Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 30 (2010)

Stronger Lower Bounds on Quantum OBDD for the Hidden Subgroup Problem

Airat Khasianov

February 25, 2010

Abstract

We consider the *Hidden Subgroup* in the context of quantum *Ordered Binary Decision Diagrams*. We show several lower bounds for this function. In this paper we also consider a slightly more general definition of the hidden subgroup problem (in contrast to that in [KH10]). It turns out that in this case the problem is intractable for the quantum OBDD. We prove exponential lower bounds for this function.

1 Introduction

Considering one-way quantum finite automata, Ambainis and Freivalds (see [AF98]) suggested that first quantum-mechanical computers would consist of a comparatively simple quantum-mechanical part connected to a classical computer. In this paper we consider another restricted model of quantum-classical computation referred to as *oblivious Ordered Read-Once Quantum Branching Programs*. It is also known as non-uniform automata.

Two models of *quantum branching programs* were introduced by Ablayev, Gainutdinova, Karpinski [AGK01] (*leveled programs*), and by Nakanishi, Hamaguchi, Kashiwabara [NHK00] (*non-leveled programs*). Later it was shown by Sauerhoff [SS04] that these two models are polynomially equivalent.

The *hidden subgroup problem* [ME99], [Høy97] is an important computational problem that has factoring and discrete logarithm as its special cases. Subsequently, an efficient algorithm for the hidden subgroup problem implies efficient solutions for both the *period finding problem*, and original *Simon problem*.

Some results of this paper were originally presented in [KH05].

2 Preliminaries and Definitions

The definition of a *linear branching program* is a generalization of the definition of quantum branching program presented in [AGK01]. Deterministic and quantum oblivious branching programs are particular cases of linear branching programs. Let $\mathbf{V}^d$ be a d-dimensional vector space.

We use $|\psi\rangle$ and $\langle\psi|$ to denote column vectors and row vectors respectively from $\mathbf{V}^d$, and $\langle\psi_1|\psi_2\rangle$ denotes the inner product. We write ψ when it is not important whether it is in column or row form.

ISSN 1433-8092

Definition 1 (Linear branching program). A Linear Branching Program P of width d and length l (a $(d,l)-LBP$) over $\mathbf{V}^d$ is defined as

$$P = \langle T, |\psi_0\rangle, \text{Accept}\rangle$$

where T is a sequence of l instructions: $T_j = \left(x_{i_j}, U_j(0), U_j(1)\right)$ determined by x_{i_j} tested on the step j where $U_j(0)$ and $U_j(1)$ are $d \times d$ matrices.

Vectors $|\psi\rangle \in \mathbf{V}^d$ are called states (state vectors) of P, $|\psi_0\rangle \in \mathbf{V}^d$ is the initial state of P, and Accept $\subseteq \{1, \ldots, d\}$ is the accepting set.

We define a computation of P on an input $\sigma = (\sigma_1, \ldots, \sigma_n) \in \{0,1\}^n$ as follows:

1. A computation of P starts from the initial state $|\psi_0\rangle$;
2. The j^{th} instruction of P queries a variable x_{i_j}, and applies the transition matrix $U_j = U_j(\sigma_{i_j})$ to the current state $|\psi\rangle$ to obtain the state $|\psi'\rangle = U_j(x_{i_j})\,|\psi\rangle$;
3. The final state is
$$|\psi(\sigma)\rangle = \left(\prod_{j=l}^{1} U_j(\sigma_{i_j})\right)|\psi_0\rangle \quad .$$

The usual complexity measures for $(d,l)-LBP$ are its width d, length l, and size $d \cdot l$.

Deterministic branching programs. A *deterministic* branching program is a linear branching program over a vector space $\mathbb{R}^d$. A state $|\psi\rangle$ of such a program is a Boolean vector with exactly one 1. The matrices U_j correspond to permutations of order d, and so have exactly one 1 in each column. For branching programs over groups this is true for the rows as well; in which case, the U_j are permutation matrices.

Quantum branching programs. We define a *quantum* branching program as a linear branching program over a Hilbert space $\mathcal{H}^d$. The $|\psi\rangle$ for such a program are complex state vectors with $\| |\psi\rangle \|_2 = 1$, and the U_j are complex-valued unitary matrices.

After the l^{th} (last) step of quantum transformation P measures its configuration $|\psi_\sigma\rangle$ where $|\psi_\sigma\rangle = U_l(\sigma_{i_l})U_{l-1}(\sigma_{i_{l-1}})\ldots U_1(\sigma_{i_1})\,|\psi_0\rangle$. Measurement is presented by a diagonal zero-one projection matrix M where $M_{ii} = 1$ if $i \in Accept$ and $M_{ii} = 0$ if $i \notin Accept$. The probability $Pr_{accept}(\sigma)$ of P accepting input σ is defined by

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

A QBP P *computes* f *with one-sided error* if there exists an $\varepsilon > 0$ such that for all $\sigma \in f^{-1}(1)$ the probability of P accepting σ is 1 and for all $\sigma \in f^{-1}(0)$ the probability of P accepting σ is less than $1 - \varepsilon$.

Note that this is a "measure-once" model analogous to the model of quantum finite automata in [MC97], in which the system evolves in a unitary manner except for a single measurement at the end.

Read-once branching programs.

Definition 2. We call an LBP P an OBDD or read-once LBP if each variable $x \in \{x_1, \ldots, x_n\}$ occurs in the sequence T of transformations of P at most once.

The "obliviousness" is inherent for an LBP and therefore this definition is consistent with the usual notion of an OBDD. We will use QOBDD for quantum read-once branching programs and OBDD for their deterministic counterparts.

The following general lower bound on the width of QOBDDs is proved in [AGK01].

Theorem 1. *Let $\epsilon \in (0, 1/2)$. Let $f(x_1, \ldots, x_n)$ be a Boolean function $(1/2 + \epsilon)$-computed (computed with margin ϵ) by a quantum read-once branching program Q. Then*

$$\text{width}(Q) = \Omega(\log \text{width}(P))$$

where P is a deterministic OBDD of minimal width computing $f(x_1, \ldots, x_n)$.

We shall reprove it for a slightly different setting in this paper.

3 The Upper Bounds for HSP and several other functions

In this sections we briefly remind several upper bounds obtained earlier. Detailed presentation of these results can be found in [AKV10].

3.1 Equality

Definition 3. $\text{EQ}_n(x, y) \equiv [x = y]$, where n is even, and $x = \{x_1, \ldots, x_{n/2}\}$, $y = \{x_{n/2+1}, \ldots, x_n\}$.

This function is easy in deterministic case for a clever choice of the variable ordering. But for the natural ordering, we consider here, it is exponentially hard.

Theorem 2. *For arbitrary $\epsilon \in (0, 1)$ the function $EQ_n(x, y)$ can be computed with one-sided error ϵ by a 1QBPof width $O(n)$, where $n = |xy|$ is the length of the input.*

Definition 4. *Palindrome*$_n(x_1, \ldots, x_n) \equiv \left[x_1 x_2 \ldots x_{\lfloor n/2 \rfloor} = x_n x_{n-1} \ldots x_{\lceil n/2 \rceil + 1}\right]$

Theorem 3. *For arbitrary $\epsilon \in (0, 1)$ the function Palindrome$_n$ can be computed with constant one-sided error ϵ by a 1QBPof width $O(n)$.*

For a set of input variables $x = \{x_0, \ldots, x_{n-1}\}$, and s – the period parameter, we define the *Periodicity* function $\text{Period}_{s,n}(x)$ that takes the input of length $n + k$, where $n = |x|$, and $k = \lceil \log n \rceil$ – the number of bits needed for s.

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

Theorem 4. *For arbitrary $\epsilon \in (0, 1)$ the function Period$_{s,n}(x)$ can be computed with constant one-sided error ϵ by a 1QBPof width $O(n)$, where $n = |x|$.*

For a set of input variables $x = \{x_0, \ldots, x_{n-1}\}$, and $s \in (0, n]$ we define the *Semi-Simon* function as follows

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

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.

Remark 1. The way we treated binary sequences in the definition above, we should adopt throughout the paper without further notice.

Theorem 5. *For any $\epsilon \in (0,1)$ and all $s \in (0,n]$ the function Semi-Simon$_{s,n}(x)$ can be computed with one-sided error ϵ by a 1QBPof width $O(n)$.*

The *Permutation Matrix* test function ($PERM_n$) is defined on n^2 variables x_{ij} $(1 \le i,j \le n)$. It tests whether the input matrix contains exactly one 1 in each row and each column. Thus, $PERM_n = 1$ if, and only if the input matrix contains exactly one 1 in each row and each column.

Note, that this function cannot be effectively computed by a deterministic OBDD – the lower bound is $\Omega(2^n n^{-5/2})$ regardless of the variable ordering [Weg00]. The width of the best known probabilistic OBDD, computing this function with one-sided error, is $O(n^4 \log n)$ [Weg00]. Our algorithm has the width $O(n \log n)$. Since the lower bound $\Omega(n - \log n)$ follows from Theorem 9, our algorithm is almost optimal.

Theorem 6. *For any $\epsilon \in (0,1)$ the function $PERM_n(x)$ can be computed with one-sided error ϵ by a 1QBPof width $O(n \log n)$.*

Definition 5. Let K be a normal subgroup of a finite group G. Let X be a finite set. For a sequence $\chi \in X^{|G|}$ let $\sigma = bin(\chi)$ be it's representation in binary. If σ encodes no correct sequence $\chi = \chi_1 \dots \chi_{|X|}$, then *Hidden Subgroup* function of σ is set to be zero, otherwise:

$$\mathrm{HSP}_{G,K,X}(\sigma) = \begin{cases} 1, & \text{if } \forall a \in G \forall i,j \in aK(\chi_i = \chi_j) \\ & \text{and } \forall a,b \in G \forall i \in aK \forall j \in bK \\ & (aK \neq bK \Rightarrow \chi_i \neq \chi_j); \\ 0, & \text{otherwise.} \end{cases}$$

Let f be the function encoded by the input sequence. We want to know if a function $f : G \to X$ "hides" the subgroup K in the group G. Our program receives G and K as *parameters*, and function f as an *input string* containing values of f it takes on G. The values are arranged in lexicographical order. See Definition 5.

Remark 2. We make two assumptions. First, we assume that the set X contains exactly $(G:K)$ elements. Indeed, having read the function f, encoded in the input sequence σ, we have X to be the set of all different values that f takes. Obviously, if $|X|$ is less or greater than $(G:K)$, then $\mathrm{HSP}_{G,K,X}(\sigma) = 0$. The second assumption, is that we replace all values of f by numbers from 1 through $(G:K)$. Thus, $\mathrm{HSP}_{G,K,X}(x_1,\dots,x_n)$ is a Boolean function of $n = |G|\lceil \log G : K \rceil$ variables.

In these two assumptions the following theorem holds.

Theorem 7. *Function HSP$_{G,K,X}(x)$ can be computed with one-sided error by a quantum OBDD of width $O(n)$.*

4 General Lower Bound

Most part of this section consists of reformulation and adjusting of some well-known results for the purpose of the proving lower bounds of the $\mathrm{HSP}_{G,K,X}(x)$ function.

Theorem 8. *Let $\epsilon \in (0,1/2)$ be a constant. Let Q_f be a one-way quantum branching program that computes function $f_n \in \mathbb{B}_n$ with one-sided error ϵ. Then for any partition Π of the input, following holds.*

$$width(Q_f) = \Omega(CC_1(f,\Pi)).$$

Proof of the theorem 8 We prove this theorem in two steps. First we show how to relate communication complexity to the width of branching programs. Then we apply the general lower bound theorem to translate this relation to the quantum branching programs.

Lemma 1. *For any deterministic OBDD P representing a Boolean function $f \in \mathbb{B}_n$, let Π be a cut of the input variables, and let k be the cutting point of Π. Then P defines a* two party one-way communication protocol Φ.

$$width\,(P) \geq 2^{CC_1(f)-1},$$

where $CC_1\,(f)$ is the deterministic one-way two-party communication complexity of the function f.

The proof is done in one step. Let $X \in \mathbb{B}^n$ be the input of the program P. For the program P that represents the function f, we define a one-way two-party communication protocol $\langle \Phi, \Pi \rangle$ computing the very same function (See [Hro97]). Let Alice read the variables in $\Pi_{L,X}$. Let Bob read the variables in $\Pi_{R,X}$. The program P is represented by a levelled (See [MT98]) directed graph. Now let us number all the vertices on the k^{th} level of P.

Alice simulates computation of the program P on the first k input variables. Any computation apart from sending messages to the parties is "free of charge" in communication complexity. Let the message c that Alice is supposed to send Bob be the number of the vertex of the kth level, where the path in P defined by the first k input variables ends.

Bob can obviously continue the simulation of P having received the message from Alice. Finally, Bob will output the desired value $f(X)$. The idea is illustrated on the **Figure 1**.

This correctly defines the protocol $\langle \Phi, \Pi \rangle$.

There can not be more than $width\,(P)$ vertices on the k^{th} level of P. Thus $\lfloor \log_2 width\,(P) \rfloor + 1$ bits is enough to send the message s as described in the protocol. **The lemma is evident.**

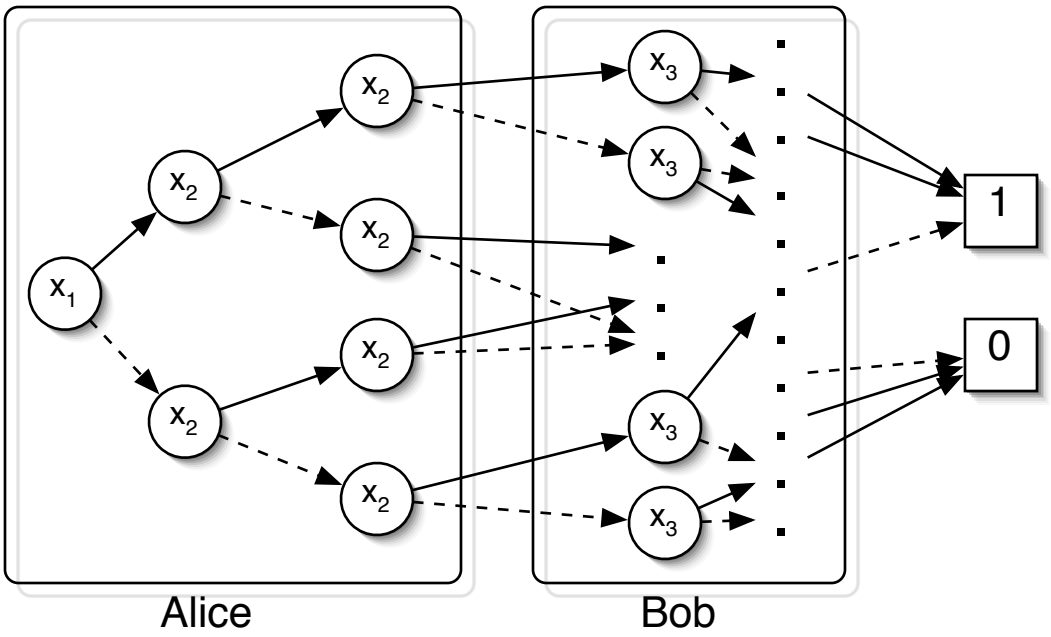

Figure 1: Communication protocol simulating OBDD

Next theorem is needed to prove our result. It is basically a reformulation of the similar theorem from [AGKMP]. The proof of the theorem essentially follows the proof designed by Rabin (Theorem 3 in [RA63]) for the probabilistic automata.

Theorem 9. *Let $\epsilon \in (0, 1/2)$. Let Q be a one-way quantum branching program that computes function $f_n \in \mathbb{B}_n$ with one-sided error ϵ. Then it holds that*

$$width\,(Q) = \Omega\left(\log_2 width\,(P)\right), \tag{1}$$

where P is a deterministic OBDD of minimal width computing $f_n(x_1, \ldots, x_n)$.

Proof. We prove the theorem by constructing for the function f_n a deterministic OBDD based on the structure of the program Q. The construction goes as follows.

Program Q is naturally levelled according to the definition. Assume $k = width\,(Q)$ is the dimension of the vector space of the states of the program Q. The computation process alters the state vector during discrete test of the classical input variables. Thus, we define equivalence classes on the state vectors of the program Q.

Definition 4.1. We call two states $|\psi_1\rangle$ and $|\psi_2\rangle$ equivalent on the level j of the program Q

$$|\psi_1\rangle \equiv_j |\psi_2\rangle \Longleftrightarrow U_j(0)\,|\psi_1\rangle \equiv_j U_j(0)\,|\psi_2\rangle \wedge U_j(1)\,|\psi_1\rangle \equiv_{j+1} U_j(1)\,|\psi_2\rangle\,;$$
$$|\psi_1\rangle \equiv_l |\psi_2\rangle \Longleftrightarrow \|M\,|\psi_1\rangle\,\|^2 = 1 \wedge \|M\,|\psi_1\rangle\,\|^2 = 1 \vee \|M\,|\psi_1\rangle\,\|^2 > \epsilon \wedge \|M\,|\psi_1\rangle\,\|^2 > \epsilon,$$

where

$l := \mathrm{Length}(Q)$,

M is the projection matrix ,

$\|M\,|\psi_1\rangle\,\|^2 = Pr_{accept}(\sigma), \sigma$ being the input of the computation.

Essentially, the two states are equivalent if and only if on any input of the program they lead to the same output.

We introduced a valid equivalence relation. During the computation the state vector transformations are unitary, hence they preserve the distance between any two vectors. Therefore we obtain valid equivalence classes on each level of the program.

Now we build a deterministic OBDD P computing f_n. Deterministic OBDDs are levelled. Each level of our program P would consist of vertices corresponding to the equivalence classes of the program Q on the same level. We might have some redundant nodes in the program P.

Lemma 2. *Let $|\psi_1\rangle$ and $|\psi_2\rangle$ – two state vectors that are not equivalent, then following holds*

$$\||\psi_1\rangle - |\psi_2\rangle\|_2 > 2 - 2\sqrt{\epsilon}$$

Unitary transformations used during the computation preserve l_2 norm, therefore it suffices to prove this lemma for the states after testing last input variable, that is at the last level of the program. That is $|\psi_1\rangle \equiv_l |\psi_2\rangle$, where $l = \mathrm{Length}(Q)$.

The two states can be written as:

$$|\psi_1\rangle = \left|\psi_1^A\right\rangle + \left|\psi_1^B\right\rangle$$
$$|\psi_2\rangle = \left|\psi_2^A\right\rangle + \left|\psi_2^B\right\rangle$$

where

$$\begin{aligned}
\left|\psi_1^A\right\rangle &= M\left|\psi_1\right\rangle \text{ corresponds to the "accepting" projection of } \left|\psi_1\right\rangle \\
\left|\psi_1^R\right\rangle &= (I-M)\left|\psi_1\right\rangle \text{ corresponds to the "rejecting" projection of } \left|\psi_1\right\rangle \\
\left|\psi_2^A\right\rangle &= M\left|\psi_2\right\rangle \text{ corresponds to the "accepting" projection of } \left|\psi_2\right\rangle \\
\left|\psi_2^R\right\rangle &= (I-M)\left|\psi_1\right\rangle \text{ corresponds to the "rejecting" projection of } \left|\psi_2\right\rangle
\end{aligned}$$

Since the two states $|\psi_1\rangle$ and $|\psi_2\rangle$ are not equivalent, w.l.o.g. for the measurement M, defined as in the definition 2 we may assume that

$$\begin{aligned}
\left\|\left|\psi_1^A\right\rangle\right\|_2^2 &= 1, \\
\left\|\left|\psi_1^R\right\rangle\right\|_2^2 &= 0; \\
\left\|\left|\psi_2^A\right\rangle\right\|_2^2 &< \epsilon, \\
\left\|\left|\psi_2^R\right\rangle\right\|_2^2 &> 1-\epsilon.
\end{aligned}$$

Now let's estimate the distance between the two states $|\psi_1\rangle$ and $|\psi_2\rangle$.

$$\begin{aligned}
\||\psi_1\rangle - |\psi_2\rangle\|_2^2 = \left\|\left|\psi_1^A\right\rangle - \left|\psi_2^A\right\rangle\right\|_2^2 + \left\|\left|\psi_1^R\right\rangle - \left|\psi_2^R\right\rangle\right\|_2^2 \geq \\
\left(\left\|\left|\psi_1^A\right\rangle\right\| - \left\|\left|\psi_2^A\right\rangle\right\|\right)^2 + \left(\left\|\left|\psi_1^R\right\rangle\right\| - \left\|\left|\psi_2^R\right\rangle\right\|\right)^2 = \\
\left(1 - \left\|\left|\psi_2^A\right\rangle\right\|\right)^2 + \left\|\left|\psi_2^R\right\rangle\right\|^2 \geq \\
\left(1-\sqrt{\epsilon}\right)^2 + 1 - \epsilon = \\
2 - 2\sqrt{\epsilon}.
\end{aligned}$$

We proved the lemma.

Since the state space is compact and inequivalent states are separated from each other by distance greater than $2-2\sqrt{\epsilon}$, the number of equivalence classes is finite. In fact, we can easily put a bound on that number, that is the width of the deterministic program P we build based upon the quantum program Q. Apparently, $width\,(P)$ is at most the number of $2-2\sqrt{\epsilon}$-radius balls in the $k-1$-dimensional surface of the k-dimensional unit sphere of the state space of the program Q. Denote $\epsilon' := 2-2\sqrt{\epsilon}$.

$$\begin{aligned}
width\,(P) \leq \frac{1}{{\epsilon'}^{k-1}} \Rightarrow \\
\Rightarrow k \geq \frac{\log_2 width\,(P)}{\log_2 \epsilon'} + 1 \Rightarrow \\
\Rightarrow width\,(Q) \in \Omega(\log_2 width\,(P')),
\end{aligned}$$

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

Now suppose Q_f is a quantum OBDD, computing $f \in \mathbb{B}_n$, and P is a deterministic OBDD of minimal width computing f_n.

$$width\,(Q_f) = \Omega\left(\log_2 width\,(P)\right) \qquad \text{lemma 1 implies}$$

$$\Rightarrow width\,(Q_f) = \Omega\left(CC_1\,(f)\right). \quad (2)$$

This proves the statement of the theorem 8.

5 Lower bounds for $\mathrm{HSP}_{G,K,X}(x)$ function

We develop a simple language to be used to formulate our technique of proving the lower bound.

First, we notice that an assignment to the string σ from the definition above is also an assignment to the input variables for any program computing $\mathrm{HSP}_{G,K,X}(\sigma)$. That is, a string $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_{|G|}),\ \sigma_i \in X, i = 1, \ldots, |G|$ defines the input variables set $(\sigma_1, \sigma_2, \ldots, \sigma_{|G|})$ that we shall denote with the same letter σ.

The indices $i = 1, \ldots, |G|$ are in one-to-one correspondence with the group elements of G. We shall further refer to the indices as to the elements of G. Naturally, if we mention a group structure on the set of indices, we mean the group structure of G, and not the structures of the semi-ring $\mathbb{N}$.

In the hidden subgroup problem we have sets of the algebraic structure of the group G. On the other hand, the communication model has its own sets. Namely, for an input σ we consider a partition Π (See [Hro97]), that defines the two sets: $\Pi_{L,\sigma}$ and $\Pi_{R,\sigma}$. In order to keep the two systems of sets separate, yet to be able to make statements containing the input partition and the algebraic properties of G, we present the following definitions.

Definition 5.1. For a group G, its subgroup K, and a set $A \subset G$, we define

$$\mathcal{C}_{G,K}(A) := \{C |\ \exists a \in A (C \text{ is a coset of } K \wedge a \in C)\}$$

A set of cosets that have elements in both of the two disjoint sets is called the set of *common cosets*.

Definition 5.2. For a group G, a subgroup K, and two disjoint sets $A, B \subset G$, define the set of *common cosets*.

$$\mathcal{CC}_{G,K}(A; B) := \{C |\ \exists a \in A, b \in B (C \text{ is a coset of } K \wedge a, b \in C)\}.$$

We can obviously define the set of the *common cosets of a partition* Π of the input σ.

$$\mathcal{CC}_{G,K}(\Pi) := \mathcal{CC}_{G,K}(\Pi_{L,\sigma}; \Pi_{R,\sigma}).$$

We call $\#\mathcal{CC}_{G,K}(A; B) = |\mathcal{CC}_{G,K}(A; B)|$ the *common cosets number* of the sets A and B. Analogously, we say $\#\mathcal{CC}_{G,K}(\Pi)$ is the common coset number of the partition Π.

As well as there are common cosets for a pair of given subsets of the group G, there can be cosets, that are not common for the two subsets.

Definition 5.3. For a group G, a subgroup K and two disjoint sets $A, B \subset G$ define the set of *independent cosets* of B with respect to A.

$$\mathcal{IC}_{G,K,A}(B) := \mathcal{C}_{G,K}(B) \setminus \mathcal{CC}_{G,K}(A; B).$$

For a partition Π of the input σ, *the set of the independent cosets of the partition* Π is defined as follows.

$$\mathcal{IC}_{G,K}(\Pi) := \mathcal{IC}_{G,K,\Pi_{L,X}}(\Pi_{R,X}) \cup \mathcal{IC}_{G,K,\Pi_{R,X}}(\Pi_{L,X}).$$

We call $\#\mathcal{IC}_{G,K,A}(B) = |\mathcal{CC}_{G,K}(\Pi)|$ the *independent cosets number* of the set B with respect to the set A. Analogously, the *independent cosets number* of the partition Π is $\#\mathcal{IC}_{G,K}(\Pi) = |\mathcal{IC}_{G,K}(\Pi)|$.

We shall consider a special kind of partitions that we call *cuts*.

Definition 5.4. A partition Π of the input variables σ is a *cut* if there is an integer k such that for all integer $i < k$ $\sigma_i \in \Pi_{L,X}$ and for all $j \geq k$ $\sigma_j \in \Pi_{R,X}$. We shall call this integer k the *cutting point*.

Finally, we present an exclusively technical definition. Its purpose is to simplify reading of the proof.

Definition 5.5. We say that a coset C *takes a value* x for a string σ if all variables with their indexes in C are assigned the value x.

This is a valid notion for all input strings σ, to which the function $\mathrm{HSP}_{G,K,X}(\sigma)$ assigns the value one. For any given coset, all its member variables are assigned the same input value in any of that strings.

Now we state the communication complexity lower bound for the hidden subgroup function.

Theorem 10. *Let K be a non-trivial subgroup of a finite group G. Let X be any finite set, such that $|X| \geq (G : K)$. For any partition Π, one-way communication complexity according to Π of the hidden subgroup test function is bounded as follows.*

$$CC_1\left(HSP_{G,K,X}(\sigma), \Pi\right) = \Omega\left(\log_2 \binom{|X|}{\#\mathcal{CC}_{G,K}(\Pi)} \#\mathcal{CC}_{G,K}(\Pi)! + \left[\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma}) > 0\right] \log_2 \binom{|X| - \#\mathcal{CC}_{G,K}(\Pi)}{\#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}(\Pi_{L,\sigma})}\right).$$

Proof. Our combinatorial proof relies on the concept of a *communication matrix* (See [Hro97]). It is more elegant to use short notation $CM := CM\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right)$, where it can cause no confusion, what function, and according to what partition, is considered. We shall use common notation CM_i to denote the ith row of the matrix CM.

Other shorthand notations used throughout the proof are presented in the list below.

$$l := |X|; \tag{3}$$

$$d := \#\mathcal{C}_{G,K}(\Pi_{L,\sigma}); \tag{4}$$

$$n := \#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}(\Pi_{L,\sigma}), \qquad \text{notice that } n = (G : K) - d; \tag{5}$$

$$m := \#\mathcal{CC}_{G,K}(\Pi), \qquad \text{notice that } m \leq d \leq l. \tag{6}$$

The function $\mathrm{HSP}_{G,K,X}(\sigma) = 1$ if, and only if, the values of the input variables with indices from the same coset equal but never equal if the variables have indices from different cosets.

Obviously, for all strings δ that fail to satisfy the two conditions, corresponding rows of CM must consist only of zero entries. Let E_Π be the set of all "bad" row indices for a given partition Π.

$$E_\Pi = \left\{\delta \middle| \ \forall \gamma \in X^{|\Pi_{R,\sigma}|} \ \left(\delta \in X^{|\Pi_{L,\sigma}|} \wedge \mathrm{HSP}_{G,K,X}(\delta;\gamma) = 0\right)\right\}. \tag{7}$$

Lemma 3. *Let CM be a communication matrix of $HSP_{G,K,X}(\sigma)$ according to a partition Π of input σ. Let $M(\delta)$ be a sub-matrix of CM that consists only of rows $CM_i, i \notin E_\Pi$, such that i assigns values to all cosets from $\mathcal{CC}_{G,K}(\Pi)$ according to the string $\delta \in X^m$. For strings $\delta_1, \delta_2 \in X^m$ we claim the following.*

$$\delta_1 \neq \delta_2 \Rightarrow M(\delta_1) \cap M(\delta_2) = \emptyset. \tag{8}$$

Let $\delta_1, \delta_2 \in X^m$ be two different strings defining assignments to the variables in $\mathcal{CC}_{G,K}(\Pi)$. Now let $i_1, i_2 \in X^{|\Pi_{L,\sigma}|}$ be row indices such that $CM_{i_1} \in M(\delta_1)$, and $CM_{i_2} \in M(\delta_2)$, clearly $i_1, i_2 \notin E_\Pi$. It is also clear that there is a column index $j \in X^{|\Pi_{R,\sigma}|}$ that assigns the elements of $\mathcal{CC}_{G,K}(\Pi)$ values defined by δ_1 and the rest of the values so that $\mathrm{HSP}_{G,K,X}(\Pi^{-1}(i_1, j)) = 1$. According to the definition of hidden subgroup function, $\delta_1 \neq \delta_2 \Rightarrow \mathrm{HSP}_{G,K,X}(\Pi^{-1}(i_2, j)) = 0$. Thus, $CM_{i_1} \neq CM_{i_2}$. **We proved the lemma.**

In the next lemma we count the number of different rows in a sub-matrix $M(\delta)$ for some $\delta \in X^m$.

Define a set of available assignments to the cosets in $\mathcal{CC}_{G,K}(\Pi)$.

$$\mathcal{W}_n^{l-m}(\Pi) = \big\{\{x_0, \ldots, x_{n-1}\} | \forall i, j, 0 \leq i, j \leq n-1 \tag{9}$$

$$(x_i \in X \setminus \mathcal{CC}_{G,K}(\Pi) \wedge i \neq j \Rightarrow x_i \neq x_j)\big\}; \tag{10}$$

$$|\mathcal{W}_n^{l-m}(\Pi)| = \binom{l-n}{m}. \tag{11}$$

Lemma 4. *Let σ be the input in $X^{|G|}$. Let $\delta \in X^m$ be a string that defines an assignment for the variables in $\Pi_{L,\sigma}$. Let $M(\delta)$ be a sub-matrix of CM as defined in the lemma above. If $\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma}) > 0$ (note that it is not the same as $n > 0$) then for any two row indices $i_1, i_2 \in X^{|\Pi_{L,\sigma}|}$ such that they assign values from different sets from $\mathcal{W}_n^{l-m}(\Pi)$ to the cosets in $\mathcal{CC}_{G,K}(\Pi)$,*

$$CM_{i_1} \neq CM_{i_2}.$$

Let U, V be two different sets in $\mathcal{W}_n^{l-m}(\Pi)$. Let i_1, i_2 be two row indices that correspond to the rows in $M(\delta)$. Assume, i_1 assigns the cosets in $\mathcal{IC}_{G,K}(\Pi)$ values from U, and i_2 assign the cosets in $\mathcal{IC}_{G,K}(\Pi)$ values from V.

Since $U \neq V$ there is a value $x \in X \setminus \mathcal{CC}_{G,K}(\Pi)$, such that $x \in U$ and $x \notin V$. By assumption, $\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma}) > 0$. Consider a column index j that assigns value x to one of its independent cosets, but $\mathrm{HSP}_{G,K,X}(\Pi^{-1}(i_2, j)) = 1$. Such an index j exists, by the definition of $\mathrm{HSP}_{G,K,X}(\sigma)$. Also, by the definition of the hidden subgroup test function, it is clear that $\mathrm{HSP}_{G,K,X}(\Pi^{-1}(i_1, j)) = 0$. **This proves the lemma.**

Let's bring together the results of the two previous lemmas and estimate the number of unequal rows.

There are exactly $\binom{l}{m}m!$ different ways to choose assignments for the cosets in $\mathcal{CC}_{G,K}(\Pi)$. According to the **Lemma 3**, rows, that have indices with different assignments of the values of the common cosets, never equal.

If $\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma}) > 0$, then there are exactly $\binom{l-m}{n}$ ways to assign values to the independent cosets. According to the **Lemma 4**, rows with indices that assign to independent cosets values from different sets from $\mathcal{W}_n^{l-m}(\Pi)$ never equal.

That means there are at least

$$\binom{l}{m}m!\binom{l-m}{n} \tag{12}$$

unequal rows in the communication matrix CM. Recall the following theorem (See e.g. [Hro97]).

Theorem 11. *For a Boolean function f defined over a set X of input variables, for a partition Π of the input X*

$$CC_1(f, \Pi) = \lceil \log_2 NRow(CM(f, \Pi)) \rceil,$$

where $NRow(f)$ is the number of different rows in $CM(f, \Pi)$.

By this theorem

$$CC_1\left(\mathrm{HSP}_{G,K,X}\left(\sigma\right),\Pi\right)=\Omega\left(\log_2\binom{l}{m}m!+\left[\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}\left(\Pi_{R,\sigma}\right)>0\right]\log_2\binom{l-m}{n}\right). \tag{13}$$

Now substitute the values for l, m and n to obtain the statement of the theorem. □

In fact, our lower bound is actually tight. In other words, it almost coincides with the best algorithm that we can construct. That means we could not prove a statement about the lower bound of the communication complexity, according to the considered kind of partitions, any stronger than we already did.

Theorem 12. *Let K be a non-trivial subgroup of a finite group G. Let X be any finite set, such that $|X| \geq (G:K)$. For any partition Π, one-way communication complexity according to Π of the hidden subgroup test function is bounded as follows.*

$$CC_1\left(HSP_{G,K,X}\left(\sigma\right),\Pi\right)=\Theta\left(\log_2\binom{|X|}{\#\mathcal{CC}_{G,K}\left(\Pi\right)}\#\mathcal{CC}_{G,K}\left(\Pi\right)!+\left[\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}\left(\Pi_{R,\sigma}\right)>0\right]\log_2\binom{|X|-\#\mathcal{CC}_{G,K}\left(\Pi\right)}{\#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}\left(\Pi_{L,\sigma}\right)}\right).$$

Proof. The lower bound is already proved in the **Theorem 10**. In order to prove the upper bound we give an informal description of the protocol computing $\mathrm{HSP}_{G,K,X}\left(\sigma\right)$ according to the partition Π.

Let us use the same short-hand notation as in the **Theorem 10**.

The protocol is straightforward. There is just one round of communication. The computer A sends message c_1c_2, if $\mathcal{IC}_{G,K,\Pi_{L,\sigma}}\left(\Pi_{R,\sigma}\right)$ is positive, and sends only c_1 otherwise. The message parts $c_1 \in \{0,1\}^a, c_2 \in \{0,1\}^b$, where

$$a := \log_2\binom{l}{m}m!, \tag{14}$$

$$b := \log_2\binom{l-m}{n}. \tag{15}$$

First part c_1 of the message specifies the sub-matrix corresponding to the assignment of the common cosets values. Second part c_2 corresponds to the assignment of the independent cosets of $\Pi_{L,\sigma}$. The latter part allows the computer B choose values for its independent cosets of $\Pi_{R,\sigma}$ so that they do not coincide with the values of $\Pi_{L,\sigma}$. By the definition of the hidden subgroup function, information sent by A to B is enough to compute the function value for any assignment of $\Pi_{R,\sigma}$.

Note, that we can encode the message sent using a prefix code without loss of the efficiency. □

We have considered communication complexity only according to a fixed partition so far. But our results hold for an arbitrary partition of the input. That is why, one-way communication complexity of the hidden subgroup test function is a direct consequence of the proven results.

Corollary 1. *Let K be a non-trivial subgroup of a finite group G. Let X be any finite set, such that $|X| \geq (G:K)$. Let σ be the input. One-way communication complexity of the hidden subgroup test function is bounded as follows.*

$$CC_1\left(HSP_{G,K,X}\left(\sigma\right)\right) = \Theta\left(\min_{\Pi \in Bal(\sigma)}\left\{\log_2\binom{|X|}{\#\mathcal{CC}_{G,K}\left(\Pi\right)}\#\mathcal{CC}_{G,K}\left(\Pi\right)! + \left[\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}\left(\Pi_{R,\sigma}\right) > 0\right]\log_2\binom{|X| - \#\mathcal{CC}_{G,K}\left(\Pi\right)}{\#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}\left(\Pi_{L,\sigma}\right)}\right\}\right).$$

Proof. By the definition of one way communication complexity (See [Hro97]), and as a consequence of **Theorem 12** the statement follows. □

The bounds, we have proved so far, hold for arbitrary partitions. However, in order to prove the "best" quantum lower bound, we need to find the "worst" partition for the hidden subgroup problem. Let's recall our tactics. Essentially, we prove a lower bound for a *classical* branching program, that we then use to obtain the quantum bounds. According to the classical branching program width definition, it is in our interest to find a cut that corresponds to the maximum width of the levelled branching program of our lower bound. What kind of cuts could be good candidates for this job? With this question in mind, we define a new kind of "balanced" partitions, designed specifically for a given instance of the hidden subgroup problem. We shall also consider a special encoding of the $\mathrm{HSP}_{G,K,X}\left(x\right)$ function, as described in remark 2.

Definition 5.6. Let G be a finite group. Let K be a non-trivial proper subgroup of G. A partition Π of the input σ is called (G,K)-*coset balanced*, if

$$\#\mathcal{C}_{G,K}\left(\Pi_{L,\sigma}\right) = \lfloor (G:K)/2 \rfloor.$$

If Π is a cut, then we call it (G,K)-*coset balanced cut*.

It is not difficult to see that for any finite group G, and for any its non-trivial proper subgroup K, there exists a (G,K)-balanced cut. For this kind of cuts we state the next result.

Theorem 13. *Let $HSP_{G,K,X}\left(x\right)$ be encoded as described in remark 2. Let K be a non-trivial proper subgroup of a finite group G. Let X be any finite set such that $|X| \geq (G:K)$. For any* (G,K)-coset balanced cut Π *of σ, one-way communication complexity of the hidden subgroup function is bounded as follows.*

$$CC_1\left(HSP_{G,K,X}\left(\sigma\right),\Pi\right) = \Omega\left(\#\mathcal{CC}_{G,K}\left(\Pi\right)\log_2|X| + \log_2\#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}\left(\Pi_{L,\sigma}\right)\right).$$

Proof. We shall use the short-hand notation from the proof of **Theorem 10**.

The partition Π is (G,K)-balanced by assumption. It implies that

$$\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}\left(\Pi_{R,\sigma}\right) = (G:K) - \#\mathcal{CC}_{G,K}\left(\Pi\right) - \#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}\left(\Pi_{L,\sigma}\right) = (G:K) - \#\mathcal{C}_{G,K}\left(\Pi_{L,\sigma}\right) \geq (G:K)/2 > 0. \quad (16)$$

According to **Theorem 10**

$$CC_1\left(\mathrm{HSP}_{G,K,X}\left(\sigma\right),\Pi\right) = \Omega\left(\log_2\binom{l}{m}m!\binom{l-m}{n}\right). \quad (17)$$

However, now the function encoding satisfies the conditions of the remark 2. Therefore, the number of sets in $\mathcal{W}_n^{l-m}(\Pi)$ is different, since the lemma 4 fails.

$$\binom{l-n}{m} \geq |\mathcal{W}_n^{l-m}(\Pi)| = n.$$

Indeed, if two sets $V, U \in \mathcal{W}_n^{l-m}(\Pi)$ contain elements that sum up to the same value $s_V = \sum_{x\in V} x = s_U = \sum_{x\in U} x = s$, then two rows with index i assigning the cosets values from V, and index j assigning the cosets values from U will correspond to equal rows: $M_j = M_j$.

However, if the sums are different $s_V \neq s_U$, then there'll be column index $k = \left(\sum_{r=1}^{l} x_r\right) - s_V$ such that $\mathrm{HSP}_{G,K,X}(ik) \neq \mathrm{HSP}_{G,K,X}(jk)$.

Therefore, the communication matrix will have at least $\binom{l}{m}m!n$ different rows, and actually the following is true:

$$CC_1\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right) = \Omega\left(\log_2 \binom{l}{m} m!n\right).$$

$$\binom{l}{m}m! = \frac{l!}{(l-m)!} = l\cdot(l-1)\cdot\ldots\cdot(l-m+1) \geq \frac{l^m}{2^m}, \tag{18}$$

because $m \leq \#\mathcal{C}_{G,K}(\Pi_{L,\sigma}) = \lfloor (G:K)/2 \rfloor$.

$$\log_2 \binom{l}{m} m!n \succeq m\log_2 l - m + \log_2 n \asymp m\log_2 l + \log_2 n \tag{19}$$

$$\tag{20}$$

The statement of the theorem follows after the substitution.

□

Note that, in the conditions of the remark 2, $l = |X| = (G:K)$, subsequently, $d = \lfloor (G:K)/2 \rfloor = \lfloor l/2 \rfloor$ and $n = (G:K) - d = (G:K) - \lfloor l/2 \rfloor = \lceil l/2 \rceil$. It is also clear that $m \leq d = l/2$.

Finally, we obtain the quantum read-once branching program lower bound as a simple corollary.

Corollary 2. *Let $HSP_{G,K,X}(x)$ be encoded as described in remark 2. Let K be a non-trivial subgroup of a finite group G. Let X be any finite set, such that $|X| \geq (G:K)$. Let $\epsilon \in (0, 1/2)$. If for all $\sigma \in \{0,1\}^n$ the function $HSP_{G,K,X}(\sigma)$ is computed with one-sided error ϵ by a 1QBP Q then for any* coset-balanced *partition Π of the input σ*

$$width(Q) = \Omega\left(\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma})\log_2|X| + \log_2 \#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}(\Pi_{L,\sigma})\right).$$

Proof. The statement follows as a corollary of **Theorem 8** and **Theorem 13**. □

A less precise, lower bound can also be shown. This time we make the statement in terms of the group order, or equally, the length of the input string $bin(\sigma)$ (See **Definition 5**). This bound shows that our upper bound (See [AKV10]) is quite tight. In fact, since the upper bound does not depend on the structure of G/K, the lower bound may simply reflect the complexity deviations for different choices of the parameters.

Corollary 3. *Let $HSP_{G,K,X}(x)$ be encoded as described in remark 2. Let G be a finite group. Let K be be a non-trivial proper subgroup of G. Let X be a finite set, such that $|X| \geq (G:K)$. Let $t := (G:1)\log_2|X|$ be the length of the binary input $bin(\sigma)$. Let $\epsilon \in (0,1/2)$. If for all $\sigma \in \{0,1\}^n$ the function $HSP_{G,K,X}(\sigma)$ is ϵ-computed by a 1QBP Q then for any* coset-balanced *partition Π of the input σ*

$$width(Q) =$$

$$CC_1\left(HSP_{G,K,X}(\sigma)\right) = \begin{cases} \Omega(|X|) = \Omega(\log_2 t) & if\ \mathcal{CC}_{G,K}(\Pi) = o((G:1)) \\ \Omega((G:K)\log_2|X|) = \Omega(t) & otherwise \end{cases} \quad (21)$$

Proof. Consider the input $bin(\sigma)$ of the length g. See the $\mathrm{HSP}_{G,K,X}(\sigma)$ definition for details. We want to establish an asymptotic lower bound on the quantum and communication complexity of $\mathrm{HSP}_{G,K,X}(\sigma)$ for $g \to \infty$.

The "hidden subgroup" K is a parameter in the function $\mathrm{HSP}_{G,K,X}(\sigma)$. As such, it has a cardinality $k := (K:1)$, that we assume constant.

We shall again use the short-hand notation from the proof of **Theorem 10**.

From **Theorem 13** it follows that

$$CC_1\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right) = \quad (22)$$

$$\Omega\left(\#\mathcal{IC}_{G,K,\Pi_{L,\sigma}}(\Pi_{R,\sigma})\log_2|X| + \log_2 \#\mathcal{IC}_{G,K,\Pi_{R,\sigma}}(\Pi_{L,\sigma})\right). \quad (23)$$

Consider two cases:

1. Suppose $\mathcal{CC}_{G,K}(\Pi) = m = o((G:1))$. Then by the definition of the coset-balanced partition (See **Definition 5.6**),

$$n + m = \Theta((G:1)) \Rightarrow n = \Theta((G:1)) \Rightarrow \quad (24)$$

$$CC_1\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right) = \Omega\left(o((G:1))\log_2 l + \log_2(G:1)\right) = \Omega(\log_2(G:1)). \quad (25)$$

On the other hand

$$(G:1) = \Theta(|X|), \quad (26)$$

thus

$$t = \Theta((G:1)\log_2(G:1)) \Rightarrow \Omega(\log_2(G:1)) = \Omega(log_2 t - \log_2\log_2 t). \quad (27)$$

Finally,

$$CC_1\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right) = \Omega(\log_2 t). \quad (28)$$

2. Suppose $(G:1) = O(m)$. We directly obtain the desired result:

$$CC_1\left(\mathrm{HSP}_{G,K,X}(\sigma), \Pi\right) = \Omega((G:1)\log_2 l) = \Omega(t). \quad (29)$$

The generalization of the result for the width of the quantum branching program follows from corollary 2. □

6 Stronger lower bound for the general case

We show the exponential lower bound for this function by von Neumann entropy arguments. The von Neumann entropy $S(\sigma)$ of the density matrix $\sigma = \sum_i p_1 |\psi_i\rangle \langle\psi_i|$ is defined as follows.

Definition 6.1. Let $\sigma = \sum_i p_1 |\psi_i\rangle \langle\psi_i|$ is a density matrix defined for the probability distribution $(p_i, |\psi_i\rangle)_i$ of the vectors in finite-dimensional Hilbert space. Assume that $(|e_i\rangle)_i$ is an orthonormal basis of eigenvectors of the density matrix σ, and that λ_i is the eigenvalue corresponding to the eigenvector $|e_i\rangle$. *Then von Neumann entropy* is defined as follows.

$$S(\sigma) = -\sum \lambda_i \log_2 \lambda_i.$$

Note that von Neumann entropy is invariant under unitary transformations. That is, $S(\sigma) = S(U\sigma U^\dagger)$, where U is unitary.

As the inspiration for the result of this section served the following theorem.

Theorem 14. *[SS04] The size of each QOBDD with bounded error for $DISJ_n$ or IP_n is $2^\Omega(n)$.*

We shall need several auxiliary results. The following lemma is due to Nayak [N99].

Lemma 5. *Let σ_0 and σ_1 be density matrices over the finite-dimensional Hilbert space $\mathcal{H}$ and let $\sigma = 1/2(\sigma_0 + \sigma_1)$. Suppose there is a projective measurement $M = (P_0, P_1)$ over $\mathcal{H}$ with results in $\{0,1\}$ such that for $b \in \{0,1\}, Pr[M(\sigma_b) = b] \geq p \geq 1/2$. Then $S(\sigma) \geq (S(\sigma_0) + S(\sigma_1))/2 + (1 - H(p))$*

Theorem 15. *The width of each QOBDD with bounded error for $HSP_{G,K,X}(x)$ is $2^\Omega((G:K))$.*

Proof. Our proof is based on the proof of the theorem 14 presented in [SS04].

Let a quantum OBDD Q with some variable order π computes $\mathrm{HSP}_{G,K,X}(x)$ in its unrestricted form (with no respect to remark 2). Let $p = 1/2 + \epsilon$ be the correct output probability margin of the program Q. We shall generate random inputs according to the following rules. W.l.o.g. we assume that elements of the same coset form a continuous substring in the input sequence.

1. We fix the size of the hidden subgroup to some value.
2. All our generated inputs would have function f of the definition of the $\mathrm{HSP}_{G,K,X}(x)$ be constant on the cosets. That is, only values of the function on different cosets would affect the value of the $\mathrm{HSP}_{G,K,X}(x)$ function on the input sequence.
3. We add elements by cosets.
4. With probability 1/2 we add a new coset to the input, so that the function f value is different from all the values the function takes on previously added cosets. With equal probability added coset would have the same image over f as the previously added coset.
5. We assign the values to the cosets sequentially, starting with 1.

We should use $\sigma(k)$ for the density matrix of the program after having read k coset values. In other words, $(G:K)k$ input variables encoded in binary.

We shall prove that $\S(\sigma(k) \geq (1 - H(p))k$ by induction. The program Q starts in a pure state. Therefore, $\sigma(0) = 0$. For $k \geq 1$ we assume $\S(\sigma(k-1) \geq (1-H(p))(k-1)$. While reading the input, the unitary transformations are applied to the density matrix of the program Q. Now consider the unitary transformations performed after having read the k^{th} coset value.

- The unitary transformation U_k^1 is applied to the density matrix, if the k^{th} coset image is k,
- and U_k^0 is applied, if the value is $k-1$.

Since the two cases appear uniformly at random,

$$\sigma k = \frac{1}{2}\left(U_k^0 \sigma(k-1)(U_k^0)^d agger + U_k^1 \sigma(k-1)(U_k^1)^d agger\right).$$

Consider the i^{th} read coset value. Let all values before and past i^{th} are different, and coincide with the coset index. Then the unitary transformation, performed upon reading these values can be fixed to some U. The output of the whole computation, however depends on the value of the i^{th} coset. If $a_i = i$, then $\mathrm{HSP}_{G,K,X}(a) = 1$, and $\mathrm{HSP}_{G,K,X}(a) = 0$, if $a_i = i-1$, where a_i is the value of the i^{th} coset. Therefore, measurement of the density matrix $\sigma = U\sigma(i)U^d agger$ in the end of the computation yields 1 if If $a_i = i$ with probability at least p by definition of the program Q. By the lemma 5 and the invariance of the von Neumann entropy under unitary transformations the next is true.

$$S(\sigma(k)) = S(\sigma) \geq$$
$$\frac{1}{2}\left(S\left(UU_k^0\sigma(k-1)(U_k^0)^d aggerU^d agger\right) + S\left(UU_k^1\sigma(k-1)(U_k^1)^d aggerU^d agger\right)\right) + 1 - H(p) \geq$$
$$\frac{1}{2}\left(S\left(\sigma(k-1)\right) + S\left(\sigma(k-1)\right)\right).$$

Now, by induction hypothesis it follows

$$S(\sigma(k) \geq (1 - H(p))k.$$

We obtain the lower bound $(1-H(p))(G:K)$ on the von Neumann entropy of the density matrix describing the state of the program Q after having read the whole input. If σ is a density matrix over a finite-dimensional Hilbert space $\mathcal{H}$, then $S(\sigma) \leq \log_2(\dim \mathcal{H})$. Hence, we obtain the lower bound $2^{(1-H(p))(G:K)}$ on the width of Q. □

A simple corollary follows from the theorem we have just proved.

Corollary 4. *Suppose $|K| \in o(n)$, i.g. $|K|$ is constant, then The width of each QOBDD with bounded error for $HSP_{G,K,X}(x)$ is $2^{\Omega}(n)$.*

7 Conclusions

Results, presented in this paper show that our upper bounds [KH05, AKV10] are pretty tight. On the other hand, these results justify our simplification of the $\mathrm{HSP}_{G,K,X}(x)$ function. Without proper conditions this function is intractable for Quantun OBDD.

Acknowledgements We thank colleges Farid Ablayev and Marek Karpinski for fruitful discussions during the time the results presented in this paper were obtained.

References

[AK97] F. Ablayev and M. Karpinski, *On the power of randomized ordered branching programs*, Tech. Report 85181-CS, University of Bonn, 1997, see also Electronic Colloquium on Computational Complexity, TR98-004, (1998), available at http://www.eccc.uni-trier.de/eccc/.

[AKK] F. Ablayev, M. Karpinski, and A. Khasianov, *Complexity of Computing Functions on Quantum Branching Programs*, Manuscript, 2008.

[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).

[AKV10] F. Ablayev, A. Khasyanov, and A. Vasiliev, *On Complexity of Quantum Branching Programs Computing Equality-like Boolean Functions*, ECCC (to appear in 2010).

[AN08] A. Ambainis and N. Nahimovs, *Improved constructions of quantum automata*. Manuscript, 2008, from personal communication.

[AV08] F. Ablayev and A. Vasiliev *On the Computation of Boolean Functions by Quantum Branching Programs via Fingerprinting*, TR08-059, (2008), available at http://www.eccc.uni-trier.de/eccc/.

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

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

[Hro97] Juraj Hromkovič, *Communication Complexity and Parallel Computing*, EATCS Texts in Theoretical Computer Science, Springer-Verlag, Berlin, 1997.

[KH05] A. Khasianov, *Complexity Bounds On Some Fundamental Computational Problems For Quantum Branching Programs*, http://nbn-resolving.de/urn:nbn:de:hbz:5N-05696.

[KH06] A. Khasianov, *Complexity Bounds On Some Fundamental Computational Problems For Quantum Branching Programs*, Manuscript (in Russian), 2006.

[KH10] A. Khasianov, *Lower Bounds on Quantum OBDD for the Hidden Subgroup Problem*, Manuscript 2010.

[MC97] C. Moore and J.P. Crutchfield, *Quantum automata and quantum grammars*. Theoretical Computer Science 237: 275–306, 2000.

[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.

[MR95] R. Motwani, P. Raghavan, *Randomized Algorithms*. Cambridge University Press, 1995.

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

[N99] A. Nayak. Optimal lower bounds for quantum automata and random access codes. In Proc. of 40th FOCS, 369377, 1999. http://arxiv.org/abs/quant-ph/9904093

[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.

[RA63] M. Rabin, Probabilistic automata. *Information and Control* 6: 230–245, 1963.

[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.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de