The Quantum Complexity of Group Testing

Sebastian Dörn
Institut für Theoretische Informatik
Universität Ulm
89069 Ulm, Germany

Thomas Thierauf
Fak. Elektronik und Informatik
HTW Aalen
73430 Aalen, Germany

{sebastian.doern,thomas.thierauf}@uni-ulm.de

Abstract

We present quantum query and time complexity bounds for group testing problems. For a set S and a binary operation on S, we consider the decision problem whether a groupoid, semigroup or quasigroup is a group. Our quantum algorithms for these problems improve the best known classical complexity bounds. We also present upper and lower bounds for testing associativity, distributivity and commutativity.

1 Introduction

Quantum algorithms have the potential to demonstrate that for some problems quantum computation is more efficient than classical computation. A goal of quantum computing is to determine for which problems quantum computers are faster than classical computers. The most important known basic quantum algorithms are Shor's and Grover's algorithm. The first one is Shor's [Sho94] polynomial time quantum algorithm for the factorization of integers. The second one is Grover's search algorithm [Gro96]. Since then, we have seen some generalisations and applications of these two basic quantum techniques. The Shor algorithm has been generalized to a quantum algorithms for the hidden subgroup problem (see e.g. [CEMM98]). Grover's search algorithm can be used for quantum amplitude amplification [BHMT02] and quantum random walk search [Amb04, Sze04, MNRS07]. The application of these quantum search tools is a fast growing area in quantum computing. For example, quantum algorithms have been presented for several problems from computer science (see e.g. [BHT98, BDHHMSW01, Amb04]), graph theory (see e.g. [DHHM04, MSS05, AS06, Doe07a, Doe07b]) and (linear) algebra (see e.g. [MN05, BS06, DT07]).

In this paper we study the quantum complexity of group testing problems. For a set S and a binary operation on S, we consider the decision problem whether a groupoid, semigroup or quasigroup is a group. We also present upper and lower bounds for testing associativity, distributivity and commutativity. In particular, we improve the quantum query complexity for testing if a operation table is associative or a quasigroup from [DT07].

The motivation for studying the query complexity of algebraic problems is twofold. On the one hand side, these are fundamental and basic problems which have many applications in computer science. For example, testing if a black box is a group is very useful in cryptography. On the other hand, we can analyze how powerful are our tools for the construction of lower and upper bounds for the quantum query complexity of these problem. For many problems we can find optimal quantum algorithms by a combination of Grover search, amplitude amplification and quantum walk search. But for some problems this doesn't seem to work. Maybe this can be a motivation for the development of new quantum techniques.

In this paper our input is a operation table for a set S of size $n \times n$. In Section 3 we consider several group problems. Given a groupoid, semigroup or quasigroup S by its operation table, we have to decide whether S is a group. We present lower and upper bounds for the quantum query complexity of these group problems. In particular, we give nearly optimal quantum query algorithms for testing whether a groupoid or quasigroup is a group.

In Section 4 we present several bounds for testing associativity, distributivity and commutativity. For associativity testing we consider the binary operation $\circ : S \times S \to S'$, where $S' \subseteq S$. Dörn and Thierauf [DT07] constructed a quantum query algorithm which is faster than the trivial Grover search over all triples of S for $|S'| < n^{3/8}$. Here we improve the quantum query complexity of their algorithm, such that the algorithm is faster than the Grover search for $|S'| < n^{3/4}$. Moreover we determine the precise quantum query complexity for deciding whether a groupoid, semigroup and monoid is commutative.

2 Preliminaries

2.1 Quantum Query Model

In the query model, the input $x_1, \ldots, x_N$ is contained in a black box or oracle and can be accessed by queries to the black box. As a query we give i as input to the black box and the black box outputs x_i. The goal is to compute a Boolean function $f : \{0,1\}^N \to \{0,1\}$ on the input bits $x = (x_1, \ldots, x_N)$ minimizing the number of queries. The classical version of this model is known as decision tree.

The quantum query model was explicitly introduced by Beals et al. [BBCMW01]. In this model we pay for accessing the oracle, but unlike the classical case, we use the power of quantum parallelism to make queries in superposition. The state of the computation is represented by $|i, b, z\rangle$, where i is the query register, b is the answer register, and z is the working register. A quantum computation with T queries is a sequence of unitary transformations

$$U_0 \to O_x \to U_1 \to O_x \to \ldots \to U_{T-1} \to O_x \to U_T,$$

where each U_j is a unitary transformation that does not depend on the input x, and O_x are query (oracle) transformations. The oracle transformation O_x can be defined as $O_x : |i, b, z\rangle \to |i, b \oplus x_i, z\rangle$. The computations consists of the following three steps:

1. Go into the initial state $|0\rangle$.
2. Apply the transformation $U_T O_x \cdots O_x U_0$.
3. Measure the final state.

The result of the computation is the rightmost bit of the state obtained by the measurement.

The quantum computation determines f with bounded error, if for every x, the probability that the result of the computation equals $f(x_1, \ldots, x_N)$ is at least $1-\epsilon$, for some fixed $\epsilon < 1/2$. In the query model of computation each query adds one to the query complexity of an algorithm, but all other computations are free. The time complexity of the algorithm is usually measured in terms of the total circuit size for the unitary operations U_i.

The quantum query complexity of black box computation has become a great interest in quantum computing. The black box model provides a simple and abstract framework for the construction of quantum algorithms. All quantum algorithms can be formulated in the black box model, we can determine the speed up against classical algorithm, and we can prove lower bounds for the quantum query complexity.

2.2 Tools for Quantum Algorithms

Here, we give three tools for the construction of our quantum algorithms.

Quantum Search. A search problem is a subset $S \subseteq [N]$ of the search space $[N]$. With S we associate its characteristic function $f_S : [N] \to \{0, 1\}$ with $f_S(x) = 1$ if $x \in S$, and 0 otherwise. Any $x \in S$ is called a solution to the search problem. Let $k = |S|$ be the number of solutions of S. It is a well known fact in quantum computing (see [Gro96, BBHT98]), that for $k > 0$, the expected quantum query complexity for finding one solution of S is $O(\sqrt{N/k})$, and for finding all solutions, it is $O(\sqrt{kN})$. Furthermore, whether $k > 0$ can be decided in $O(\sqrt{N})$ quantum queries to f_S. The running time complexity of Grover search is larger than its query complexity by a logarithmic factor.

Amplitude Amplification. The quantum amplitude amplification is a generalization of Grover's search algorithm. Let $\mathcal{A}$ be an algorithm for a problem with small success probability at least ϵ. Classically, we need $\Theta(1/\epsilon)$ repetitions of $\mathcal{A}$ to increase its success probability from ϵ to a constant, for example 2/3. There is a corresponding technique in the quantum case (see [BHMT02]). Let $\mathcal{A}$ be a quantum algorithm with one-sided error and success probability at least ϵ. Then there is a quantum algorithm $\mathcal{B}$ that solves $\mathcal{A}$ with success probability 2/3 by $O(\frac{1}{\sqrt{\epsilon}})$ invocations of $\mathcal{A}$.

Quantum Walk. Quantum walks are the quantum counterpart of Markov chains and random walks. Let $P = (p_{xy})$ be the transition matrix of an ergodic symmetric Markov chain on the state space X. Let $M \subseteq X$ be a set of marked states. Assume that the search algorithms use a data structure D that associates some data $D(x)$ with every state $x \in X$. From $D(x)$, we would like to determine if $x \in M$. When operating on D, we consider the following three types of costs:

- *Setup cost* s: The worst case cost to compute $D(x)$, for $x \in X$.
- *Update cost* u: The worst case cost for transition from x to y, and update $D(x)$ to $D(y)$.
- *Checking cost* c: The worst case cost for checking if $x \in M$ by using $D(x)$.

Theorem 2.1 [MNRS07] *Let $\delta > 0$ be the eigenvalue gap of an ergodic Markov chain P and let $\frac{|M|}{|X|} \geq \epsilon$. Then there is a quantum algorithm that determines if M is empty or finds an element of M with cost*

$$s + \frac{1}{\sqrt{\epsilon}}\left(\frac{1}{\sqrt{\delta}}u + c\right).$$

In the most practical applications (see [Amb04, MSS05]) the quantum walk takes place on the Johnson graph $J(n, r)$, which is defined as follows: the vertices are subsets of $\{1, \ldots, n\}$ of size r and two vertices are connected iff they differ in exactly one number. It is well known, that the spectral gap δ of $J(n, r)$ is $\Theta(1/r)$ for $1 \leq r \leq \frac{n}{2}$.

We apply the quantum walk on the graph categorical product of two Johnson graphs. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs, the *graph categorical product* $G = (V, E) = G_1 \times G_2$ of G_1, G_2 is defined as follows: $V = V_1 \times V_2$, and $((g_1, g_2), (g_1', g_2')) \in E$ iff $(g_1, g_1') \in E_1$ and $(g_2, g_2') \in E_2$.

2.3 Tool for Quantum Query Lower Bounds

In this paper, we use the following special case of a method by Ambainis [Amb02] to prove lower bounds for the quantum query complexity.

Theorem 2.2 [Amb02] *Let $\mathcal{F} = \{f : [n] \times [n] \to [n]\}$ be the set of all possible input function, and $\Phi : \mathcal{F} \to \{0, 1\}$. Let $A, B \subset \mathcal{F}$ such that $\Phi(f) = 1$ and $\Phi(g) = 0$ for all $f \in A$ and $g \in B$. Let $R \subset A \times B$, and m, m', l, l' be numbers such that*

1. *for every $f \in A$, there are at least m different $g \in B$ such that $(f, g) \in R$.*
2. *for every $g \in B$, there are at least m' different $f \in A$ such that $(f, g) \in R$.*
3. *for every $f \in A$ and $x, y \in [n]$, there are at most l different $g \in B$ such that $(f, g) \in R$ and $f(x, y) \neq g(x, y)$.*
4. *for every $g \in B$ and $x, y \in [n]$, there are at most l' different $f \in A$ such that $(f, g) \in R$ and $f(x, y) \neq g(x, y)$.*

Then every bounded-error quantum algorithm that computes Φ has quantum query complexity $\Omega\left(\sqrt{\frac{m \cdot m'}{l \cdot l'}}\right)$.

Let $f, g : [n] \times [n] \to [n]$ be two functions, we define

$$d(f,g) = |\{\, x, y \in [n] \mid f(x,y) \neq g(x,y) \,\}|.$$

In some cases, we consider the special case $A, B \subset \{0,1\}^{n \times n}$ and $(f,g) \in R$ if and only if f and g differ in exactly one position. Then it is $l = l' = 1$, and every bounded-error quantum algorithm that computes f has quantum query complexity of $\Omega\left(\sqrt{m \cdot m'}\right)$.

3 Group Problems

In this section we consider the decision problems whether a groupoid, semigroup or quasigroup S of size n with a binary operation $\circ$ is in fact a group. A *groupoid* is a finite set S with a binary operation $\circ$ represented as operation table. The groupoid is called a *semigroup*, if it is associative. A *monoid* is a semigroup with an identity element. A *quasigroup* is a groupoid, where all equations $a \circ x = b$ and $x \circ a = b$ have unique solutions, and a *loop* is a quasigroup with an identity element.

3.1 Group testing for Groupoids

We consider the problem whether a groupoid $(S, \circ)$ is in fact a group. There is a $O(n^2 \log n)$ deterministic algorithm for this problem by [RS00]. We develop a quantum algorithm that has time complexity $O(n^{\frac{13}{12}} \log^2 n)$. Furthermore, we present an $O(n \log n)$ query algorithm for this problem, that has time complexity $O(n^{3/2} \log n)$ however. The latter algorithm is nearly optimal with respect to the query complexity, as we prove a linear lower bound for this problem.

We need a generalization of a lemma from [RS00].

Definition 3.1 *Let $(S, \circ)$ be a groupoid represented by its operation table T. A row of T is called* cancellative*, if it is a permutation of S.*

Lemma 3.2 [RS00] *Let $\circ$ be cancellative in r rows. If $\circ$ is nonassociative then it has at least $r/4$ nonassociative triples.*

Theorem 3.3 *Whether a groupoid is a group can be decided by a quantum algorithm within $O(n^{\frac{13}{12}} \log^c n)$ expected steps, for some constant c.*

Proof. Let $(S, \circ)$ be a groupoid represented by its operation table T. Note that if S is a group, then every row of A is cancellative. Our first step is to determine whether the operation is associative. To do so, we choose an arbitrary subset A of S of size r. We determine r later. Then we check whether T is cancellative in the rows indexed by A. This is not the case, if we find a row with two equal elements. Hence we can solve this with a Grover search and the element distinctness quantum algorithm by Ambainis [Amb04]. The quantum query complexity of this procedure is $O(\sqrt{r} n^{\frac{2}{3}})$.

If any of the considered rows in not cancellative then we are done. Otherwise we randomly choose three elements $a, b, c \in S$ and check whether $(a \circ b) \circ c \neq$

$a \circ (b \circ c)$. If the operation is not associative, then the probability of finding a nonassociative triple is at least $\frac{r}{4n^3}$ by Lemma 3.2. By using the quantum amplitude amplification we have an $O(n^{\frac{3}{2}}/\sqrt{r})$ quantum query algorithm for finding a nonassociative triple.

If there are no nonassociative triple, then $(S, \circ)$ is a semigroup. Whether this semigroup is a group can be decided with $O(n^{\frac{11}{14}} \log n)$ quantum queries by Theorem 3.6. The expected quantum query complexity of the whole algorithm we get

$$O\left(\sqrt{r}n^{\frac{2}{3}} + \frac{n^{\frac{3}{2}}}{\sqrt{r}} + n^{\frac{11}{14}} \log n\right),$$

which is minimized for $r = n^{\frac{5}{6}}$. Hence the expected time complexity of this algorithm is $O(n^{\frac{13}{12}} \log^c n)$ for a constant c, since the element distinctness procedure has running time of $O(n^{2/3} \log^c n)$. □

We can further improve the query complexity of the problem if we allow a larger running time.

Theorem 3.4 *Whether a groupoid is a group can be decided with $O(n \log n)$ expected quantum queries.*

Proof. Let $(S, \circ)$ be a groupoid represented by its operation table T. A well known fact from algebra is, that if $(S, \circ)$ is a quasigroup, then a random subset $R \subset S$ with $c \log n$ elements is a set of generators with probability at least $1 - \exp(c)$ (see [RS00]). We choose a random subset R of $O(\log n)$ elements of S. Then we check whether R is a generating set of $(S, \circ)$. To do so, let $S_0 = R$. We compute inductively $S_i = S_{i-1} \cup (R \circ S_{i-1})$. This adds at least one element in a step, until we reach some $k \leq n$ such that $S_k = S$. In this case, R is a set of generators. For each element a added to some set S_i, we query the $\log n$ elements $R \circ a$ to look for further elements. In total we query at most the $O(n \log n)$ elements of the $R \times S$ submatrix of T. The quantum time is bounded by $O(n^{3/2} \log n)$.

If R is a set of generators, we have to verify whether the multiplication table is associative. Light observed (see [CP61]) that if R is a set of generators of S, then it suffices to test all triples a, b, c in which b is an element of R. By using Grover search, the quantum query for finding a nonassociative triple (if there is one) is $O(n\sqrt{\log n})$. By Theorem 3.6 we can decide whether this semigroup is a group. The total quantum query complexity of is $O(n \log n)$. □

The upper bound of Theorem 3.4 almost matches the lower bound we have.

Theorem 3.5 *Whether a groupoid is a group requires $\Omega(n)$ quantum queries.*

Proof. We apply the Theorem 2.2. Let A be the operation table T of $\mathbb{Z}_n$ and let $\circ$ be the addition modular n. Then T is a group. The set B consists of all $n \times n$ matrices T', where one entry of T' is modified. Therefore the tables of B forming no groups. The relation R is defined by

$$R = \{\, (T, T') \in (A, B) \mid d(T, T') = 1 \,\}.$$

Then R satisfies that $m = n^2(n-1),\ m' = 1,\ l = n-1$ and $l' = 1$. Therefore the quantum query complexity is $\Omega(n)$. $\square$

3.2 Group testing for Semigroups and Quasigroups

Dörn and Thierauf [DT07] considered the problem whether a finite monoid $(S, \circ)$ is in fact a group. They showed that the problem can be solved with $O(n^{\frac{3}{2}})$ queries by a (classical) randomized algorithm, and with $O(n^{\frac{11}{14}} \log n)$ expected queries by a quantum algorithm.

Now suppose that the input is only known to be a semigroup and we want to decide whether it is in fact a group. To do so, we first search for an identity element and then use the algorithm from [DT07]. To find the identity element, we start by choosing an element a of S and search for an element $e \in S$ such that $a \circ e = a$. Then e is our candidate for the identity element. Recall that we finally want to decide whether S is a group. In this case, the identity element is unique. Hence if our candidate e doesn't work we can safely reject the input, even in the case that S actually has an identity element. To test our candidate e, it suffices to check whether $b \circ e = b$ for all $b \in S$. Obviously the two steps can be done in $O(n)$ queries classically and $O(\sqrt{n})$ quantum queries with Grover search. We summarize the observation:

Theorem 3.6 *Whether a given semigroup is a group can be decided with*

1. $O(n^{\frac{3}{2}})$ *queries by a randomized algorithm.*
2. $O(n^{\frac{11}{14}} \log n)$ *by a quantum query algorithm.*

The result should be contrasted with the following: if we want to decide whether a given semigroup is in fact a monoid, then the best known algorithms make $O(n^2)$ queries classically and $O(n)$ queries in the quantum setting.

Next we assume that the input $(S, \circ)$ is a quasigroup. That is, every row in the operation table is cancellative. To check associativity, we can apply Lemma 3.2 with $r = n$ and the test from the proof of Theorem 3.3. This yields an $O(n)$ quantum query algorithm to test whether the operation is associative, and hence a group. We show that this bound is tight up to constant factor.

Theorem 3.7 *Whether a given quasigroup or a loop is a group can be decided with quantum query complexity* $\Theta(n)$.

Proof. For the lower bound, we apply Theorem 2.2 in connection with an idea of [RS00] for proving an $\Omega(n^2)$ lower bound for this problem in classical computing. The set A consists of the operation table T of the group $(\mathbb{Z}_2^m, +)$, where $+$ is the vector addition modulo 2. Let $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in \mathbb{Z}_2^m$ with $\boldsymbol{a} \neq \boldsymbol{0}$. The set B consists of all operation tables of $(\mathbb{Z}_2^m, \circ)$, where $\circ$ is equal to $+$ except in the following four positions:

1. $\boldsymbol{b} \circ \boldsymbol{c} = \boldsymbol{b} + (\boldsymbol{a} + \boldsymbol{c})$,
2. $\boldsymbol{b} \circ (\boldsymbol{a} + \boldsymbol{c}) = \boldsymbol{b} + \boldsymbol{c}$,
3. $(\boldsymbol{a} + \boldsymbol{b}) \circ \boldsymbol{c} = \boldsymbol{b} + \boldsymbol{c}$,
4. $(\boldsymbol{a} + \boldsymbol{b}) \circ (\boldsymbol{a} + \boldsymbol{c}) = \boldsymbol{a} + \boldsymbol{b} + \boldsymbol{c}$.

All tables of B are quasigroups because the above modifications simply exchange two elements in two rows of the table T, but they are not associative, since

$$\boldsymbol{a} + \boldsymbol{b} = (\boldsymbol{c} \circ (\boldsymbol{a} + \boldsymbol{b})) \circ \boldsymbol{c} \neq \boldsymbol{c} \circ ((\boldsymbol{a} + \boldsymbol{b}) \circ \boldsymbol{c}) = \boldsymbol{b}.$$

The relation R is defined by

$$R = \{\, (T, T') \in (A, B) \mid T' \text{ originates of the above four modifications of } T \,\}.$$

Then R satisfies $m = \Omega(n^3),\ m' = 1,\ l = \Omega(n)$ and $l' = 1$. □

4 Testing Associativity, Distributivity and Commutativity

4.1 The Semigroup Problem

We consider the following semigroup problem. We have given two sets S and $S' \subseteq S$ and a binary operation $\circ : S \times S \to S'$ represented by a table. We denote with n the size of the set S. One has to decide whether S is a semigroup, that is, whether the operation on S is associative.

The complexity of this problem was first considered by Rajagopalan and Schulman [RS00], who gave a randomized algorithm with time complexity of $O(n^2 \log \frac{1}{\delta})$, where δ is the error probability. They also showed a lower bound of $\Omega(n^2)$. The previously best known algorithm was the naive $\Omega(n^3)$-algorithm that checks all triples.

In the quantum setting, one can do a Grover search over all triples $(a, b, c) \in S^3$ and check whether the triple is associative. The quantum query complexity of the search is $O(n^{3/2})$. Dörn and Thierauf [DT07] constructed a quantum query algorithm which is faster than the Grover search for $|S'| < n^{3/8}$. They also proved a quantum query lower bound of $\Omega(n)$. Here we improve the quantum query complexity of their algorithm by a more detailed analysis. Furthermore our algorithm is faster than the Grover search for $|S'| < n^{3/4}$.

Theorem 4.1 *Let $k = n^\alpha$ be the size of S' with $0 < \alpha \leq 1$. The quantum query complexity of the semigroup problem is*

$$\begin{cases} O(n^{\frac{5+\alpha}{4}}), & \textit{for } 0 < \alpha \leq \frac{1}{3}, \\ O(n^{\frac{6+2\alpha}{5}}), & \textit{for } \frac{1}{3} < \alpha \leq \frac{3}{4}, \\ O(n^{\frac{3}{2}}), & \textit{for } \frac{3}{4} < \alpha \leq 1. \end{cases}$$

Proof. We use the quantum walk search scheme of Theorem 2.1. The quantum walk is done on the categorical graph product G_J of two Johnson graphs $J(n, r)$. Let A and B two subsets of S of size r. We will determine r later. We search for a pair $(a, b) \in S^2$, such that a, b are two elements of a nonassociative triple.

Then the marked vertices of G_J correspond to pairs (A, B) with $(A \circ B) \circ S \neq A \circ (B \circ S)$. In every step of the walk, we exchange one row and one column of A and B. The database of our quantum walk is the set

$$D(A, B) = \{ (a, b, a \circ b) \mid a \in A \cup S' \text{ and } b \in B \cup S' \}.$$

Now we compute the quantum query costs for the setup, update and checking. The setup cost for the database $D(A, B)$ is $O((r + k)^2)$ and the update cost is $O(r + k)$. To check whether a pair (A, B) is marked, we have to test if $(A \circ B) \circ S \neq A \circ (B \circ S)$.

Now we claim, that the quantum query cost to check this inequality is $O(\sqrt{nrk})$. Therefore we search for a pair $(b, c) \in B \times S$ with $(A \circ b) \circ c \neq A \circ (b \circ c)$. The computation of $A \circ (b \circ c)$ requires only one query, by using our database, since $(b \circ c) \in S'$. The result is a vector of size r, which we denote by $(y_1, \ldots, y_r)$. The evaluation of $(A \circ b)$ needs no queries by using our database, let $(c_1, \ldots, c_r)$ be the result. This vector consists of at most k different entries $(x_1, \ldots, x_k)$. Now we use Grover's algorithm for searching an $i \in [k]$, such that $x_i \circ c \neq y_j$ for an $j \in [r]$ with $x_i = c_j$. This search can be done in $O(\sqrt{k})$ quantum queries. Therefore, by applying two Grover search subroutines, the checking cost is $O(\sqrt{nrk})$.

The spectral gap of the walk on G_J is $\delta = O(1/r)$ for $1 \leq r \leq \frac{n}{2}$, see [BS06]. If there is a triple (a, b, c) with $(a \circ b) \circ c \neq a \circ (b \circ c)$, then there are at least $\binom{n-1}{r-1}^2$ marked sets (A, B). Therefore we have

$$\epsilon \geq \frac{|M|}{|X|} \geq \left(\frac{\binom{n-1}{r-1}}{\binom{n}{r}} \right)^2 \geq r^2/n^2.$$

Let $r = n^\beta$ for $0 < \beta < 1$. Assuming $r > k$, then the quantum query complexity of the semigroup problem is

$$O\left(r^2 + \frac{n}{r}\left(\sqrt{r} \cdot r + \sqrt{nrk}\right)\right) = O\left(n^{2\beta} + n^{1+\frac{\beta}{2}} + n^{\frac{3+\alpha-\beta}{2}}\right).$$

Now we choose β depending on α such that this expression is minimal. Suppose that $2\beta \leq 1 + \frac{\beta}{2}$, i.e. $\beta \leq \frac{2}{3}$. From the equation $1 + \frac{\beta}{2} = \frac{3+\alpha-\beta}{2}$, we get $\beta = \frac{1+\alpha}{2}$. Then the quantum query complexity of the semigroup problem is $O(n^{\frac{5+\alpha}{4}})$ for $r = n^{\frac{1+\alpha}{2}}$ and $\alpha \leq \frac{1}{3}$. Otherwise if $2\beta > 1 + \frac{\beta}{2}$, i.e. $\beta > \frac{2}{3}$, we get $\beta = \frac{3+\alpha}{5}$ from the equation $2\beta = \frac{3+\alpha-\beta}{2}$. Then the quantum query complexity is $O(n^{\frac{6+2\alpha}{5}})$ for $r = n^{\frac{3+\alpha}{5}}$ and $\alpha > \frac{1}{3}$. If $\alpha > \frac{3}{4}$, the query complexity is bigger than $O(n^{\frac{3}{2}})$, therefore we use Grover search instead of quantum walk search. □

Note that the time complexity of our algorithm is $O(n^{1.5} \log n)$.

4.2 The Distributivity Problem

In the distributivity problem we are given a set S and two binary operations $\oplus : S \times S \to S$ and $\otimes : S \times S \to S$ represented by tables. One has to decide whether $(S, \oplus, \otimes)$ is distributive, i.e. we have to test whether the two equation

$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$ and $(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$ are satisfied. A triple $(a, b, c) \in S^3$ that fulfills both equations is called a *distributive triple*. In classical computing, it is not known whether this problem can be solved in less than cubic time. In the quantum setting, one can do a Grover search over all triples $(a, b, c) \in S^3$ and check whether each triple is distributive. The quantum query complexity of the search is $O(n^{3/2})$. We show a linear lower bound on the query complexity.

Theorem 4.2 *The distributivity problem requires $\Omega(n)$ quantum queries.*

Proof. Let $S = \{0, 1, \ldots, n-1\}$. We apply the Theorem 2.2. The set A consists of all pairs of $n \times n$ matrices $T_\oplus$ and $T_\otimes$, where $T_\otimes$ is the zero-matrix, and the entry at position $(1, 0)$ in $T_\oplus$ is 1, and 0 otherwise. It is easy to see, that the tables of A are distributive, since $x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z) = 0$ for all $x, y, z \in S$. The set B consists of all pairs of $n \times n$ matrices $T'_\oplus$ and $T'_\otimes$, where the entry of position $(1, 0)$ in $T'_\oplus$, and (a, b) in $T'_\otimes$ is 1, for $a, b \in S - \{0, 1\}$, and 0 otherwise. Then $a \otimes (b \oplus c) = 0$ and $(a \otimes b) \oplus (a \otimes c) = 1$ with $b \neq c$. Therefore the tables of B are not distributive.

From each $(T_\oplus, T_\otimes) \in A$, we can obtain $(T'_\oplus, T'_\otimes) \in B$ by replacing the entry 0 of $T_\otimes$ at (a, b) by 1, for any $a, b \notin \{0, 1\}$. Hence we have $m = \Omega(n^2)$. From each $(T'_\oplus, T'_\otimes) \in B$, we can obtain $(T_\oplus, T_\otimes) \in A$, by replacing the entry 1 of $T_\otimes$ at position (a, b) by 0, for $a, b \notin \{0, 1\}$. Thus we have $m' = 1$. By Theorem 2.2, the quantum query complexity is $\Omega(\sqrt{m \cdot m'}) = \Omega(n)$. □

If $(S, \oplus)$ is a commutative quasigroup, then we can get a faster algorithm to check distributivity. The key is that one nondistributive triple implies the existence of more such triples. Similar to Lemma 3.2, we have the following lemma.

Lemma 4.3 *Let S be a set and $\oplus, \otimes$ be two binary operations on S, such that $(S, \oplus)$ is a commutative quasigroup. If $(S, \oplus, \otimes)$ is nondistributive, then it has at least $\Omega(n)$ nondistributive triples.*

Proof. Let (a, b, c) be a nondistributive triple. Let $a = a' \oplus a''$ and consider the following cycle.

$$\begin{aligned}
(a' \oplus a'') \otimes (b \oplus c) &= ((a' \oplus a'') \otimes b) \oplus ((a' \oplus a'') \otimes c) \\
&= ((a' \otimes b) \oplus (a'' \otimes b)) \oplus ((a' \oplus a'') \otimes c) \\
&= (a' \otimes b) \oplus (a'' \otimes b) \oplus (a' \otimes c) \oplus (a'' \otimes c) \\
&= (a' \otimes b) \oplus (a' \otimes c) \oplus (a'' \otimes (b \oplus c)) \\
&= (a' \otimes (b \oplus c)) \oplus (a'' \otimes (b \oplus c)) \\
&= (a' \oplus a'') \otimes (b \oplus c).
\end{aligned}$$

Suppose that $a \otimes (b \oplus c) \neq (a \otimes b) \oplus (a \otimes c)$. Then at least one of the above equations does not hold. Therefore at least one of the following triples must be nondistributive:

$$(a', a'', b), \quad (a', a'', c), \quad (a'', b, c), \quad (a', b, c), \quad (a', a'', b \oplus c).$$

Since $(S, \oplus)$ is a quasigroup, a can be written as $a' \oplus a''$ in n different ways. For each of these, distributivity fails in at least one of the five categories from above. Therefore there exists a category for which there are $\geq n/5$ failures.

The case that $(a \oplus b) \otimes c \neq (a \otimes c) \oplus (b \otimes c)$ can be handled similarly $\square$

By using Lemma 4.3 in combination with the amplitude amplification (similar to Theorem 3.3) we have

Theorem 4.4 *Let $(S, \oplus)$ be a commutative quasigroup and $(S, \otimes)$ a groupoid. Whether $(S, \oplus, \otimes)$ is distributive can be decided with quantum query complexity of $O(n)$.*

4.3 The Commutativity Problem

In the commutativity problem we have given a finite set S of size n with a binary operation $\circ : S \times S \rightarrow S$ represented by a table. One has to decide whether S is a commutative. In the quantum setting, one can solve the problem in linear time by a Grover search over all tuple $(a, b) \in S^2$ that checks whether the tuple is commutative. We show that the commutativity problem requires $\Omega(n)$ quantum queries, even when S is a monoid.

Theorem 4.5 *The quantum query complexity of the commutativity problem for groupoids, semigroups, and monoids is $\Theta(n)$.*

Proof. We start by showing the lower bound for semigroups via Theorem 2.2. Let $S = \{0, 1, \ldots, n-1\}$. The set A consists of the zero matrix of order n. The set B consists of all $n \times n$ matrices, where the entry of position (a, b) is 1, for $a \neq b \in S - \{0, 1\}$, and 0 otherwise. All operation tables of the sets A and B are semigroups. Then we have $m = \Omega(n^2)$, $m' = 1$, and the quantum query lower bound for testing if a given semigroup is commutative is $\Omega(n)$.

We reduce the commutativity problem for semigroups to the commutativity problem for monoids. Let S be a semigroup represented as a operation table. We define a monoid $M = S \cup \{e\}$ with the identity element $e \notin S$, that is, with $a \circ e = e \circ a = a$, for all $a \in S$. Then the semigroup S is commutative iff the monoid M is commutative. $\square$

Magniez and Nayak [MN05] quantize a classical Markov chain for testing the commutativity of a black box group given by the generators. The constructed an $O(k^{2/3} \log k)$ quantum query algorithm, where k is the number of generators of the group. In the case when $(S, \circ)$ is a quasigroup, a random set of $c \log n$ elements will be a set of generators with probability at least $1 - \exp(c)$ [RS00]. Therefore we obtain the following result:

Theorem 4.6 *Whether a quasigroup, loop or group is commutative can be decided with quantum query complexity $O((\log n)^{\frac{2}{3}} \log \log n)$.*

Conclusions

The table below summarizes the quantum query complexity (**QQC**) and the quantum time complexity (**QTC**) of the algebraic problems considered in the paper. Some of these results are proved in [DT07]. It remains open to close the gaps between the upper and the lower bounds where they don't match.

Problem	**Description**	**QQC**	**QTC**
Semigroup	Decide if $S \times S \to S'$ is a semigroup for constant size of S'.	$\Omega(n)$ $O(n^{\frac{5}{4}})$	$O(n^{\frac{3}{2}} \log n)$
Identity	Decide if a groupoid has an identity element.	$\Theta(n)$	$O(n \log n)$
Quasigroup	Decide if a groupoid is a quasigroup.	$\Omega(n)$ $O(n^{\frac{7}{6}})$	$O(n^{\frac{7}{6}} \log n)$
Group I	Decide if a groupoid is a group.	$\Omega(n)$ $O(n \log n)$	$O(n^{\frac{13}{12}} \log^c n)$
Group II	Decide if a semigroup is a group.	$O(n^{\frac{11}{14}} \log n)$	$O(n^{\frac{11}{14}} \log^c n)$
Group III	Decide if a quasigroup is a group.	$\Theta(n)$	$O(n \log n)$
Group Commut. I	Decide if a groupoid/ semigroup/monoid is commutative.	$\Theta(n)$	$O(n \log n)$
Group Commut. II	Decide if a quasigroup/group is commutative.	$\widetilde{O}((\log n)^{\frac{2}{3}})$	$\widetilde{O}((\log n)^{\frac{2}{3}})$

In particular, we are interested in the following questions:

1. Is there a quantum algorithm for the semigroup problem which is better than $O(n^{1.5})$ for $|S'| = n$?
2. Is there are a classical or a quantum algorithm for the distributivity problem which is faster than the trivial bounds of $O(n^3)$ resp. $O(n^{1.5})$?
3. Can we prove nontrivial lower bounds for the decision problem whether a semigroup or monoid is a group?

References

[Amb02] A. Ambainis, *Quantum Lower Bounds by Quantum Arguments*, Journal of Computer and System Sciences 64: pages 750-767, 2002.

[Amb04] A. Ambainis, *Quantum walk algorithm for element distinctness*, Proceedings of FOCS'04: pages 22-31, 2004.

[AS06] A. Ambainis, R. Špalek, *Quantum Algorithms for Matching and Network Flows*, Proceedings of STACS'06: pages 172-183, 2006.

[BBCMW01] R. Beals, H. Buhrman, R. Cleve, M. Mosca, R. de Wolf, *Quantum lower bounds by polynomials*, Journal of ACM 48: pages 778-797, 2001.

[BBHT98] M. Boyer, G. Brassard, P. Høyer, A. Tapp, *Tight bounds on quantum searching*, Fortschritte Der Physik 46(4-5): pages 493-505, 1998.

[BDHHMSW01] H. Buhrman, C. Dürr, M Heiligman, P. Høyer, F. Magniez, M. Santha, R. de Wolf, *Quantum Algorithms for Element Distinctness*, Proceedings of CCC'01: pages 131-137, 2001.

[BHMT02] G. Brassard, P. Hóyer, M. Mosca, A. Tapp, *Quantum amplitude amplification and estimation*, AMS Contemporary Mathematics, Vol. 305: pages 53-74, 2002.

[BHT98] G. Brassard, P. Hóyer, A. Tapp, *Quantum Cryptanalysis of Hash and Claw-Free Functions*, Proceedings of LATIN'98: pages 163-169, 1998.

[BS06] H. Buhrman, R. Špalek, *Quantum Verification of Matrix Products*, Proceedings of SODA'06: pages 880-889, 2006.

[CEMM98] R. Cleve, A. Ekert, C. Macchiavello, M. Mosca, *Quantum algorithms revisited*, Proceedings of the Royal Society of London, Series A: pages 339-354, 1998. pages 339-354, 1998.

[CP61] A.H. Clifford, G.B. Preston, *The Algebraic Theory of Semigroups*, American Mathematical Society, 1961.

[Doe07a] S. Dörn, *Quantum Complexity Bounds of Independent Set Problems*, Proceedings of SOFSEM'07 (SRF): pages 25-36, 2007.

[Doe07b] S. Dörn, *Quantum Algorithms for Graph Traversals and Related Problems*, Proceedings of CIE'07: pages 123-131, 2007.

[DT07] S. Dörn, T. Thierauf, *The Quantum Query Complexity of Algebraic Properties*, Proceedings of FCT'07: pages 250-260, 2007.

[DHHM04] C. Dürr, M. Heiligman, P. Høyer, M. Mhalla, *Quantum query complexity of some graph problems*, Proceedings of ICALP'04: pages 481-493, 2004.

[Gro96] L. Grover, *A fast mechanical algorithm for database search*, Proceedings of STOC'96: pages 212-219, 1996.

[MN05] F. Magniez, A. Nayak, *Quantum complexity of testing group commutativity*, Proceedings of ICALP'05: pages 1312-1324, 2005.

[MNRS07] F. Magniez, A. Nayak, J. Roland, M. Santha, *Search via Quantum Walk*, Proceedings of STOC'07: pages: 575-584, 2007.

[MSS05] F. Magniez, M. Santha, M. Szegedy, *Quantum Algorithms for the Triangle Problem*, Proceedings of SODA'05: pages 1109-1117, 2005.

[RS00] S. Rajagopalan, L. J. Schulman, *Verification of identities*, SIAM J. Computing 29(4): pages 1155-1163, 2000.

[Sho94] P. Shor, *Algorithms for quantum computation: discrete logarithms and factoring*, Proceedings of FOCS'94: pages 124-134, 1994.

[Sze04] M. Szegedy, *Quantum speed-up of Markov chain based algorithms*, Proceedings of FOCS'04: pages 32-41, 2004.