Complexity of Computing Functions by Quantum Branching Programs

Farid Ablayev* Marek Karpinski† Airat Khasianov‡

Abstract

We consider the problems of computing certain types of boolean functions which we call *Equality*, *Semi-Simon* and *Periodicity* functions. For all these problems, we prove linear lower complexity bounds on oblivious *Ordered Read-Once Quantum Branching Programs* (quantum *Ordered Binary Decision Diagrams*). We present also two different approaches to prove existence of efficient quantum branching programs for those boolean functions matching the lower complexity bounds.

1 Introduction

It has turned out to be quite a subtle task to come up with some good quantum algorithms. The most famous discoveries in quantum computing still remain those made by Shor [Sho97] and Grover [Gro97], i.e. those that drew initially attention to this field.

In this paper we consider a restricted model of computation known as an *oblivious Ordered Read-Once Quantum Branching Program*. The main reason for the investigation of restricted models of quantum algorithms was first proposed by Ambainis and Freivalds in 1996 (See [AF98]). Considering one-way quantum finite automata, they suggested that first quantum-mechanical computers would consist of a comparatively simple quantum -mechanical part connected to a classical computer.

A good text on *branching programs* is the book of Wegener [Weg00], another good introduction to the subject is done by Meinel and Theobald [MT98]. The *quantum branching programs* were first introduced by Ablayev, Gainutdinova, Karpinski [AGK01] (*leveled programs*), and independently by Nakanishi, Hamaguchi, Kashiwabara [NHK00] (*non-leveled programs*). Later it was shown by Sauerhoff [SS04] that *non-leveled* programs can be transformed into *leveled programs* with no more than a factor $(n+1)^2$ increase in size, where n is the number of levels.

The most comprehensive introduction to the field of *quantum computations*published so far is the book of Nielsen and Chuang [NC00].

In this paper we essentially use the *fingerprinting* technique developed by Freivald in 1979 [Fre79]. It was later successfully applied in the *quantum automata* setting by Ambainis and Freivald in 1998 [AF98]. Subsequently, the same technique was adapted for the quantum branching programs by Ablayev, Gainutdinova and Karpinski in 2001 [AGK01]. Results presented in this paper naturally extend previous research.

For oblivious *Read-Once Quantum Branching Programs*, we consider three computational problems:

1. *Equality* tells if two binary strings a and b equal each other;
2. *Periodicity* tells wether a given Boolean function f has a period s;

*Dept. of Computer Science, Kazan University. Supported by SNF grant 200021-107327/1.

†Dept. of Computer Science, University of Bonn. Supported in part by DFG grants and Hausdorff Center research grant EXC59-1.

‡Dept. of Computer Science, Kazan University. The research supported by the BIGS-MPA Graduate School program at the University of Bonn.

3. *Semi-Simon Problem* given a Boolean function f and parameter s the problem is to determine wether for any argument x of f it holds that $f(x) = f(x \oplus s)$, where the addition is bitwise modulo 2.

Note here, the functions *Periodicity Semi-Simon* are rather simplified versions of the well known period finding and Simon problems (see [ME99, NC00] for details). Nevertheless, we believe that upper bounds on these functions give a good insight to how to solve more general *hidden subgroup problem* [ME99, Høy97]. Subsequently, an efficient algorithm for the hidden subgroup problem would imply efficient solutions for both the *period finding problem*, and original *Simon problem*.

In this paper we prove linear upper and lower bounds for all three Boolean functions mentioned in the introduction. Our upper bounds hold for arbitrary *ordering* of the input variables. The lower bounds show that the upper bounds are almost tight.

2 Preliminaries

We start with some basic definitions

Definition 2.1 (as defined in [AGK01]). A Quantum Branching Program $\mathcal{P}$ of width d and length l (a $(d,l) - QBP$) based on a quantum system in $\mathcal{H}^d$ is defined as a triple.

$$\mathcal{P} = \langle T, |\psi_0\rangle, F\rangle,$$

where T is a sequence (of length l) of d-dimensional quantum transformations on $\mathcal{H}^d$:

$$T = (j_i, U_i(0), U_i(1))_{i=1}^{l}$$

$|\psi_0\rangle$ is the initial configuration of P. $F \subseteq \{1, \ldots, d\}$ is the set of accepting states.
We define a computation on $\mathcal{P}$ for an input $\sigma = \sigma_1, \ldots, \sigma_n \in \{0,1\}^n$ as follows:

1. A computation of $\mathcal{P}$ starts with $|\psi_0\rangle$. On the i-th step, $1 \leq i \leq l$, of computation $\mathcal{P}$ performs transformation $|\psi\rangle \longrightarrow U_i(\sigma_{j_i})|\psi\rangle$

2. After the l-th (last) step of transformations, $\mathcal{P}$ measures its configuration $|\psi_\sigma\rangle = U_l(\sigma_{j_l}) U_{l-1}(\sigma_{j_{l-1}}) \ldots U_1(\sigma_{j_1})|\psi_0\rangle$. The measurement is represented by a diagonal zero-one projection matrix M, where $M_{ii} = 1$ if $i \in F$ and $M_{ii} = 0$ if $i \notin F$. The probability $P_{accept}(\sigma)$ of $\mathcal{P}$ accepting input σ is defined by the following equation.

$$P_{accept}(\sigma) = ||M|\psi_\sigma\rangle||^2.$$

Definition 2.2. We say a function $f \in \mathbb{B}_n$ is *ϵ-computed* (*ϵ-accepted*) by a quantum branching program $\mathcal{P}$ if for every binary input $\sigma \in \{0,1\}^n$ the probability that $f(\sigma)$ equals the output of $\mathcal{P}$ on the input σ is at least $1/2 + \epsilon$.

Definition 2.3. We call a Quantum Branching Program $\mathcal{P}$ an *Ordered Read-Once Quantum Branching Program* (Quantum Ordered Binary Decision Diagram) if each variable $x \in \{x_1, \ldots, x_n\}$ occurs in the sequence T of the quantum transformation of the program $\mathcal{P}$ at most once, and according to a certain ordering of the input variables. If not specified otherwise, we assume the natural ordering $(x_1, \ldots, x_n)$ for all the branching programs considered in this paper.

In this paper we consider only ordered read-once programs. Therefore, our central *complexity measure* is *width* of a quantum branching program.

Definition 2.4. Let $\mathcal{P}$ be a quantum branching program acting in d-dimensional Hilbert space $\mathcal{H}^d$. We define *width* of $\mathcal{P}$ as follows:

$$Width(\mathcal{P}) = d.$$

Notice following, while *width* of a quantum branching program $\mathcal{P}$ is the dimensionality of the space it is acting in, the number of physical particles needed to implement $\mathcal{P}$ would only be $\log(Width(\mathcal{P}))$.

Definition 2.5. We shall use the notation $1QBP$for the class of all *Ordered Read-Once Quantum Branching Program* (Quantum Ordered Binary Decision Diagram).

Definition 2.6. For a set of input variables $x = \{x_1, \ldots, x_n\}$, we denote by σ_x an assignment of binary values (zeros or ones) to the variables $x_1, \ldots, x_n$. Clearly, σ_x is a binary sequence, such that $|\sigma_x| = |x|$.

Now we define several fundamental Boolean functions that we investigate in this research.

Definition 2.7. Let us define the *Equality* function $\mathrm{EQ}_n(x, y)$.

$$\begin{aligned}&\mathrm{EQ}_n(x, y) \equiv [x = y], \text{where}\\ &x = \{x_1, \ldots, x_{n/2}\},\\ &y = \{y_1, \ldots, y_{n/2}\}\end{aligned}$$

The binary sequence comparison is not trivial only if the two sequences are of the same length. Therefore, it is a reasonable assumption to take that n is even, and $|x| = |y| = n/2$.

Definition 2.8. For a set of input variables $x = \{x_0, \ldots, x_{n-1}\}$, and s – the period parameter, we define the *Periodicity* function $\mathrm{Period}_{s,n}(x)$.

$$\mathrm{Period}_{s,n}(x) \equiv \begin{cases} 1 & \text{if } x_i = x_{i+s \mod n}, i = \overline{0, n-1};\\ 0 & \text{otherwise.}\end{cases}$$

Definition 2.9. For a set of input variables $x = \{x_0, \ldots, x_{n-1}\}$, and $s \in (0, n]$ we define the *Semi-Simon* function $\text{Semi-Simon}_{s,n}(\sigma)$.

$$\text{Semi-Simon}_{s,n}(x) \equiv \begin{cases} 1 & x_i = x_{i \oplus s}, i = \overline{0, n-1};\\ 0 & \text{otherwise.}\end{cases}$$

Note that $\oplus$ is a bitwise addition modulo 2. Here we treat i both ways: as a natural number, and as a binary sequence representing the number.

3 The upper bound for Equality

Theorem 1. *The function $EQ_n(x, y)$ can be computed with one-sided error $o(1) \leq \frac{1}{7}, (n \to \infty)$ by a $1QBP$ of width $O(n)$, where $n = |xy|$ is the length of the input.*

Proof. We shall build an $(O(n))$–$1QBP$ $\mathcal{C}$.

Let us first introduce one qubit (acting in 2-dimensional Hilbert space) 1QBP*s* $\mathcal{A}_k = \langle T_k, |\psi_0\rangle, \Psi_k, F\rangle$, where $k \in \{1, \ldots, p-1\}$. Let $|0\rangle$ and $|1\rangle$ be two orthonormal fixed states to form a basis of $\mathcal{H}^2$. Now we show how for every k we can construct a program $\mathcal{A}_k$, for a $p \in (\sigma, 2\sigma) \cap PRIMES$. According to *Bertrand's postulate* there's always such p [Nag51].

1. The $QOBDD$ receives the input assignment σ, assigning zeros and ones to the input variables $x_1, \ldots, x_{n/2}, y_1, \ldots, y_{n/2}$.
2. $|\psi_0\rangle = |0\rangle$;
3. $F = \{|0\rangle\}$;
4. $\Psi_k = \{|0\rangle, |1\rangle\}$

5. $T_k = (i, U_i(0), U_i(1))_{i=1}^n$

   (a) On the "x-part" of the input σ:

$$U_i(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi k\sigma_i 2^i}{p} & \sin\frac{2\pi k\sigma_i 2^i}{p} \\ -\sin\frac{2\pi k\sigma_i 2^i}{p} & \cos\frac{2\pi k\sigma_i 2^i}{p} \end{pmatrix};$$

   (b) On the "y-part" of the input σ:

$$U_i(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi k\sigma_i 2^i}{p} & -\sin\frac{2\pi k\sigma_i 2^i}{p} \\ \sin\frac{2\pi k\sigma_i 2^i}{p} & \cos\frac{2\pi k\sigma_i 2^i}{p} \end{pmatrix}.$$

Clearly if $x = y$ then $\mathcal{A}_k$ accepts σ with probability 1,

Now we use the well-known *fingerprinting* technique the same way it was applied in **Theorem 5** from [AGKMP], and the **Theorem 8** from [AF98]. We shall explain it here for convenience of the reader.

Lemma 1. *For any σ, an assignment of xy, such that $x \neq y$, at least $(p-1)/2$ of all $\mathcal{A}_k$ for different k reject σ with probability at least $1/2$.*

Proof. Our branching program is defined so that reading the input $\sigma = xy$ it first rotates the state vector (initially $|\psi_0\rangle$) through the angle $\theta_k(x) = \frac{2k\pi}{p}x$ in the direction towards $|1\rangle$ axis, and then backwards, while reading y, through the angle $\theta_k(y) = -\frac{2k\pi}{p}y$

After reading the input σ the assignment of xy ($x \neq y$) the branching program ends up in the state ψ^k.

$$|\psi^k\rangle = (\cos\theta_k)\,|0\rangle + (\sin\theta_k)\,|1\rangle, \tag{1}$$

$$\theta_k = \theta_k(x) + \theta_k(y) = \frac{2k\pi}{p}x - \frac{2k\pi}{p}y = \frac{2k\pi(x-y)}{p} \tag{2}$$

Since $k \in \{1, \ldots, p-1\}$, it is co-prime with p, thus θ_k would constitute a finite cyclic additive group of residues modulo p. Clearly the elements of $I = \{\theta_k | k \in \{1, \ldots, p-1\}\}$ are uniformly distributed on the circumference of a unit circle.

Now the angle θ_k is corresponds to a program rightfully rejecting the input with probability larger than $1/2$ if and only if $\theta_k \in \left[\frac{\pi}{4}, \frac{3\pi}{4}\right] \cup \left[\frac{5\pi}{4}, \frac{7\pi}{4}\right]$, for only then have we $\cos^2\theta_k < \frac{1}{2}$. Next we conclude the proof of the lemma.

$$\left| I \cap \left(\left[\frac{\pi}{4}, \frac{3\pi}{4}\right] \cup \left[\frac{5\pi}{4}, \frac{7\pi}{4}\right] \right) \right| \geq \left\lfloor \frac{p}{2} \right\rfloor \geq \frac{p-1}{2}. \tag{3}$$

Thus the lemma follows. □

Lemma 2. *There is a set S of width-2 quantum branching programs such that*

1. $|S| = t = \lceil 24 \log p \rceil$,
2. *For arbitrary σ, such that $\sigma_x \neq \sigma_y$, at least $1/4$ of all programs in S reject σ with probability at least $1/2$.*

Proof. The following nonuniform construction proves the lemma.

The construction is trivial: for a fixed input $\sigma \leq p-1$ a branching program from $\{\mathcal{A}_1, \ldots, \mathcal{A}_{p-1}\}$ is selected uniformly and randomly and added to the initially empty set S.

Let's call a program $\mathcal{A}_i$ "good" if for input assignment σ such that $\sigma_x \neq \sigma_y$ program $\mathcal{A}_i$ rejects the input with probability at least $1/2$.

Consider a sum $X = \Sigma_{i=1}^{n} X_i$, where $X_i = [\mathcal{A}_i \; is \; not \; "good"]$. By the definition of the set S and previous lemma, probability that every given program $\mathcal{A}_i \in S$ is not "good" equals $q = \frac{1}{2}$. Now set parameter $\theta = \frac{1}{2}$ and apply Chernoff's bound (See e.g. [Pap94]).

$$\Pr\left[X \geq (1+\theta)qt\right] \leq e^{-\frac{\theta^2}{3}qt}, \tag{4}$$

that after substitution gives

$$\Pr\left[X \geq \frac{3}{4}t\right] \leq e^{-\log p} = \frac{1}{p}. \tag{5}$$

That is, the probability of constructing a set S with less than $\frac{1}{4}$ of "good" (for any given $\sigma < p$) entries is not greater than $\frac{1}{p}$. Let's call such a set S *"bad"*. Thus the probability $\Pr\left[S \; is \; "bad"\right]$ that the set S is "bad" for *at least one* $\sigma < p$ is at most the following positive fraction.

$$\Pr\left[S \; is \; "bad"\right] \leq \frac{p-1}{p}. \tag{6}$$

Therefore, there exists a set which is not "bad" for all inputs $\sigma < p$. Recalling that $|S| = t = \lceil 24 \log p \rceil$ the lemma follows. □

At length, we construct the quantum OBDD $\mathcal{C}$, computing equality with the constant error 1/8. The program $\mathcal{C}$ is constructed out of elementary programs $\mathcal{A}_i$ described in the beginning of the proof. We take only those of $\mathcal{A}_i$ that belong to the *nonconstructive* set S. W.l.o.g. we assume that set $S = \{\mathcal{A}_1, \ldots, \mathcal{A}_t\}$ Existence of the set S was already proved based on Chernoff bound. The set S was defined so that at least $\frac{1}{4}$ of all programs in S reject $\sigma = xy, \; x \neq y$ wit probability at least $\frac{1}{2}$. Recall that $t = |S|$.

$\mathcal{C} = \langle T, |\psi_0\rangle, \Psi, F\rangle$:

1. The $QOBDD$ receives input σ, an assignment to input variables $xy = \{x_1, \ldots, x_{n/2}, y_1, \ldots, y_{n/2}\}$;

2. For each program $\mathcal{A}_k \in S$ we introduce corresponding states $|0_k\rangle$ and $|1_k\rangle$, where $|0\rangle$ is replaced with $|0_k\rangle$, and $|1\rangle$ is replaced with $|1_k\rangle$ in each respective program $\mathcal{A}_k$. Where $|0_k\rangle$ is a $2t$-dimensional *zero-one* column-vector with the only *one* in the $(2k+1)$th position, and $|1_k\rangle$ is a $2t$-dimensional *zero-one* column-vector with the only *one* in the $2(k+1)$th position. Thus $\Psi = \{|0_k\rangle, |1_k\rangle \mid \mathcal{A}_k \in S\}$, where $|\Psi| = 2t \in O(n)$.

3. The initial state
$$|\psi_0\rangle = \sum_{|0_k\rangle \in \Psi} \frac{1}{\sqrt{t}} |0_k\rangle = \frac{1}{\sqrt{t}} \left(\overbrace{1,0,1,0,\ldots,1,0}^{2t} \right)^T;$$

4. The accepting set
$$F = \{|\psi_0\rangle\};$$

5. T - the transition function is defined by block-diagonal unitary matrices $T = \left(i, U_i^{\mathcal{C}}(0), U_i^{\mathcal{C}}(1)\right)_{i=1}^{n}$:

$$U_i^{\mathcal{C}}(\sigma_i) = \begin{bmatrix} U_i^{\mathcal{A}_1}(\sigma_i) & \ldots & \ldots & \ldots \\ \ldots & U_i^{\mathcal{A}_2}(\sigma_i) & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & U_i^{\mathcal{A}_2}(\sigma_i) \end{bmatrix},$$

where $U_i^{\mathcal{A}_k}(\sigma_i)$ is the unitary transformation applied on the i-th step of the program $\mathcal{A}_k$.

It is easy to see that $Width(\mathcal{C}) = 2t \in O(n)$.

Notice that for inputs $\sigma = xy, x = y$ $\mathcal{C}$ never errs, the reason is that every $\mathcal{A}_i$ ends up in the accepting state $|0\rangle$. However, if $x \neq y$, the probability $P_{rej}[xy]$ is bounded as follows:

$$P_{rej}[xy] = \sum_{i=1}^{t} \left| \frac{a_i}{\sqrt{t}} \right|^2 \geq \frac{t}{4} \cdot \frac{1}{2t} = \frac{1}{8}.$$

The theorem follows. □

In fact, the error can be reduced arbitrarily having $d = d(\epsilon)$ copies of the program $\mathcal{C}$ taken and run uniformly at random.

4 The upper bound for Periodicity function

Next theorem is inspired by the proof of the **Theorem 1**.

Theorem 2. *For all $s \in (0, n]$ and $\forall \sigma \in \{0,1\}^n$ the function $Period_{s,n}(\sigma)$ can be computed with constant one-sided error < 1 by a $1QBP$ of width $O(s)$.*

Proof. The main idea of the proof is to divide the input sequence into words of length s, and using the elementary 2-state branching programs for equality, compare all resulted subsequences with a chosen sample subsequence of the same set.

Recall that $\text{Period}_{s,n}(\sigma) = 1 \iff \forall i, 1 \leq i \leq n\ (\sigma_i = \sigma_{i+s \mod n})$. Thus, $\text{Period}_{s,n}(\sigma) = 1 \iff$ the elements of the input sequence form a cyclic group of size s over addition of their indicies modulo n. Subsequently, the input having been split into words of length s would consist only of equal words if and only if $\text{Period}_{s,n}(\sigma) = 1$.

The only obstacle on the way is that input length might not be divisible by s. Following lemma adresses this question.

Lemma 3. *Given binary sequence $X = \{x\}_{i=1}^{ks+r}, k, r \in \mathbb{N} \cup \{0\}, s \in \mathbb{N}, r < s$, consider a sequences of s-element subsequences of X with corresponding tails:*
$W = \left(x_1 \dots x_s, x_{s+1} \dots x_{2s}, \cdots, x_{(k-1)s+1} \dots x_{ks}, x_{(k-1)s+r+1} \dots x_{ks+r}\right)$;
$t_{left} = x_1 \dots x_s$,
$t_{right} = x_{(k-1)s+r+1} \dots x_{ks+r}$.
Following statement holds.

$$\forall w, w' \in W, \forall i, 1 \leq i \leq n (x_i = x_{i+s} \iff w = w').$$

Proof. We prove the lemma.
$\Rightarrow$ This direction is straightforward.
$\Leftarrow$ Suppose that $\forall w, w' \in W\ w = w'$ and $t_{left} = t_{right}$. Clearly for all indicies i such that $1 \leq i \leq (k-1)s + r$ it holds that $x_i = x_{i+s}$. Now if there is a number $i_0, (k-1)s + r + 1 \leq i_0 \leq ks + r$ such that $x_{i_0} \neq x_{i_0+s \mod n}$, it would contradict to what follows from that $t_{left} = t_{right}$.
Thus we have proved the lemma. □

It remains to present the $1QBP$ computing $\text{Period}_{s,n}(\sigma)$ with desired properties. It is done by cascading 2-state elementary programs from **Theorem 1** that compute equality, and then applying the same, as in previously mentioned theorem, technique to amplify correct answer probability.

Consider σ as a sequence of words of length s, just like in the statement of **Lemma 3**, $\sigma \to w_1 w_2 \dots w_{\lceil n/s \rceil}$, $w_1 = t_{left}$ and $w_{\lceil n/s \rceil} = t_{right}$.

Similarly the proof of the **Theorem 1**, we shall use elementary two-state branching programs as subroutines of the whole computational process. However, this time we shall use three slightly different types of 2-state branching programs.

1. The first subroutine ($j = 1$) computes $\mathrm{EQ}_s(w_i, w_{i+1})$;
2. The second type ($j = 2$) subroutine program computes $\mathrm{EQ}_s(w_{i+1}, w_{i+2})$, $i = \overline{1, \lceil n/s \rceil - 2}$;
3. The third type ($j = 3$) subroutine checks the equality of the first and the last word in the sequence: $\mathrm{EQ}_s(t_{left}, t_{right})$.

Now we construct $\mathcal{D}_{kj}$, $j \in \{1, 2, 3\}$ for a $p \in \left(2^s, 2^{s+1}\right) \cap PRIMES$. *Bertrand's postulate* assures us again there's always such p [Nag51].

1. The $1QBP$ receives input σ;
2. $|\psi_0\rangle = |0\rangle$;
3. $F = \{|0\rangle\}$;
4. $\Psi_{kj} = \{|0\rangle, |1\rangle\}$
5. $T_{kj} = (i, U_i(0), U_i(1))_{i=1}^{n}$, $n = |\sigma|$. We define the transition function explicitly only for $j = 1$, two other cases are easily derived from this one.
   (a) On the left part of the corresponding $2s$-symbol segment of σ:

$$U_{i1}(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} & \sin\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} \\ -\sin\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} & \cos\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} \end{pmatrix};$$

   (b) On the right part of the corresponding $2s$-symbol segment of σ:

$$U_{i1}(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} & -\sin\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} \\ \sin\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} & \cos\frac{2\pi k p_{\lceil i/s \rceil}\sigma_i 2^i}{p} \end{pmatrix}.$$

   Here $p_{\lceil i/s \rceil} \neq p, i = \overline{1, n}$ are prime numbers.

According to **Lemma 3**, if σ is such that $\mathrm{Period}_{s,n}(\sigma) = 1$ then $\mathcal{D}_k$ accepts σ with probability 1. Moreover, the program attempts to check the chain of equalities $w_1 = w_2 = \ldots = w_{\lceil n/s \rceil}$ in order to reject all σ that have $\mathrm{Period}_{s,n}(\sigma) = 0$.

Let us write down a state, first of the three elementary programs would be in after having read the input. It is similarly easy to do for the remaining two. In the given above notation we can obtain equation for θ_k.

$$|\psi_1^k\rangle = (\cos\theta_k)|0\rangle + (\sin\theta_k)|1\rangle, \tag{7}$$

$$\theta_k = \frac{2\pi k}{p} \sum_{l=1}^{\lceil n/s \rceil - 1} (w_l - w_{l+1}) p_l \tag{8}$$

$$k, w_l < p, l = \overline{1, \lceil n/s \rceil}; p \nmid p_l. \tag{9}$$

Subsequently, since construction of the **Theorem 1** does not depend on the particular matter of the angles we can apply the same technique here.

We treat each of the three elementary subroutine programs analogously to the construction of the **Theorem 1**. Thus for each of them we obtain a program of width $O(s)$ that computes its sub-chain of equalities with the probability of correct answer at least $1/8$. Finally, we take a direct sum of the three to obtain a program $\mathcal{D}$. Clearly, the error probability of that program would not exceed $63/64 < 1$.

We present a formal description of the $1QBP$ $\mathcal{D}$ that computes $\mathrm{Period}_{s,n}(\sigma)$.

$\mathcal{D} = \langle T, |\psi_0\rangle, \Psi, F\rangle$:

1. The $QOBDD$ receives an input σ;
2. $\Psi = \{|0_{k_1}\rangle, |1_{k_1}\rangle \dots |0_{k_{s'}}\rangle, |1_{k_{s'}}\rangle\}$;
3. $$|\psi_0\rangle = \frac{1}{\sqrt{s'}} \left(\overbrace{1,0,1,0,\dots,1,0}^{2s'} \right)^T ;$$
4. $$F = \{|\psi_0\rangle\} ;$$
5. $T = (i, U_i(0), U_i(1))_{i=1}^n$ The transition function definition is clear from the discussion presented above;

where $s' \in O(s)$. The theorem follows. □

In an obvious way, a similar result can be provided for the slightly differently defined *Periodicity´* function.

Definition 4.1. For a set of input variables $x = \{x_0, \dots, x_{n-1}\}$, and s $(1 < s < n)$ – the period parameter, we define the *Periodicity´* function $\text{Period}'_{s,n}(\sigma)$.

$$\text{Period}'_{s,n}(\sigma) \equiv \begin{cases} 1 & \text{if } x_i = x_{i+s}, i = \overline{0, n-s-1}; \\ 0 & \text{otherwise.} \end{cases}$$

5 The upper bound for Semi-Simon problem

This result is essentially a corollary of the **Theorem 1** in contrast to the **Theorem 2** that is based on the proof of **Theorem 1**, but was not derived from the result.

Theorem 3. *For all $s \in (0, n]$ and $\forall \sigma \in \{0,1\}^n$ the function Semi-Simon$_{s,n}(\sigma)$ can be computed with one-sided constant error $1/8$ by a $1QBP$ of width $O(n)$.*

Proof. Computing of equality function will be again in the core for the proof.

Definition 5.1. Denote $S_n = \{i | 1 \le i \le n, i \oplus s \ne i\}$. A set defined for all n-element binary sequences. Clearly, S does not depend on any particular function σ.

Lemma 4. *Set S_n can be partitioned into two sets of the same cardinality $S_n = S_n^1 + S_n^2$ so that for $l = 1, 2$ $\forall i, j \in S_n^l (j \ne i \oplus s)$.*

Proof. First we show that following statement holds. For any $i \in S_n$, if $i \oplus s = j$, there is no $k \in S_n, k \ne j$ so that $k = i \oplus s$.

Clearly $k \ne i$, since it would yield $i = i \oplus s$ that contradicts $i \in S_n$. Moreover if $k = i \oplus s$, and $j = i \oplus s$, then $j \oplus s = i = k \oplus s$, that is $j = k$.

It is also clear that if $i \ne i \oplus s$, $j = i \oplus s$ is contained in S_n. Since $j \in \{0, \dots, n\}$, and $\{0, \dots, n\} = S_n + S_n^c$. So if i does not equal its bitwise sum modulo two with s, i.e. not in S_n^c, it must be in S_n.

Now we take any $i \in S_n$ and add it to initially empty S_n^1, $j = i \oplus s$ we add to also initially empty S_n^2. Then we remove i and j from S_n. Since the latter set is finite and contains even number of elements that can be coupled into pairs $i, i \oplus s$ we will eventually exceed all elements of S_n. Note that the two new sets contain exactly the same number of elements, by construction. Thus, desired partitioning is achieved. We showed the lemma holds. □

Consider following binary sequences:

Definition 5.2.

$$S_{left}(\sigma) = \{\sigma_i\}_{i \in S_n^1},$$
$$S_{right}(\sigma) = \{\sigma_j\}_{j \in S_n^2},$$

where $S_{right}(\sigma)$ is ordered so that $S_{right}(\sigma)_i = \sigma_j$ and $j = i \oplus s$.

Now it is straightforward that

$$\text{Semi-Simon}_{s,n}(\sigma) = 1 \iff \text{EQ}_n(S_{left}(\sigma), S_{right}(\sigma)) = 1. \tag{10}$$

We already know how to compute equality efficiently. In order to build a program computing *Semi-Simon* function we shall have to rearrange rotation angles according to the permutation of the elements in $S_{right}(f)$. Clearly $|S_{left}(\sigma)| = |S_{right}(\sigma)| \in O(n)$ thus **Theorem 1** gives us exactly what we claimed. □

6 The lower bounds

We prove lower bounds in two steps. First, we find lower bounds of the problems for the worst ordering π deterministic π-OBDD. Then we use a general lower bound theorem for quantum OBDD (1QBP) [AGK01]:

Theorem 4 (Ablayev, Gainutdinova, Karpinski). *Let $\epsilon \in (0, 1/2)$. Let f_n be a Boolean function which is $(1/2 + \epsilon)$-computed (computed with a margin ϵ) by a $1QBP$ Q. Then it holds that*

$$width(Q) \in \Omega(\log width(P)), \tag{11}$$

where P is a deterministic OBDD of minimal width computing f_n.

Let's start with the lower bound for the *Equality* function. It turns out that the worst-case ordering $\pi = id$ for *Equality*. We won't further state the order explicitly, assuming it is the "natural" ordering *id*.

Theorem 5. *Let $\epsilon \in (0, 1/2)$. If the function $EQ_n(x, y)$ is ϵ-computed by a $1QBP$ Q then $width(Q) \in \Omega(n)$, where $n = |xy|$ is the length of the input.*

Proof. As we described in the beginning of the section, first we consider deterministic OBDD complexity for the function.

Lemma 5. *If the function $EQ_n(x, y)$ is computed by a deterministic OBDD P then $width(P) \in \Omega(2^n)$, where $n = |xy|$ is the length of the input.*

Proof. We shall prove the lemma by contradiction. Let xy be the input, $|xy| = n = 2m$. Suppose that there's a program P of $width(P) < 2^m$. Now let denote $Vertex(x)$ is the vertex that the path defined by x input leads to. There are 2^m posible inputs for x. On the other hand, by the hypothesis, there are at most $2^m - 1$ states on each level of the OBDD. That is, there exist two different binary sequences σ_1 and σ_2 - inputs for x - so that $Vertex(\sigma_1) = Vertex(\sigma_2)$, by pigeon hole principle. Now whatever input for y would follow in our read-once leveled (oblivious) branching program, the two comparisons $\sigma_1 \stackrel{?}{=} y$ and $\sigma_2 \stackrel{?}{=} y$, for any fixed y, could not be distinguished by the program. Thus having an input $y = \sigma_1$, the program would either accept both of the combinations $\sigma_1\sigma_1, \sigma_1\sigma_2$ or reject them thus contradicting the fact it was computing function $\text{EQ}_n(x, y)$ (see *Definition* 2.7). The lemma follows. □

Final step of the proof is to refer to the *Theorem 4*. Which assures every $1QBP\ \ Q$ computing $\mathrm{EQ}_n(x,y)$ would satisfy relation 11:

$$width(Q) \in \Omega\left(\log 2^n\right) = \Omega(n), \tag{12}$$

This concludes the proof.

□

In order to prove a lower bound for the *Periodicity* function, we reduce to it the *Equality* function.

Theorem 6. *Let $\epsilon \in (0, 1/2)$. If for all $\sigma \in \{0,1\}^n$ the function $Period_{s,n}(\sigma)$ is ϵ-computed by a $1QBP\ \ Q$ then $width(Q) \in \Omega(s)$, where s is the period parameter.*

Proof. We simply reduce $\mathrm{EQ}_n(x,y)$ to $\mathrm{Period}_{s,n}(xy)$. Indeed, let $|xy| = 2s = n$, and let σ be an assignment for $xy = \{x_1, \ldots, x_n, y_1, \ldots, y_n\}$. It is straightforward why the following holds.

$$\mathrm{EQ}_n(x,y) \equiv \mathrm{Period}_{s,n}(xy). \tag{13}$$

Thus the lower bound for the *Periodicity* follows from the lower bound for the *Equality*. This proves the theorem. □

Similarly we prove the lower bound for *Semi-Simon* problem.

Theorem 7. *Let $\epsilon \in (0, 1/2)$. If for all $\sigma \in \{0,1\}^n$ the function $Semi\text{-}Simon_{s,n}(\sigma)$ is ϵ-computed by a $1QBP\ \ Q$ then $width(Q) \in \Omega(n)$.*

Proof. In order to reduce $\mathrm{EQ}_n(x,y)$ to $\text{Semi-Simon}_{s,n}(z)$ we notice that the latter is essentially an equality computation with its input bits mixed up according to the permutation defined by s. Recall, $\text{Semi-Simon}_{s,n}(\sigma) = 1 \iff \forall i \in [1,n](\sigma_i = \sigma_{i \oplus s})$. For an arbitrary input σ and a positive s, computing function $\text{Semi-Simon}_{s,n}(\sigma)$ is equivalent to evaluating following equality.

$$\sigma_1 \ldots \sigma_n \stackrel{?}{=} \sigma_{1 \oplus s} \ldots \sigma_{n \oplus s}. \tag{14}$$

Now let us define s as shown below.

$$s = \overbrace{10\ldots0}^{n} \tag{15}$$

For s defined above, **Expression 14** is exactly the *Equality* evaluation.

$$\begin{aligned} \sigma_1 \ldots \sigma_{n/2} \sigma_{n/2+1} \sigma_n \stackrel{?}{=} & \sigma_{n/2+1} \ldots \sigma_n \sigma_1 \ldots \sigma_{n/2} \sim \\ \sim \sigma_1 \ldots \sigma_{n/2} \stackrel{?}{=} & \sigma_{n/2+1} \ldots \sigma_n \sim \\ \sim \mathrm{EQ}_n(x,y), & \text{where } \sigma \text{is an assignment for} xy. \end{aligned} \tag{16}$$

Finally we notice that for $\sigma = xy$ n would be even, and our reduction goes as follows.

$$\mathrm{EQ}_n(x,y) \equiv \text{Semi-Simon}_{s,n}(xy), \text{ where } s = n/2. \tag{17}$$

That finishes the proof. □

7 A constructive approach

The upper bounds presented so far all suffer the same deficiency – neither of them provides the quantum branching program it claims existence of. The reason is that we do not know how to construct the "good" set S described in the proofs of the upper bound theorems (See proof of the **Theorem 1**). However, quantum branching programs can be used in actually efficiently computing the functions.

Theorem 8. *The function $EQ_n(x,y)$ can be computed using n^2 quantum OBDD of constant width 2 with one-sided constant error, for n large enough.*

Proof. We prove the statement by describing how to construct a $1QBP$ that computes $\mathrm{EQ}_{nw}(x,y)$ith requested properties.

Let us first introduce one qubit (acting in 2-dimensional Hilbert space) 1QBPs $\mathcal{A} = \langle T, |\psi_0\rangle, \Psi, F\rangle$. Let $|0\rangle$ and $|1\rangle$ be two orthonormal fixed states to form a basis of $\mathcal{H}^2$. Now we show how we can construct a program $\mathcal{A}$, for a $p \in (\sigma, 2\sigma) \cap PRIMES$. According to *Bertrand's postulate* there's always such p [Nag51].

1. The $QOBDD$ receives the input assignment σ, assigning zeros and ones to the input variables $x_1, \ldots, x_{n/2}, y_1, \ldots, y_{n/2}$.
2. $|\psi_0\rangle = |0\rangle$;
3. $F = \{|0\rangle\}$;
4. $\Psi = \{|0\rangle, |1\rangle\}$
5. $T = (i, U_i(0), U_i(1))_{i=1}^n$
   (a) On the "x-part" of the input σ:
   $$U_i(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi\sigma_i 2^i}{p} & \sin\frac{2\pi\sigma_i 2^i}{p} \\ -\sin\frac{2\pi\sigma_i 2^i}{p} & \cos\frac{2\pi\sigma_i 2^i}{p} \end{pmatrix};$$
   (b) On the "y-part" of the input σ:
   $$U_i(\sigma_i) = \begin{pmatrix} \cos\frac{2\pi\sigma_i 2^i}{p} & -\sin\frac{2\pi\sigma_i 2^i}{p} \\ \sin\frac{2\pi\sigma_i 2^i}{p} & \cos\frac{2\pi\sigma_i 2^i}{p} \end{pmatrix}.$$

Clearly, if an assignment σ of xy is such that $\sigma_x \neq \sigma_y$ then $\mathcal{A}$ accepts σ with probability 1. If $\sigma x \neq \sigma y$ the probability of correct answer is bounded as follows.

$$P^{\mathcal{A}}_{rej}[\sigma] \geq \left(\sin\frac{2\pi}{p}\right) =$$

$$= \left(\sum_{k=1}^{\infty}\frac{(-1)^{k-1}}{(2k-1)!}\left(\frac{2\pi}{p}\right)^{2k-1}\right)^2 \geq$$

$$\left(\sum_{k=1}^{\infty}\frac{(-1)^{k-1}}{(2k-1)!}\left(\frac{\pi}{n}\right)^{2k-1}\right)^2 \geq$$

$$\geq \left(\frac{2\pi}{p} - \frac{1}{6}\left(\frac{2\pi}{p}\right)^3 + r(p)\right)^2 >$$

$$\text{(where } r(p) > 0\text{)}$$

$$> \left(\frac{2\pi}{p} - \frac{1}{6}\left(\frac{2\pi}{p}\right)^3\right)^2 \geq$$

$$\left(\frac{2\pi}{p} - \frac{\pi}{p}\right)^2,\ p \to \infty$$

$$\text{since } p \leq 2n, \text{and finally}$$

$$\left(\frac{2\pi}{p} - \frac{\pi}{p}\right)^2 = \frac{\pi^2}{p^2} \geq \frac{\pi^2}{4n^2}. \quad (18)$$

Now we use standard probability amplification technique for one-sided error computation. Take n^2 copies of $\mathcal{A}$, and run them independently. Result of the computation would be 1 if and only if all copies after measurement in the standard basis $\{|0\rangle, |1\rangle\}$ end up in the state $|1\rangle$, otherwise result of the computation is 0.

Let us estimate the error probability of the entire computation. It never errs, if $x = y$. An error might only occur if an assignment σ of xy is such that $\sigma_x \neq \sigma_y$. Our computational process is wrong if and only if all copies of $\mathcal{A}$ are wrong. We can easily write down the expression for the probability of such event, when $\sigma_x \neq \sigma_y$. Clearly the error probability equals the accepting probability in this case.

$$P_{acc}[\sigma] = (1 - P^{\mathcal{A}}_{rej}[\sigma])^{n^2} \leq (1 - \frac{\pi^2}{4n^2})^{n^2} \to \frac{1}{e^{\frac{4}{\pi^2}}}. \quad (19)$$

That finishes the proof. □

8 Conclusions

For arbitrary variable ordering we have shown linear upper bounds for three problems:

1. *Equality*;
2. *Semi-Simon* - the simplified Simon problem;
3. *Periodicity* - the simplified version of period finding.

That hints how efficient upper bounds for quantum branching programs can be proved for even more general problems, including the notorious *hidden subgroup problem* [Høy97, ME99].

For the worst case variable ordering, we have shown the upper bounds to be *optimal*. Moreover, exponentially smaller than the lower bounds for deterministic setting.

Unfortunately, this can not raise hope that quantum OBDD are more powerful than their classical counterparts. The incomparability result was proved by Sauerhoff and Sieling in 2004 [SS04]. Therefore, we can only hope for quantum OBDD performing great for some particular functions.

9 Acknowledgements

We would like to thank *Bonn International Graduate School in Mathematics, Physics and Astronomy (BIGS-MPA)* at the University of Bonn for the support of this research as well as for the help in visiting Kazan State University while working on this topic.

References

[AF98] A. Ambainis and R. Freivalds, *1-way quantum finite automata: strengths, weaknesses and generalization*, Proceeding of the 39th IEEE Conference on Foundation of Computer Science, 1998, See also arXiv:quant-ph/9802062 v3, pp. 332–342.

[AGK01] F. Ablayev, A. Gainutdinova, and M. Karpinski, *On computational power of quantum branching programs*, Lecture Notes in Computer Science, no. 2138, Springer-Verlag, 2001, See also arXiv:quant-ph/0302022 v1, pp. 59–70.

[AGKMP] F. Ablayev, A. Gainutdinova, M. Karpinski, C. Moore, and C. Pollette, *On the computational power of probabilistic and quantum branching programs of constant width*, Information and Computation (2005).

[Fre79] R. Freivalds, *Fast probabilistic algorithms*, FCT'79, LNCS 74 (Berlin, New York), Springer-Verlag, 1979, pp. 57–69.

[Gro97] L. K. Grover, *Quantum mechanics helps in searching for a needle in a haystack*, Phys. Rev. Lett. (1997), no. 79, 325–328.

[Høy97] Peter Høyer, *Conjugated operators in quantum algorithms*, Tech. report, University of Southern Denmar, 1997.

[ME99] M. Mosca and A. Ekert, *The hidden subgroup problem and eigenvalue estimation on a quantum computer*, arXive e-print quant-ph/9903071, 1999.

[MT98] Christoph Meinel and Thorsten Theobald, *Algorithms and data structures in VLSI design OBDD - foundations and applications*, Springer-Verlag Berlin Heidelberg, 1998.

[Nag51] T. Nagell, *Introduction to number theory*, New York: Wiley, 1951.

[NC00] Michael A. Nielsen and Isaac L. Chuang, *Quantum computation and quantum information*, Cambridge University Press, 2000.

[NHK00] Masaki Nakanishi, Kiyoharu Hamaguchi, and Toshinobu Kashiwabara, *Ordered quantum branching programs are more powerful than ordered probabilistic branching programs under a bounded-width restriction*, Computing and Combinatorics, LNCS 1858 (Sydney, Australia), 6th Annual International Conference, COCOON 2000, Springer-Verlag, July 2000, pp. 467–476.

[Pap94] Christos H. Papadimitriou, *Computational complexity*, Addison-Wesley, 1994.

[Sho97] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. on Computing (1997), no. 26(5), 1484–1509.

[SS04] M. Sauerhoff and Detlef Sieling, *Quantum branching programs and space-bounded nonuniform quantum complexity*, Theoretical Computer Science (2004), no. 334, 177–225.

[Weg00] I. Wegener, *Branching programs and binary decision diagrams*, SIAM Monographs on Discrete Mathematics and Applications, SIAM Press, 2000.