Quantum Speedup and Limitations on Matroid Properties

Xiaowei Huang
Institute of Quantum Computing and Computer Theory, School of Computer and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Jingquan Luo
Institute of Quantum Computing and Computer Theory, School of Computer and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Lvzhou Li
Institute of Quantum Computing and Computer Theory, School of Computer and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Abstract

In this paper, we study for the first time how well quantum algorithms perform on matroid properties. We show that quadratic quantum speedup is possible for the calculation problem of finding the girth or the number of circuits (bases, flats, hyperplanes) of a matroid, and the decision problem of deciding whether a matroid is uniform or Eulerian, by giving a uniform lower bound \( \Omega(\sqrt{n \lfloor n/2 \rfloor}) \) on the query complexity for all these problems. On the other hand, for the uniform matroid decision problem, we present an asymptotically optimal quantum algorithm which achieves the lower bound, and for the girth problem, we give an almost optimal quantum algorithm with query complexity \( O(\log n \sqrt{n \lfloor n/2 \rfloor}) \). In addition, for the paving matroid decision problem, we prove a lower bound \( \Omega(\sqrt{n \lfloor n/2 \rfloor}/n) \) on the query complexity, and give an \( O(\sqrt{n \lfloor n/2 \rfloor}) \) quantum algorithm.

2012 ACM Subject Classification Theory of computation \( \rightarrow \) Computational complexity; Theory of computation \( \rightarrow \) Quantum complexity theory; Theory of computation \( \rightarrow \) Quantum query complexity; Mathematics of computing \( \rightarrow \) Matroids and greedoid

Keywords and phrases Combinatorial, Matroid, Quantum Computing, Complexity

Digital Object Identifier 10.4230/LIPIcs...

Funding Xiaowei Huang: [funding] Jingquan Luo: [funding] Lvzhou Li: [funding]

1 Introduction

The idea of quantum computing can be traced back to the point in 1981 that Feynman observed that it is almost impossible to effectively simulate the evolution of a general quantum system on a classical computer and the computational model based on quantum mechanics principles should be used [10]. In 1985, Deutsch showed that there exists a problem for which quantum computers can solve more efficiently than classical computers [7]. Since then, determining when quantum computers can provide a computational speedup over classical ones and finding quantum algorithms with speedup are one of the core issues in the quantum computing filed. Compared with a large number of problems in reality, currently few problems have been discovered for which quantum algorithms show speedup over their classical counterparts. Many of these problems, including the ones solved by Deutsch-Jozsa

1 L. Li is the corresponding author (lilvzh@mail.sysu.edu.cn)
Algorithm [8], Simon’s Algorithm [22] and Shor’s Algorithm [21], can be regarded as instances of the hidden subgroup problem discovered by Kitaev [14]. This is similar to the well-known theorem in greedy algorithms: an optimization problem in an independent system can be efficiently solved by a greedy algorithm if and only if the independent system is a matroid [9].

The concept of matroids was originally introduced by Whitney [23] in 1935 as a generalization of linear algebra and graph theory. After eighty years of development, the matroid theory has become an important branch of mathematics. It has already become an effective mathematical tool to study other mathematic branches and has many applications in geometry, topology, network theory and coding theory, especially in combinatorial optimization [9, 16, 2, 12]. A matroid is a tuple \( M = (V, I) \) where \( V \) is a finite ground set and \( I \subseteq 2^V \) satisfies three conditions (See Definition 1). A matroid property \( \Pi \) is a mapping from the set of all matroids to some set (the set of truth values, the set of integers, etc.) such that any two isomorphic matroids have the same image under this mapping. In this paper, we always consider these matroid property problems (i.e. “Is \( M \) uniform?” or “Find the girth of \( M \)” etc.). Given a finite set \( V \) with \( |V| = n \), the number of nonisomorphic matroids on \( V \) is at least \( 2^{2^{n - (3/2) \log n + \Omega(\log \log n)}} \) [15]. It is easy to see that no matter which way is chosen to represent a matroid, the size of the input for a matroid property problem on an \( n \)-set will be \( O(2^n) \).

The matroid property problems have been studied for a long time. Note that there are several different definitions for matroids but they are equivalent to each other. Different oracles are derived from these different definitions, and matroids are usually assumed to be accessed through an oracle. One oracle often used is the independence oracle \( O_i \): given a matroid \( M = (V, I) \), for a subset \( S \subseteq V \), \( O_i(S) = 1 \) iff \( S \in I \). In the early 1980s, Robinson and Welsh [20] studied the computational complexity of some fundamental matroid properties in different oracles, and showed that the \( \lfloor n/2 \rfloor \)-UNIFORM property is a hardest property with respect to the independence oracle and its lower bound is \( \Omega(\binom{n}{\lfloor n/2 \rfloor}) \) where \( n \) is the size of the ground set. Jensen and Korte [13] showed the query complexities of a series of matroid property problems (from 1 to 8 in Table 1) with respect to the independence oracle are exponential, which thus implies that no polynomial-time algorithm exists for these problems.

1.1 Our contributions.

As far as we know, there has been no work considering quantum algorithms for problems in matroid theory. Perhaps this is a potential field to find quantum advantage. Thus, in this paper we try to explore the possibility of quantum speedup on matroid properties. Assuming that a matroid can only be accessed through the independence query oracle, we study how well quantum query algorithms perform on the matroid property problems (in Table 1). Our main results are as follows.

1. For the calculation problem of finding the girth or the number of circuits (bases, flats, hyperplanes), a quantum algorithm has to query the independence oracle at least \( \Omega(\sqrt{\binom{n}{\lfloor n/2 \rfloor}}) \) times (see Theorem 17).
2. For the decision problem of deciding whether a matroid is uniform or Eulerian, a quantum algorithm has to query the independence oracle at least \( \Omega(\sqrt{\binom{n}{\lfloor n/2 \rfloor}}) \) times (see Theorem 18).
3. For the uniform matroid decision problem, there is an \( O(\sqrt{\binom{n}{\lfloor n/2 \rfloor}}) \) quantum algorithm which is asymptotically optimal (see Theorem 19), and for the problem of finding the
girth, there is an $O(\log n \sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ quantum algorithm which is almost optimal (see Theorem 20).

4. For the paving matroid decision problem, there is a quantum algorithm using $O(\sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ queries (see Theorem 21) and any quantum algorithm has to call at least $\Omega(\sqrt{\binom{n}{\lfloor n/2 \rfloor}}/n)$ queries (see Theorem 22).

**Table 1** The query complexity of matroid properties, where the quantum bounds are obtained in this paper.

| Type             | Matroid Property Problems                        | CL               | QL               | QU               |
|------------------|-------------------------------------------------|------------------|------------------|------------------|
| Calculation Problems | 1. Find the girth of $M$.                         | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $O(\log n \sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ |
|                  | 2. Find the number of circuits of $M$.            | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | —                |
|                  | 3. Find the number of bases of $M$.               | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | —                |
|                  | 4. Find the number of flats of $M$.              | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | —                |
|                  | 5. Find the number of hyperplanes of $M$.         | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | —                |
| Decision Problems | 6. Is $M$ an uniform matroid?                    | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $O(\binom{n}{\lfloor n/2 \rfloor})$ |
|                  | 7. Is $M$ an Eulerian matroid?                   | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}\right)$ | —                |
|                  | 8. Is $M$ a paving matroid?                      | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}/n\right)$ | $\Omega\left(\binom{n}{\lfloor n/2 \rfloor}/\sqrt{n}\right)$ | $O(\binom{n}{\lfloor n/2 \rfloor})$ |
|                  | 9. Is $M$ a trivial matroid?                     | $\Omega(n)$      | $\Omega(\sqrt{n})$ | $O(\sqrt{n})$    |
|                  | 10. Is $M$ a loopless matroid?                   | —                | $\Omega(\sqrt{n})$ | —                |

*) CL: Classical Lower Bound. QL: Quantum Lower Bound. QU: Quantum Upper Bound.

The remainder of this paper is organized as follows. In Section 2, some basic concepts in matroid theory and quantum query model are introduced. In Section 3, we obtain lower bounds for some matroid properties problems and present quantum algorithms for some of these problems. Section 4 concludes this paper and presents some further problems.

## 2 preliminaries

### 2.1 Matroid Theory

In this subsection, we will give some basic definitions used in this paper. Since the matroid theory was established as a generalization of linear algebra and graph theory, many concepts in matroid theory are derived from these two disciplines. So if one is familiar with linear algebra and graph theory, it will be helpful for understanding the following definitions. For more details in matroid theory, see [19].

**Definition 1 (Matroid).** A matroid is a combinational object defined by the tuple $\mathcal{M} = (V, \mathcal{I})$ for finite set $V$ and $\mathcal{I} \subseteq 2^V$ such that the following properties hold:

10. $\emptyset \in \mathcal{I}$;
11. If $A' \subseteq A$ and $A \in \mathcal{I}$, then $A' \in \mathcal{I}$;
12. For any two sets $A, B \in \mathcal{I}$ with $|A| < |B|$, there exist an element $v \in B - A$ such that $A \cup \{v\} \in \mathcal{I}$.
Definition 2 (Independent Set). For a matroid $\mathcal{M} = (V, \mathcal{I})$, we call $S \subseteq V$ independent if $S \in \mathcal{I}$ and dependent otherwise.

Definition 3 (Circuit). For a matroid $\mathcal{M} = (V, \mathcal{I})$, if a dependent set $C \subseteq V$ satisfies that for any $e \in C$, $C - e \in \mathcal{I}$, we call $C$ a circuit of $\mathcal{M}$. If $C = \{e\}$ is a circuit, we call $e$ be a loop. If $C = \{e_1, e_2\}$ is a circuit, we call $e_1, e_2$ are parallel.

Definition 4 (Girth). For a matroid $\mathcal{M}$, the girth $g(\mathcal{M})$ of $\mathcal{M}$ is the minimum circuit size of $\mathcal{M}$ unless $\mathcal{M}$ has no circuits, in which case, $g(\mathcal{M}) = \infty$.

Definition 5 (Rank). For a matroid $\mathcal{M} = (V, \mathcal{I})$, we define the rank of $\mathcal{M}$ as $\text{rank}(\mathcal{M}) = \max_{S \in \mathcal{I}} |S|$. Further, for any $S \subseteq V$ we define $\text{rank}_\mathcal{M}(S) = \max_{T \subseteq S \in \mathcal{I}} |T|$, for simplicity we use $\text{rank}(S)$ or $r(S)$ in place of $\text{rank}_\mathcal{M}(S)$ if there is no unambiguous in the context.

Definition 6 (Base). For a matroid $\mathcal{M} = (V, \mathcal{I})$, if $B \in \mathcal{I}$ such that $\text{rank}(B) = \text{rank}(\mathcal{M})$, we call $B$ a base of $\mathcal{M}$. By the matroid’s property (I2), we can see that if $B_1$ and $B_2$ are two distinct bases of $\mathcal{M}$, then $|B_1| = |B_2|$. Furthermore, if $e$ is any element of $B_1$, then there is an element $f \in B_2$ such that $(B_1 - \{e\}) \cup \{f\}$ is also a base. A matroid can also be defined by a set of bases, which is equivalent to Definition 7.

Definition 7 (Free Matroid and Trivial Matroid). For a matroid $\mathcal{M} = (V, \mathcal{I})$, $\mathcal{M}$ is called a free matroid if $V$ is the only base and a trivial matroid if $\emptyset$ is the only base.

Definition 8 (Loopless Matroid). For a matroid $\mathcal{M} = (V, \mathcal{I})$, $\mathcal{M}$ is called a loopless matroid if all the singleton in $V$ are independent. In other words, $\mathcal{M}$ does not have any circuit with size 1.

Definition 9 (Uniform Matroid). For a matroid $\mathcal{M} = (V, \mathcal{I})$, let $|V| = n$. If there exists an integer $r$ with $0 \leq r \leq n$ such that $\mathcal{I} = \{S \subseteq V : |S| \leq r\}$, $\mathcal{M}$ is called an uniform matroid of rank $r$ and denoted by $U_{r,n}$.

Definition 10 (Paving Matroid). For a matroid $\mathcal{M} = (V, \mathcal{I})$, if any circuit $C$ of $\mathcal{M}$ satisfies that $|C| \geq \text{rank}(\mathcal{M})$, we call $\mathcal{M}$ a paving matroid. Obviously, an uniform matroid is also a paving matroid.

Definition 11 (Closure). Given a matroid $\mathcal{M} = (V, \mathcal{I})$ on ground set $V$ with rank function $r$. Let $\text{cl}$ be the function from $2^V$ to $2^V$ defined, for all $X \subseteq V$, by $\text{cl}(X) = \{x \in V : r(X \cup x) = r(X)\}$. This function is called the closure operator of $\mathcal{M}$, and we call $\text{cl}(X)$ the closure or span of $X$ in $\mathcal{M}$.

Definition 12 (Flat and Hyperplane). Given a matroid $\mathcal{M} = (V, \mathcal{I})$ on ground set $V$ and its closure operator $\text{cl}$, a subset $X$ of $V$ for which $\text{cl}(X) = X$ is called a flat or a closed set of $\mathcal{M}$. A hyperplane of $\mathcal{M}$ is a flat of rank $r(\mathcal{M}) - 1$. A subset $X$ of $V$ is a spanning set of $\mathcal{M}$ if $\text{cl}(X) = V$. We also say that $X$ spans a subset $Y$ of $V$ if $Y \subseteq \text{cl}(X)$.

Definition 13 (Isomorphic Matroids). Two matroids $\mathcal{M}_1 = (V_1, \mathcal{I}_1)$ and $\mathcal{M}_2 = (V_2, \mathcal{I}_2)$ are isomorphic, written $\mathcal{M}_1 \cong \mathcal{M}_2$, if there is a bijection $\psi$ from $V_1$ to $V_2$ such that, for all $X \subseteq V_1$, the set $\psi(X)$ is independent in $\mathcal{M}_2$ if and only if $X$ is independent in $\mathcal{M}_1$. We call such a bijection $\psi$ an isomorphism from $\mathcal{M}_1$ to $\mathcal{M}_2$.

Definition 14 (Eulerian Matroid). For a matroid $\mathcal{M} = (V, \mathcal{I})$, it is called an Eulerian matroid if there exist disjoint circuits $C_1, \cdots, C_p$ such that $V = C_1 \cup \cdots \cup C_p$. 
Notations. We will always denote by $\mathcal{M}$ a matroid $(V, \mathcal{I})$ on a finite ground set $V$ with $\mathcal{I} \subseteq 2^V$. Given a matroid $\mathcal{M}$, we will denote the ground set and the set of independent sets of $\mathcal{M}$ by $V(\mathcal{M})$ and $\mathcal{I}(\mathcal{M})$, respectively. Similarly, $\mathcal{C}(\mathcal{M})$, $\mathcal{B}(\mathcal{M})$, $\mathcal{F}(\mathcal{M})$, $\mathcal{H}(\mathcal{M})$, is the set of circuits, bases, flats, hyperplanes of $\mathcal{M}$, respectively. $\text{cl}_\mathcal{M}$ is the closure operator of $\mathcal{M}$. A set having $r$ elements will be called a $r$-set. $[n]$ denotes $\{1, 2, \cdots , n\}$. Given a ground set $V$ with $|V| = n$, $A \subseteq V$ and an integer $1 \leq r \leq n$, $J_r = \{J \subseteq V : |J| = r\}$, $A_e = A \cup \{e\}$ for any $e \in V - A$, $J_A = \{A_e : e \in V - A\}$.

Matroid Representation. For the convenience of in the following text, we use a $2^n$-bit $0 - 1$ string to represent a matroid on the ground set $V$ with $|V| = n$ and each bit represents a subset of $V$. A matroid $\mathcal{M}$ on the ground set $V$, denoted by $x(\mathcal{M})$ with $x(\mathcal{M}) \in \{0, 1\}^{2^n}$. Similarly, we use $\mathcal{M}(x)$ to denote the matroid determined by $x \in \{0, 1\}^{2^n}$ if $x$ encodes a matroid. For any $i \in [2^n]$, $x_i = 1$ indicates that the subset corresponding to the $i$-th bit is an independent set of $\mathcal{M}$, otherwise $x_i = 0$. If we know the rank $r$ of a matroid on the ground set $V$, we usually use a $(r^n)$-bit $0 - 1$ string to represents the matroid, each bit representing a $r$-set which is set to 1 when the $r$-set is a base of the matroid, otherwise is set to 0. Given a 0-1 string, which represents a matroid, it is easy to know it is a subset representation or a $r$-set representation from the context.

Matroid Query Oracles. Given a matroid $M = (V, \mathcal{I})$, there are several query oracles by which we access the matroid: independence query oracle $O_i$, rank query oracle $O_r$, circuit query oracle $O_c$, base query oracle $O_b$, closure query oracle $O_c$, etc. Give a set $S \subseteq V$, $O_i(S) = 1$ if $i \in S$; $O_r(S) = 1$ if $S$ is a circuit; $O_c(S)$ is set to 1 when the $S$ is a closure; $O_b(S)$ is set to 1 when the $S$ is a base; $O_c(S)$ is set to 1 when the $S$ is a closed. Throughout this paper, we use two basic tools: (i) Grover’s algorithm and (ii) Ambainis’s quantum adversary method.

2.2 Quantum Computation

For the basic concepts and notations on quantum computing, we refer the reader to the textbook by Nielsen and Chuang [18]. Throughout this paper, we use two basic tools: (i) Grover’s algorithm and (ii) Ambainis’s quantum adversary method.

2.2.1 Quantum Query Model

In the quantum query model [3], the input bits of a boolean function $f : \{0, 1\}^N \rightarrow \{0, 1\}$ can be accessed by queries to an oracle $O$. We use $O_x$ to denote the query transformation corresponding to an input $x = (x_1, \cdots , x_N)$. Given $i \in [N]$ to the oracle $O_x$, it returns $x_i$.

A quantum computation with $T$ queries is a sequence of unitary transformations

$$U_0 \rightarrow O_x \rightarrow U_1 \rightarrow O_x \rightarrow \cdots \rightarrow U_{T-1} \rightarrow O_x \rightarrow U_T$$

where $U_j$ can be any unitary transformations that do not depend on the input $x = x_1 \cdots x_N$. $O_x$ are query (oracle) transformations. The oracle $O_x$ can be defined as $O_x : |i, b, z\rangle \rightarrow |i, b \oplus x_i, z\rangle$, where $\oplus$ is exclusive or operation. Also, we can define $O_x$ as $O_x : |i, b, z\rangle \rightarrow (-1)^{b \cdot x_i}|i, b, z\rangle$, where $i$ is the query register, $b$ is the answer register, and $z$ is the working register. These two definitions of $O_x$ are equivalent: one query of one type can be simulated by one query of the other type. The quantum computation are the following three steps:

1: Prepare the initial state to $|0\rangle$.
2: Then apply $U_0, O_x, \cdots , O_x, U_T$.
3: Measure the final state.
The result of the computation is the rightmost bit of the state obtained by measurement. The quantum computation computes $f$ with bounded error if, for every $x = (x_1, \ldots, x_N)$, the probability that the rightmost bit of $U_T O_x U_{T-1} \cdots O_x U_0 |0\rangle$ equals $f(x_1, \ldots, x_N)$ is at least $1 - \epsilon$ for some fixed $\epsilon < \frac{1}{2}$. The quantum query complexity of $f$ is the number of queries needed to compute $f$.

This model can be extended to functions defined on a larger set or functions having more than two values.

2.2.2 Quantum Search

A search problem in an $n$ elements set $[n]$ is a subset $J \subseteq [n]$ with the characteristic function $f : [n] \rightarrow \{0, 1\}$ such that

$$f(x) = \begin{cases} 1, & \text{if } x \in J, \\ 0, & \text{otherwise}. \end{cases}$$

Any $x \in J$ is called a solution of the search problem.

In this paper, we use a generalization of Grover’s search algorithm as a quantum sub-routine, denoted by $\text{GroverAlgorithm}$, to determine whether there is any solution in a search space of size $N$. The quantum sub-routine needs $O(\sqrt{N})$ queries. We state the generalization of Grover’s search algorithm as the following theorem.

▷ Theorem 15 (see [11, 5]). Let $J$ be a search problem in an $n$ elements set $[n]$ and $f$ be the characteristic function of $J$. Given a search space $S \subseteq [n]$ with $|S| = N$, determining that whether $J \setminus S$ is empty can be done in $O(\sqrt{N})$ quantum queries to $f$ with probability of at least a constant.

2.2.3 Quantum Query Lower Bounds

In this paper, we use a quantum adversary method introduced by Ambainis to prove lower bounds for quantum query complexity.

▷ Theorem 16 (Ambainis’s quantum adversary method [1]). Let $f(x_1, x_2, \ldots, x_n)$ be a function of $n$ variables with values from a some finite set and $X, Y$ be two sets of inputs such that $f(x) \neq f(y)$ if $x \in X$ and $y \in Y$. Let $R \subseteq X \times Y$ be a relation such that

1. For every $x \in X$, there exist at least $m$ different $y \in Y$ such that $(x, y) \in R$;
2. For every $y \in Y$, there exist at least $m'$ different $x \in X$ such that $(x, y) \in R$;
3. For every $x \in X$ and $i \in [n]$, there are at most $l$ different $y \in Y$ such that $(x, y) \in R$ and $x_i \neq y_i$;
4. For every $y \in Y$ and $i \in [n]$, there are at most $l'$ different $x \in X$ such that $(x, y) \in R$ and $x_i \neq y_i$.

Then any quantum algorithm computing $f$ uses $\Omega(\sqrt{\frac{mn}{m'}})$ queries.

3 Quantum lower bounds and algorithms for matroid property problems

3.1 A Uniform Lower Bounds For Matroid Properties

As mentioned in [13], for a large number of matroid properties there is no good algorithm for determining whether these properties holds for general matroids. These properties include uniform matroid, paving matroid, Eulerian matroid and bipartite matroid decision problems.
and some calculation problems, such as finding the girth, circuit number, base number, flat number, hyperplane number and the size of the largest hyperplane. We will prove that there is no quantum algorithm with polynomial independence query oracles for these problems.

Theorem 17. For the calculation problem of finding the girth or the number of circuits (bases, flats, hyperplanes), a quantum algorithm has to query the independence oracle at least \(\Omega(\sqrt{\binom{n}{[n/2]}})\) times.

Proof. Let \(V\) be a \(n\)-set, \(r\) be an integer with \(0 \leq r \leq n\), and \(U_{r,n}\) be an uniform matroid with rank \(r\) on \(V\). For a \(r\)-set \(A \subseteq V\), \(A^1_{r,n}\) is a matroid on \(V\) with bases all \(r\)-set excepts \(A\). We encode every matroid with rank \(r\) on \(V\) to a \(\binom{n}{r}\)-bit \(0-1\) string (see Matroid Representation in Section 2.1). For every valid representation \(x \in \{0,1\}^{\binom{n}{r}}\) and any \(i \in \binom{n}{r}\), \(x[i] = 1\) indicates that a \(r\)-set which is encoded to the \(i\)-the bit is a base of \(\mathcal{M}(x)\). Let \(X = \{x(U_{r,n}) \in \{0,1\}^{\binom{n}{r}}\}, Y = \{x(A^1_{r,n}) \in \{0,1\}^{\binom{n}{r}} : \forall A \subseteq V \text{ with } |A| = r\}\). Define the following functions from \(\{0,1\}^{\binom{n}{r}}\) to a finite set:

\[
g : \{0,1\}^{\binom{n}{r}} \rightarrow [r + 1],
\]
\[
c : \{0,1\}^{\binom{n}{r}} \rightarrow \binom{\binom{n}{r}}{2},
\]
\[
b : \{0,1\}^{\binom{n}{r}} \rightarrow \binom{\binom{n}{r}}{2},
\]
\[
f : \{0,1\}^{\binom{n}{r}} \rightarrow [2^n],
\]
\[
h : \{0,1\}^{\binom{n}{r}} \rightarrow \binom{\binom{n}{r}}{2},
\]

which corresponds to finding the girth, the number of circuits, the number of bases, the number of flats, the number of hyperplanes, the size of the largest hyperplane of \(\mathcal{M}\), respectively.

For any function \(F \in \{g, c, b, f, h\}\), we have \(F(x(U_{r,n})) \neq F(x(A^1_{r,n}))\). Let \(R \subseteq X \times Y\) be a relation such that

1. For every \(x \in X\), there are \(\binom{n}{r}\) different \(y \in Y\) such that \((x, y) \in R\).
2. For every \(y \in Y\), there is one \(x \in X\) such that \((x, y) \in R\).
3. For every \(x \in X\) and \(i \in \binom{\binom{n}{r}}{2}\), there is at most one \(y \in Y\) such that \((x, y) \in R\) and \(x_i \neq y_i\).
4. For every \(y \in Y\) and \(i \in \binom{\binom{n}{r}}{2}\), there is at most one \(x \in X\) such that \((x, y) \in R\) and \(x_i \neq y_i\).

It can be seen that any quantum algorithm computing \(F\) uses \(\Omega(\sqrt{\binom{n}{r}})\) queries. When \(r = \lfloor n/2 \rfloor\), the theorem is proved.

Theorem 18. Let \(\mathcal{M} = (V, \mathcal{I})\) be a matroid with \(|V| = n\). Then any quantum algorithm to decide whether \(\mathcal{M}\) is uniform or Eulerian has to query the independence oracle at least \(\Omega(\sqrt{\binom{n}{[n/2]}})\) times.

Proof. Let \(r = \lfloor n/2 \rfloor\) and \(U_{r,n}\) be the uniform matroid with rank \(r\) on the ground set \(V\). We can see that \(U_{r,n}\) is not Eulerian, since the size of its circuits are \(r + 1\) implies that any two circuits are intersectant. For any \(r\)-set \(A \subseteq V\), let \(A^1_{r,n}\) be the matroid on \(V\) with bases all \(r\)-set except \(A\). Similarly, when \(n\) is even, let \(A^2_{r,n}\) be the matroid on \(V\) with bases all \(r\)-set except \(A\) and \(V - A\). (It is not difficult to verify that \(A^1_{r,n}\) and \(A^2_{r,n}\) are matroids with rank \(r\) on \(V\).)

In the following, we show that \(A^1_{r,n}\) and \(A^2_{r,n}\) are Eulerian matroids based on the parity of \(n\). When \(n\) is odd, we can see that \(|V - A| = r + 1\) and any proper subset of \(|V - A|\) is independent in \(A^1_{r,n}\). So \(V - A\) is a circuit of \(A^1_{r,n}\). By the definition of \(A^1_{r,n}\), we also know that \(A\) is a circuit of \(A^1_{r,n}\). Thus we can show that \(A^1_{r,n}\) is Eulerian for the union of disjoint
sets $A$ and $V - A$ is $V$. When $n$ is even, by the definition of $A_{r,n}^2$, we can see that $A$ and $V - A$ are two disjoint circuits of $A_{r,n}^2$. Thus $A_{r,n}^2$ is also Eulerian.

We encode every matroid with rank $r$ on $V$ to a $(\binom{n}{r})$-bit 0-1 string (see Matroid Representation in Section 2.1). Let $X = \{x(U_{r,n}) \in \{0, 1\}^{(C)} : \forall A \subseteq V$ with $|A| = r\}$, $Y_1 = \{x(A_{1,n}^1) \in \{0, 1\}^{(C)} : x(U_{r,n}) \in \{0, 1\}^{(C)} : \forall A \subseteq V$ with $|A| = r\}$. Define two Boolean functions $u : \{0, 1\}^{(C)} \rightarrow \{0, 1\}$ and $e : \{0, 1\}^{(C)} \rightarrow \{0, 1\}$, where $u(x) = 1$ if and only if $\mathcal{M}(x)$ is uniform and $e(x) = 1$ if and only if $\mathcal{M}(x)$ is Eulerian. Let $f \in \{u, e\}$. $Y = Y_2$ when $f = e$ and $n$ is even, otherwise $Y = Y_1$. We can see that $f(x) \neq f(y)$ for any $x \in X$ and $x \in Y$. Let $R \subseteq X \times Y$ be a relation such that

1. For every $x \in X$, there are $\binom{n}{r}(\binom{n}{r})$ different $y \in Y$ such that $(x, y) \in R$.
2. For every $y \in Y$, there one $x \in X$ such that $(x, y) \in R$.
3. For every $x \in X$ and $i \in [\binom{n}{r}]$, there is at most one $y \in Y$ such that $(x, y) \in R$ and $x_i \neq y_i$.
4. For every $y \in Y$ and $i \in [\binom{n}{r}]$, there is at most one $x \in X$ such that $(x, y) \in R$ and $x_i \neq y_i$.

It can be seen that any quantum algorithm computing $f$ uses $\Omega(\sqrt{\binom{n}{r}})$ queries. Because of $r = \lfloor n/2 \rfloor$, the theorem is proved.

### 3.2 Quantum Algorithm for Uniform Matroid Decision Problem

As mentioned in [20], the property $\lfloor n/2 \rfloor$-UNIFOR$\text{M}$ is a hardest property with respect to the independence query oracle. It requires $\Omega(\binom{n}{\lfloor n/2 \rfloor})$ independence query oracles to determine whether a matroid is uniform on a ground set with $n$ elements. We will present a quantum algorithm using $O(\sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ independence query oracles to solve this problem and thus the algorithm is asymptotically optimal.

**Fact 1.** Let $\mathcal{M} = (V, I)$ be a matroid with $|V| = n$. There is a greedy algorithm to compute the rank of $\mathcal{M}$ using $O(n)$ independence query oracles.

**Proof.** Let $J = \emptyset$, for every $e \in V$, if $J \cup \{e\}$ is an independent set in $\mathcal{M}$, then update $J$ (i.e. $J := J \cup \{e\}$, otherwise discard $e$). This procedure, denoted by $\text{GreedyAlgorithm}$, is essentially a greedy algorithm with all the elements being the same weight. We must traverse all elements in $V$. The fact is proved.

**Theorem 19.** Let $\mathcal{M} = (V, I)$ be a matroid with $|V| = n$. There is a quantum algorithm to decide whether it is uniform using $O(\sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ independence query oracles.

**Proof.** For determining whether a matroid is uniform, we just need to check the sets whose size are equal to its rank. Consider the Algorithm [1] Uniform Matroid. First, we use Fact 1 to compute the rank $r$ of $\mathcal{M}$. Then use GroverAlgorithm to determine that whether there is a $S \subseteq V$ with $|S| = r$ such that $\mathcal{O}(S) = 0$ in the set $\{J \subseteq V : |J| = r\}$. If we find such a $S$, we can affirmatively conclude that $\mathcal{M}$ is not a uniform matroid. Otherwise we repeat GroverAlgorithm several times. Finally, if all the results of GroverAlgorithm such that $\mathcal{O}(S) = 1$, it can be shown that $\mathcal{M}$ is an uniform matroid with a high probability. The total number of independence query oracle required is $O(n) + \text{MAX REPEAT} \cdot \sqrt{\binom{n}{r}}$. And the number $\text{MAX REPEAT}$ is a constant. When the rank $r = \lfloor n/2 \rfloor$, the algorithm uses $O(\sqrt{\binom{n}{\lfloor n/2 \rfloor}})$ independence query oracles.
Algorithm 1 Uniform Matroid

```c
int UniformMatroid(M, V)
{
    int r = GreedyAlgorithm(M, V);
    int k = MAX_REPEAT;
    while (k > 0) // Repeat GroverAlgorithm up to MAX_REPEAT times.
    {
        S = GroverAlgorithm({J ⊆ V : |J| = r});
        if (O_i(S) == 0) return 0;
        k = k - 1;
    }
    return 1;
}
```

By Theorem 19 and Theorem 18 we can see that the presented algorithm is asymptotically optimal. This means that the quantum query complexity of the uniform decision problem is $\Theta\left(\binom{n}{\lfloor n/2 \rfloor}\right)$.

### 3.3 Quantum Algorithm for Finding Girth

**Theorem 20.** Let $\mathcal{M} = (V, \mathcal{I})$ be a matroid with $|V| = n$. There is a quantum algorithm to find the girth of $\mathcal{M}$ using $O(\log n \sqrt{n \binom{n}{\lfloor n/2 \rfloor}})$ independence query oracles.

**Proof.** Consider the Algorithm 2: Compute Girth, where a greedy algorithm is used to compute the rank $r$ of $\mathcal{M}$ and GroverAlgorithm is used to determine that whether there is a $S \in \{J \subseteq V : |J| = r\}$ such that $O_i(S) = 0$.

Suppose the girth of $\mathcal{M}$ is not $\infty$ and consider the worst case that the outer while is executed $\log n$ times. Each time the outer while is executed, we repeatedly call GroverAlgorithm up to MAX_REPEAT times. The number MAX_REPEAT is a constant. And the maximum number of independence oracle queried by GroverAlgorithm is $\sqrt{n \binom{n}{\lfloor n/2 \rfloor}}$. The total number of query to independence oracle is not more than $O(n) + MAX\_REPEAT \cdot \log n \sqrt{n \binom{n}{\lfloor n/2 \rfloor}}$.

So ComputeGirth uses $O(\log n \sqrt{n \binom{n}{\lfloor n/2 \rfloor}})$ independence queries to compute the girth.

By Theorem 20 and Theorem 17 we can see that the quantum algorithm is almost optimal.
Algorithm 2 Compute Girth

```
/*
Input: Matroid $\mathcal{M}$ accessed by $O_i$ and ground set $V$.
Output: The girth of $\mathcal{M}$.
*/

int ComputeGirth(M, V)
{
    int girth = $\infty$;
    if ($O_i(V) == 1$) return girth;
    int r = GreedyAlgorithm(M, V);
    int lindex = 1, rindex = r + 1, k = 0;
    while (lindex != rindex)
    {
        girth = (lindex + rindex) / 2;
        k = MAX_REPEAT; // Repeat GroverAlgorithm up to MAX_REPEAT times
        while (k)
        {
            S = GroverAlgorithm($\{J \subseteq V : |J| = girth\}$);
            if ($O_i(S) == 0$)
            {
                rindex = girth;
                break;
            }
            k = k - 1;
        }
        if (k == 0)
        {
            lindex = girth;
        }
    }
    return girth;
}
```

3.4 Quantum Complexity and Algorithm for Paving Matroid Decision Problem

Paving matroids are a very important type of matroids in matroid theory. In the early 1970’s, Blackburn, Crapo, and Hig
c [4] noticed that most of the matroids on a ground set of up to 8 elements are paving matroids. Crapo and Rota [6] suggested that perhaps paving matroids “would actually predominate in any asymptotic enumeration of geometries”. Mayhew et al. [17] gave a conjecture, “Asymptotically, almost every matroid is paving”. Here we give an almost optimal quantum algorithm to determine whether a matroid is paving.

▶ Fact 2. Given a matroid $\mathcal{M} = (V, I)$ with rank $r \geq 2$, if there exists a circuit $C$ in $\mathcal{M}$ with $|C| \leq r - 2$. Then there must be a dependent set $J$ with $|J| = r - 1$ in $\mathcal{M}$.

Proof. This fact is obvious. Let $X \subseteq V \setminus C$ with $|X| = r - 1 - |C|$, then $C \subseteq J = C \cup X$ is a dependent set of $\mathcal{M}$ with $|J| = r - 1$.

▶ Theorem 21. Let $\mathcal{M} = (V, I)$ be a matroid with $|V| = n$. There is a quantum algorithm to decide whether it is paving using $O(\sqrt{\binom{n}{[n/2]}})$ independence query oracles.
Proof. Consider the Algorithm Paving Matroid. By Fact 2, we can see that a matroid $\mathcal{M}$ with rank $r \geq 2$ is a paving matroid if and only if every the $(r-1)$-set is an independent set. First we use Fact 1 to compute the rank $r$ of $\mathcal{M}$. Then we use GroverAlgorithm to determine whether there is a a $S \subseteq V$ with $|S| = r - 1$ such that $\mathcal{O}_i(S) = 0$ in the set $\{ J \subseteq V : |J| = r - 1 \}$. If we find such a $S$, we can affirmatively conclude that $\mathcal{M}$ is not a paving matroid. Otherwise we repeat GroverAlgorithm several times. Finally, if all the results of GroverAlgorithm such that $\mathcal{O}_i(S) = 1$, it can be shown that $\mathcal{M}$ is a paving matroid with a high probability. The total number of independence query oracle required is $O(n) + \text{MAX\_REPEAT} \cdot \sqrt{n/r}$. And the number MAX\_REPEAT is a constant. When $r - 1 = \lfloor n/2 \rfloor$, the total number of independence query oracles used is $O(\sqrt{n/\lfloor n/2 \rfloor})$.

\begin{algorithm}
\begin{algorithmic}[1]
\State 
\Comment Paving Matroid
\Function{PavingMatroid}{\textit{M}, \textit{V}}
\State int \textit{r} = \text{GreedyAlgorithm}(\textit{M}, \textit{V});
\State int \textit{k} = \text{MAX\_REPEAT};
\While{(\textit{k} > 0)} // Repeat GroverAlgorithm up to MAX\_REPEAT times.
\State $\textit{S} = \text{GroverAlgorithm}(\{ J \subseteq V : |J| = r - 1 \})$;
\If{($\mathcal{O}_i(S) = 0$)} \textbf{return} 0;
\EndIf
\State \textit{k} = \textit{k} - 1;
\EndWhile
\textbf{return} 1;
\EndFunction
\end{algorithmic}
\end{algorithm}

\begin{fact}
Given a ground set $V$ with $|V| = n$, an integer $r$ with $1 \leq r \leq n$ and $A, C \subseteq V$ with $|A| = |C| = r - 1$. Define $J_r, J_A, J_C$ as Notations in Section 2.7. Let $B_A = J_r - J_A$, $B_C = J_r - J_C$. We use a $\binom{n}{r}$ bit 0-1 string $x(M)$ encodes a matroid $\mathcal{M}$ with rank($\mathcal{M}$) = $r$. For $i \in \binom{n}{r}$, $x(\mathcal{M})[i] = 1$ indicates that some $r$-set be a base of $\mathcal{M}$. For any different $A$ and $C$, the two matroids determined by $B_A$ and $B_C$ as the collection of bases, denoted by $\mathcal{M}_A$ and $\mathcal{M}_C$, then there is at most one $i \in \binom{n}{r}$ such that $x(\mathcal{M}_A)[i] = x(\mathcal{M}_C)[i] = 0$.
\end{fact}

\begin{proof}
For any two different $(r-1)$-set $A$ and $C$, we have $0 \leq |A \cap C| \leq r - 2$, $|A \cap C| < r - 2$ implies that for any $e \in V - A$ and $e' \in V - C$, we have $A_e \neq C_{e'}$. Thus for any $i \in \binom{n}{r}$, $x(\mathcal{M}_A)[i]$ and $x(\mathcal{M}_C)[i]$ will not be 0 at the same time. $|A \cap C| = r - 2$ implies that there exists one and only one $r$-set (that is $A \cup C$) be the superset of $A$ and $C$ with cardinality $r$. The fact is proved.
\end{proof}

\begin{theorem}
Let $\mathcal{M} = (V, I)$ be a matroid with $|V| = n$. Then any quantum query algorithm to decide whether $\mathcal{M}$ is paving requires at least $\Omega(\sqrt{n/\lfloor n/2 \rfloor}/n)$ independence query oracles.
\end{theorem}

\begin{proof}
Given an integer $r$ with $1 \leq r \leq n$ and a subset $A \subseteq V$ with $|A| = r - 1$, $e \in V - A$, define $J_r, J_A$ as Notations in Section 2.1. Let $B_A = J_r - J_A$. Then there is a matroid on $V$ that can be determined by $B_A$ as the collection of bases, denoted by $\mathcal{M}_A$. Furthermore, $\mathcal{M}_A$ is not a paving matroid (because $A \notin I(\mathcal{M}_A)$ and $\text{rank}(\mathcal{M}_A) = r > |A|$).
\end{proof}
We encode every matroid with rank \( r \) on \( V \) to a different \( \binom{n}{r} \)-bit 0–1 string (see Matroid Representation \( 2.1 \)). Let \( X = \{x(U_{1,n}) \in \{0,1\}^{\binom{n}{r}} \} \), \( Y = \{x(M_A) \in \{0,1\}^{\binom{n}{r}} : \forall A \subseteq V \text{ with } |A| = r - 1 \} \). Define a Boolean function \( f : \{0,1\}^{\binom{n}{r}} \to \{0,1\} \), \( f(x) = 1 \) if and only if the matroid \( M(x) \) is a paving matroid. By the definition of \( X \) and \( Y \), it is easy to verify that every \( x \in X \) is a paving matroid and every \( y \in Y \) is not a paving matroid. So \( X,Y \) are two sets of inputs such that \( f(x) \neq f(y) \) if \( x \in X \) and \( y \in Y \). Let \( R \subseteq X_r \times Y_r \) be a relation such that

1. For every \( x \in X \), there exist \( \binom{r-1}{n-1} \) different \( y \in Y \) such that \( (x,y) \in R \).
2. For every \( y \in Y \), there exists one \( x \in X \) such that \( (x,y) \in R \).
3. For every \( x \in X \) and \( i \in \binom{r}{n-1} \), there are at most \( \binom{r-1}{n-1} \) (known from Fact \( 3 \)) different \( y \in Y \) such that \( (x,y) \in R \) and \( x_i \neq y_i \).
4. For every \( y \in Y \) and \( i \in \binom{r}{n} \), there is at most one \( x \in X \) such that \( (x,y) \in R \) and \( x_i \neq y_i \).

It can be seen that any quantum algorithm computing \( f \) uses \( \Omega(\sqrt{\binom{n}{r-1}}/r) \) queries. When \( r - 1 = \lfloor n/2 \rfloor \), we obtain the lower bound \( \Omega(\sqrt{\binom{n}{\lfloor n/2 \rfloor}}/n) \).

By Theorem \( 21 \) and Theorem \( 22 \) one can see that our quantum algorithm is almost optimal.

### 3.5 Trivial and Loopless Decision Problems

**Theorem 23.** The quantum query complexity of trivial and loopless decision problems are \( \Theta(\sqrt{n}) \).

**Proof.** From Definitions \( 7 \) and \( 8 \) we can see that a matroid is a trivial matroid if and only if all the singleton in \( V \) are dependent, and a matroid is loopless if and only if all the singleton in \( V \) are independent. In other words, for the trivial(loopless) matroid decision problem, if there is an \( e \in V \) such that \( \mathcal{O}_i(\{e\}) = 1(0) \) implies the matroid is not a trivial(loopless) matroid. So these two decision problems are essentially the unordered search problem in the set \( V \). By \( 11, 5, 24 \), the quantum query complexity of the trivial and loopless decision problems are \( \Theta(\sqrt{n}) \). ▶

### 4 Conclusion

In this paper, we discussed quantum speedup and limitations on matroid properties, assuming that a matroid can be accessed through the independence query oracle. We obtained lower bounds on the quantum query complexity for the calculation problem of finding the girth or the number of circuits (bases, flats, hyperplanes) of a matroid, and for the decision problem of deciding whether a matroid is uniform or Eulerian. These lower bounds imply that there is no polynomial-time quantum algorithm for these problems. We also presented quantum algorithms with quadratic speedup over classical ones for some of these problems and the algorithms are asymptotically optimal or almost optimal.

There are a large number of matroids in a ground set with \( n \) elements. This gives us many potential possibilities. There are some interesting questions worthy of further consideration.

1. Are there matroid problems for which quantum algorithms can show superpolynomial speedup over their counterparts? If yes, can we characterize the properties owned by these problems?
2. Robinson and Welsh have shown that the capabilities of oracles given by different definitions are very different. What about in the quantum case?

3. In this paper, we simply use Grover’s algorithm to solve some relatively simple problems in [13]. What about other problems?

References

1. Andris Ambainis. Quantum lower bounds by quantum arguments. In F. Frances Yao and Eugene M. Luks, editors, Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA, pages 636–643. ACM, 2000. doi:10.1145/335305.335394

2. Riccardo Bassoli, Hugo Marques, Jonathan Rodriguez, Kenneth W. Shum, and Rahim Tafazolli. Network coding theory: A survey. IEEE Commun. Surv. Tutorials, 15(4):1950–1978, 2013. doi:10.1109/SURV.2013.013013.00104

3. Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. In 39th Annual Symposium on Foundations of Computer Science, FOCS ’98, November 8-11, 1998, Palo Alto, California, USA, pages 352–361. IEEE Computer Society. doi:10.1109/SFCS.1998.743485

4. John E. Blackburn, Henry H. Crapo, and Denis A. Higgs. A catalogue of combinatorial geometries. Mathematics of Computation, 27(121):155–s95, 1973. URL: https://doi.org/10.1090/S0025-5718-1973-0419270-0

5. Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. Fortschritte der Physik: Progress of Physics, 46(4-5):493–505, 1998. URL: https://doi.org/10.1002/3527603093.ch10Citations

6. Henry H Crapo and Gian-Carlo Rota. On the foundations of combinatorial theory: Combinatorial geometries. MIT press Cambridge, Mass., 1970.

7. D. Deutsch. Quantum theory, the church-turing principle and the universal quantum computer. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 400(1818):97–117, 1985. doi:10.1098/rspa.1985.0070

8. D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. Proceedings of the Royal Society of London Series A, 439(1907):553–558, December 1992. doi:10.1098/ rspa.1992.0167

9. Jack R. Edmonds. Matroids and the greedy algorithm. Math. Program., 1(1):127–136, 1971. doi:10.1007/BF01584082

10. Richard P. Feynman. Simulating physics with computers. International Journal of Theoretical Physics, 21(6):467–488, 1982. doi:10.1007/BF02650179

11. Lov K. Grover. A fast quantum mechanical algorithm for database search. In Gary L. Miller, editor, Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996, pages 212–219. ACM, 1996. doi:10.1145/237814.237866

12. M. Iri. Applications of matroid theory. In Achim Bachem, Bernhard Korte, and Martin Grötschel, editors, Mathematical Programming The State of the Art, XIth International Symposium on Mathematical Programming, Bonn, Germany, August 23-27, 1982, pages 158–201. Springer, 1982. doi:10.1007/978-3-642-68874-6_8

13. Per M. Jensen and Bernhard Korte. Complexity of matroid property algorithms. SIAM J. Comput., 11(1):184–190, 1982. doi:10.1137/0211014

14. Alexei Y. Kitaev. Quantum measurements and the abelian stabilizer problem. Electron. Colloquium Comput. Complex., (3), 1996. URL: https://eccc.weizmann.ac.il/eccc-reports/1996/TR96-003/index.html

15. Donald E. Knuth. The asymptotic number of geometries. J. Comb. Theory, Ser. A, 16(3):398–400, 1974. doi:10.1016/0097-3165(74)90063-6
Quantum Speedup and Limitations on Matroid Properties

16 Eugene L. Lawler. *Combinatorial optimization: networks and matroids*. Holt, Rinehart and Winston, New York-Montreal, Que.-London, 1976.

17 Dillon Mayhew, Mike Newman, Dominic Welsh, and Geoff Whittle. On the asymptotic proportion of connected matroids. *Eur. J. Comb.*, 32(6):882–890, 2011. [doi:10.1016/j.ejc.2011.01.016]

18 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. [doi:10.1017/CBO9780511976667]

19 James Oxley. *Matroid theory*, volume 21 of *Oxford Graduate Texts in Mathematics*. Oxford University Press, Oxford, second edition, 2011. [doi:10.1093/acprof:oso/9780198566946.001.0001]

20 G. C. Robinson and D. J. A. Welsh. The computational complexity of matroid properties. *Mathematical Proceedings of the Cambridge Philosophical Society*, 87(1):29–45, 1980. [doi:10.1017/S0305004100056498]

21 Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 124–134. IEEE Computer Society, 1994. [doi:10.1109/SFCS.1994.365700]

22 Daniel R. Simon. On the power of quantum computation. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 116–123. IEEE Computer Society, 1994. [doi:10.1109/SFCS.1994.365701]

23 Hassler Whitney. On the abstract properties of linear dependence. *American Journal of Mathematics*, 57(3):509–533, 1935. URL: [http://www.jstor.org/stable/2371182](http://www.jstor.org/stable/2371182)

24 Christof Zalka. Grover’s quantum searching algorithm is optimal. *Phys. Rev. A*, 60:2746–2751, Oct 1999. URL: [https://link.aps.org/doi/10.1103/PhysRevA.60.2746](https://link.aps.org/doi/10.1103/PhysRevA.60.2746) [doi:10.1103/PhysRevA.60.2746]