich, O., Levin, L.: A hard-core predicate for all one-way functions. In: Proceedings of the 21st Annual ACM Symposium on Theory of Computing, pp. 25–32
8. Gowers, W.T.: A new proof of Szemerédi's theorem. Geom. Funct. Anal. GAFA **11**(3), 465–588 (2001)
9. Green, B., Tao, T.: An inverse theorem for the Gowers U_3 norm. Proc. Edinb. Math. Soc. **51**, 75–153 (2008)
10. Hillery, M., Andersson, E.: Quantum tests for the linearity and permutation invariance of Boolean functions. Phys. Rev. A **84**, 062329 (2011). https://doi.org/10.1103/PhysRevA.84.062329
11. Hoeffding, W.: Probability inequalities for sums of bounded random variables. J. Am. Stat. Assoc. **58**, 13–30 (1963)
12. Linial, N., Mansour, Y., Nisan, N.: Constant depth circuits, Fourier transforms and learnability. J. ACM **40**(3), 607–620 (1993)
13. Maitra, S., Mukhopadhyay, P.: The Deutsch–Jozsa algorithm revisited in the domain of cryptographically significant Boolean functions. Int. J. Quantum Inf. **03**(02), 359–370 (2005)
14. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information, 10th edn. Cambridge University Press, Cambridge (2011)
15. O'Donnell, R.: Analysis of Boolean Functions. Cambridge University Press, Cambridge (2014)
16. Rieffel, E., Polak, W.: Quantum Computing: A Gentle Introduction, 1st edn. The MIT Press, Cambridge (2011)
17. Valiant, L.: A theory of learnable. Commun. ACM **27**(11), 1134–1142 (1984)
18. Xie, Z., Qiu, D., Cai, G.: Quantum algorithms on Walsh transform and Hamming distance for Boolean functions. Quantum Inf. Process. **17**, 139 (2018)