Algorithmica (2002) 34: 449–461
DOI: 10.1007/s00453-002-0978-1

Algorithmica

© 2002 Springer-Verlag New York Inc.

Sharp Quantum versus Classical Query Complexity Separations[1]

J. Niel de Beaudrap,[2] Richard Cleve,[2] and John Watrous[2]

Abstract. We obtain the strongest separation between quantum and classical query complexity known to date—specifically, we define a black-box problem that requires exponentially many queries in the classical bounded-error case, but can be solved exactly in the quantum case with a single query (and a polynomial number of auxiliary operations). The problem is simple to define and the quantum algorithm solving it is also simple when described in terms of certain quantum Fourier transforms (QFTs) that have natural properties with respect to the algebraic structures of finite fields. These QFTs may be of independent interest, and we also investigate generalizations of them to noncommutative finite rings.

Key Words. Quantum algorithms, Quantum Fourier transform.

1. Introduction. Shor's algorithm [17] for factoring integers in polynomial-time on a quantum computer evolved from a series of quantum algorithms in the query model. This model appears to be useful for exploring the computational power of quantum information. In the query model the input data is embodied in a black-box and the goal is to deduce some property of the black-box efficiently. Efficiency is measured in terms of the number of queries made to the black-box. A secondary measure of efficiency is also considered: the number of auxiliary operations that must be performed to generate the input to the queries and process the output. We implicitly require that the number of auxiliary operations scales polynomially with the number of bits/qubits input to each query.

The first instance of a quantum algorithm outperforming a classical algorithm in the query model was due to Deutsch [10], where a quantum algorithm is able to solve a 2-bit query problem with one query (see also [7]), whereas any classical algorithm for the problem requires two queries. (A *k-bit* query is one that takes k bits/qubits as input and returns k bits/qubits as output.) This was extended by Deutsch and Jozsa [11], who defined an $(n+1)$-bit query problem that can be solved exactly with one query by a quantum algorithm whereas it requires $\Omega(2^n)$ queries to solve exactly classically. In spite of the apparent strength of this separation, the problem is only hard in the classical setting if the algorithm must be exact, meaning that no probability of error is tolerated. A bounded-error algorithm is one that is allowed to err, provided that for any black-box instance its error probability is bounded below some constant smaller than $\frac{1}{2}$. There is a classical algorithm that solves the problem in [11] with bounded error using only $O(1)$ queries.

[1] This research was partially supported by Canada's NSERC.

[2] Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4. {jd, cleve, jwatrous}@cpsc.ucalgary.ca.

Received March 7, 2001. Communicated by M. Mosca and A. Tapp.
Online publication September 16, 2002.

Subsequent work by Bernstein and Vazirani [3] included an $(n+1)$-bit query problem that can be solved exactly with a quantum algorithm making one query, whereas any bounded-error classical algorithm for it requires n queries. They also showed that a recursively defined version of this problem results in a $\Theta(n)$-bit query problem whose exact quantum and bounded-error classical query complexities are $O(n \log n)$ and $n^{\Omega(\log n)}$, respectively. This was improved by Simon [18], who gives a fairly simple $O(n)$ versus $\Omega(2^{n/2})$ bounded-error quantum versus bounded-error classical query separation. Brassard and Høyer [5] later showed that the problem considered by Simon can in fact be solved exactly in the quantum setting with $O(n)$ queries.

When cast in the query model, Shor's factoring algorithm can be viewed as an extension of Simon's work—it is a quantum algorithm that solves a $3n$-bit query problem with bounded-error with $O(1)$ quantum queries, while any classical algorithm for this problem requires $\Omega(2^{n/3}/\sqrt{n})$ queries (the lower bound is proved in [6]).

What is the sharpest quantum versus classical query complexity separation possible? For problems that can be solved exactly with a single quantum query, it appears that the maximum classical bounded-error query complexity previously known for such a problem is n [3]. However, if the efficiency and performance of the quantum algorithm are relaxed to allow $O(1)$ queries and a bounded-error result, then there is a problem whose classical bounded-error query complexity is exponential [17], [6].

Presently, we show that the best of the above two scenarios is possible by exhibiting a $2n$-bit query problem such that:

- In the quantum setting, a single query suffices to solve the problem exactly. Moreover, the auxiliary operations are very simple; they consist of $O(n)$ Hadamard gates followed by $O(n^2)$ classical gate operations that can occur after a measurement is made.
- In the classical setting, $\Omega(2^{n/2})$ queries to the black-box are necessary to solve the problem with bounded error.

The problem that achieves the above, which we call the *hidden linear structure* problem, is defined over the field $GF(2^n)$ as follows. Assume elements of the finite field $GF(2^n)$ are identified with strings in the set $\{0, 1\}^n$. Let π be an arbitrary permutation on $GF(2^n)$ and let $s \in GF(2^n)$. Define the black-box B as computing the mapping from $GF(2^n) \times GF(2^n)$ to itself defined as $B(x, y) = (x, \pi(y + sx))$. The goal of the query problem is to determine the value of s.

It should be noted that this problem is related to, but different from, the *hidden linear function* problem considered by Boneh and Lipton [4]. In our problem the linear structure occurs over the field $GF(2^n)$ (and involves the multiplicative structure of $GF(2^n)$), whereas for the hidden linear function problem of Boneh and Lipton the linear structure is of certain periodic functions from the additive group $\mathbb{Z}^k$ to some arbitrary range. This does not result in the quantum versus classical query complexity separation that we obtain.

It should also be noted that our hidden linear structure problem is a special case of the *hidden subgroup* problem defined by Brassard and Høyer [5] and Mosca and Ekert [16]. (This relationship was pointed out to us by Hallgren [14].) However, using standard techniques for the hidden subgroup problem results in a quantum algorithm solving the hidden linear structure problem with $\Theta(n)$ queries, as opposed to a single query as required by our algorithm.

Finally, one may also consider a variant of our hidden linear structure problem defined over a finite ring (such as $\mathbb{Z}_{2^n}$) rather than a field. However, the exponential classical query complexity lower bound depends on the field structure and does not always hold for finite rings. For example, in the case of $\mathbb{Z}_{2^n}$, the classical query complexity is $n+1$ rather than exponential (this is explained in Section 3).

Our single-query quantum algorithm for the hidden linear structure problem is based on an extension of the quantum Fourier transform (QFT) to finite fields whose behavior has natural properties with respect to the field structure. This QFT is motivated and defined in Section 2, where an efficient quantum algorithm for it is also given. The quantum algorithm and classical lower bound for the hidden linear structure problem are given in Section 3. In Section 4 the QFT is generalized to rings of matrices over finite fields.

Related work. Van Dam and Hallgren have independently proposed a definition for QFTs over finite fields that is similar to ours, and have applied these transforms in the context of black-box problems called the "shifted quadratic character problems." Their work first appeared as [9] and the preliminary version of this paper appeared as [2].

2. Quantum Fourier Transforms for Finite Fields. In this section we propose a definition for QFTs over finite fields, whose behavior has natural properties with respect to a given field's structure. We also show how to compute these transformations efficiently.

We assume the reader is familiar with basic concepts regarding finite fields and computations over finite fields (see, for instance, [8], [12], and [15]). As usual, we let $GF(q)$ denote the finite field having $q = p^n$ elements for some prime p. We assume that an irreducible polynomial $f(Z) = Z^n - \sum_{j=0}^{n-1} a_j Z^j$ over $GF(p)$ is fixed, and that elements of $GF(q)$ are represented as polynomials over $GF(p)$ modulo f in the usual way. We write $x = (x_0, \ldots, x_{n-1})$ to denote the field element corresponding to $x_0 + x_1 Z + \cdots + x_{n-1} Z^{n-1}$, and we identify x with the column vector $\vec{x} = [x_0, \ldots, x_{n-1}]^{\mathsf{T}}$.

DEFINITION 2.1. Let φ: $GF(q) \to GF(p)$ be any nonzero linear mapping (viewing elements of $GF(q)$ as n-dimensional vectors over $GF(p)$ as above). Then we define the *quantum Fourier transform* (*QFT*) *over* $GF(q)$ *relative to* φ (denoted $F_{q,\varphi}$) as follows. For each $x \in GF(q)$,

$$F_{q,\varphi}\colon |x\rangle \mapsto \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{\varphi(xy)} |y\rangle$$

for $\omega = e^{2\pi i/p}$, and let $F_{q,\varphi}$ be extended to arbitrary quantum states by linearity.

A natural choice for φ is the trace, since this gives a transform independent of the choice of f. However, we do not require this property, and so we allow φ to be arbitrary. It should be noted that, for any prime q, the above Fourier transform is essentially identical in form to the conventional cyclic Fourier transform modulo q.

An important property of these transformations is illustrated in Figure 1, where F denotes the QFT and the two-register gate labeled by $s \in GF(q)$ denotes the mapping

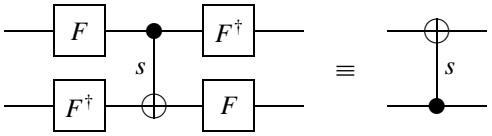

Fig. 1. The control/target inversion property.

$|x\rangle|y\rangle \mapsto |x\rangle|y + sx\rangle$. We refer to the latter gate as a *controlled-ADD$_s$* gate, with its first input called the *control* register and its second input called the *target* register. The property illustrated in the figure is referred to as the *control/target inversion property*. In words, conjugating a controlled-ADD$_s$ gate by $F \otimes F^\dagger$ switches its control and target registers. In the special case of $GF(2)$, F is the Hadamard gate and the two-qubit gate is the controlled-NOT gate (when $s = 1$).

THEOREM 1. *For $q = p^n$ and any nonzero linear mapping φ: $GF(q) \to GF(p)$, $F_{q,\varphi}$ is unitary and satisfies the control/target inversion property of Figure* 1.

PROOF. First we show that $F_{q,\varphi}^\dagger F_{q,\varphi}|x\rangle = |x\rangle$ for every $x \in GF(q)$. We have

$$\begin{aligned} F_{q,\varphi}^\dagger F_{q,\varphi}|x\rangle &= F_{q,\varphi}^\dagger \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{\varphi(xy)}|y\rangle = \frac{1}{q} \sum_{y \in GF(q)} \sum_{z \in GF(q)} \omega^{\varphi(xy)-\varphi(yz)}|z\rangle \\ &= \sum_{z \in GF(q)} \left(\frac{1}{q} \sum_{y \in GF(q)} \omega^{\varphi(y(x-z))} \right) |z\rangle = |x\rangle, \end{aligned}$$

following from the fact that $\varphi(w)$ must be uniformly distributed over $GF(p)$ as w ranges over $GF(q)$ (since φ is linear and not identically zero).

Next we verify that the control/target inversion property holds, namely that for A_s and B_s defined by $A_s|x\rangle|y\rangle = |x\rangle|y + sx\rangle$ and $B_s|x\rangle|y\rangle = |x + sy\rangle|y\rangle$ we have

$$(F_{q,\varphi}^\dagger \otimes F_{q,\varphi}) A_s (F_{q,\varphi} \otimes F_{q,\varphi}^\dagger) = B_s.$$

To prove this relation holds, we define

$$|\psi_x\rangle = F_{q,\varphi}|x\rangle = \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{\varphi(xy)}|y\rangle$$

for each $x \in GF(q)$, and note that for P_w defined by $P_w|x\rangle = |x + w\rangle$ we have

$$P_w|\psi_{-x}\rangle = \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{-\varphi(xy)}|y + w\rangle = \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{-\varphi(xy-xw)}|y\rangle = \omega^{\varphi(xw)}|\psi_{-x}\rangle.$$

Now, for each $x, y \in GF(q)$ we have

$$\begin{aligned}
&(F_{q,\varphi}^{\dagger} \otimes F_{q,\varphi}) A_s (F_{q,\varphi} \otimes F_{q,\varphi}^{\dagger}) |x\rangle|y\rangle \\
&\quad = (F_{q,\varphi}^{\dagger} \otimes F_{q,\varphi}) A_s \left(\frac{1}{\sqrt{q}} \sum_{z \in GF(q)} \omega^{\varphi(xz)} |z\rangle |\psi_{-y}\rangle \right) \\
&\quad = (F_{q,\varphi}^{\dagger} \otimes F_{q,\varphi}) \left(\frac{1}{\sqrt{q}} \sum_{z \in GF(q)} \omega^{\varphi(xz)} \omega^{\varphi(ysz)} |z\rangle |\psi_{-y}\rangle \right) \\
&\quad = (F_{q,\varphi}^{\dagger} \otimes F_{q,\varphi}) |\psi_{x+sy}\rangle |\psi_{-y}\rangle \\
&\quad = |x + sy\rangle |y\rangle \\
&\quad = B_s |x\rangle |y\rangle
\end{aligned}$$

as required. □

Next we describe quantum circuits for performing $F_{q,\varphi}$ and analyze their complexity. Let $C(p, \varepsilon)$ denote the minimum size of a quantum circuit approximating the QFT modulo p to within accuracy ε. Note that $C(p, 0) \in O(p^2 \log p)$ [1] and, for $\varepsilon > 0$, $C(p, \varepsilon) \in O(\log p \log\log p + \log p \log 1/\varepsilon)$ when $\varepsilon \in \Omega(1/p)$ [13].

THEOREM 2. *For $q = p^n$ and any nonzero linear mapping φ: $GF(q) \to GF(p)$, $F_{q,\varphi}$ can be performed with accuracy ε by a quantum circuit of size $O(n^2(\log p)^2) + nC(p, \varepsilon/n)$.*

Thus, when $p = 2$ (or any constant), the QFT circuit size is $O(n^2)$ in the exact case.

PROOF OF THEOREM 2. For any choice of φ (linear and nonzero), there exists a uniquely determined $n \times n$ matrix M_φ over $GF(p)$ such that $\varphi(xy) = \vec{x}^{\mathsf{T}} M_\varphi \vec{y}$. We show how to obtain efficiently such a matrix M_φ explicitly for any given φ below. The quantum circuit performing $F_{q,\varphi}$ will depend on M_φ, and we note that M_φ must be invertible.

We have

$$\begin{aligned}
F_{q,\varphi} |x\rangle &= \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{\vec{x}^{\mathsf{T}} M_\varphi \vec{y}} |y\rangle \\
&= \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{\vec{x}^{\mathsf{T}} \vec{y}} |M_\varphi^{-1} \vec{y}\rangle = \frac{1}{\sqrt{q}} \sum_{y \in GF(q)} \omega^{(M_\varphi^{\mathsf{T}} \vec{x})^{\mathsf{T}} \vec{y}} |y\rangle.
\end{aligned}$$

From this we conclude that

$$F_{q,\varphi} = M_\varphi^{-1} (F_p \otimes \cdots \otimes F_p) = (F_p \otimes \cdots \otimes F_p) M_\varphi^{\mathsf{T}},$$

where F_p denotes the usual QFT modulo p and, for $A \in \{M_\varphi^{-1}, M_\varphi^{\mathsf{T}}\}$, we identify A with the reversible operation that maps each $|\vec{x}\rangle$ to $|A\vec{x}\rangle$. This relation is illustrated in Figure 2.

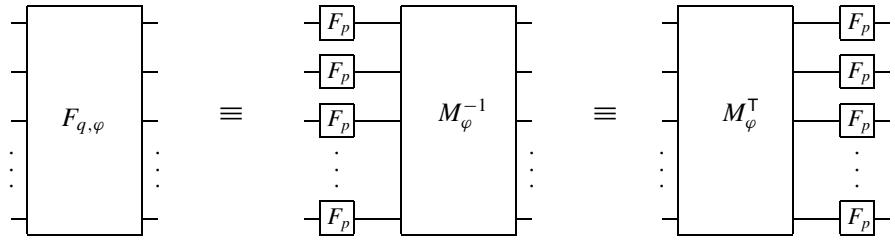

Fig. 2. Equivalent circuits for $F_{q,\varphi}$.

The upper bound of $O(n^2(\log p)^2) + nC(p, \varepsilon/n)$ now follows from the observation that in order to implement $F_{q,\varphi}$ with accuracy ε it suffices to implement each circuit for F_p with accuracy ε/n (contributing $nC(p, \varepsilon/n)$ gates to the final circuit) and to implement the circuit for multiplication by either M_φ^{T} or M_φ^{-1} exactly. Let $A \in \{M_\varphi^{-1}, M_\varphi^{\mathsf{T}}\}$. Multiplication of an n-dimensional vector v by A can be done with $O(n^2)$ arithmetic operations in $GF(p)$, each of which can be performed by a circuit of size $O((\log p)^2)$, resulting in a circuit of size $O(n^2(\log p)^2)$. In order to implement this transformation reversibly within the same size bound, it suffices to be able to invert the computation in this size bound. Inverting this computation is simply multiplication by A^{-1}, which can be performed in precisely the same size bound. (Note that the circuit itself does not need to invert A, but rather information about A and A^{-1} is pre-computed and "hard-coded" into the appropriate circuit for $F_{q,\varphi}$.)

Now we return to the question of determining the matrix M_φ corresponding to a given φ. First, note that multiplication of field elements satisfies

$$(z_0, \ldots, z_{n-1}) = (x_0, \ldots, x_{n-1}) \cdot (y_0, \ldots, y_{n-1}),$$

where

$$z_i = \vec{x}^{\mathsf{T}} B_i \vec{y} \tag{1}$$

for a certain sequence of $n \times n$ matrices $B_0, \ldots, B_{n-1}$ over $GF(p)$.

Let us explicitly construct a sequence $B_0, \ldots, B_{n-1}$ that satisfies (1). To do this, it will be helpful to review the notion of *Hankel matrices*. An $n \times n$ Hankel matrix A is a matrix of the form

$$A = \begin{bmatrix} t_0 & t_1 & t_2 & \cdots & t_{n-1} \\ t_1 & t_2 & t_3 & \cdots & t_n \\ t_2 & t_3 & t_4 & \cdots & t_{n+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{n-1} & t_n & t_{n+1} & \cdots & t_{2n-2} \end{bmatrix}. \tag{2}$$

That is, the "anti-diagonals" each contain only one element (or, equivalently, $A[i, j]$ depends only on $i + j$). The Hankel matrix in (2) will be denoted $\mathrm{Hankel}(t_0, t_1, \ldots, t_{2n-2})$.

Recall that we have

$$Z^n \equiv \sum_{j=0}^{n-1} a_j Z^j \pmod{f(Z)},$$

where f is as described at the beginning of the current section. Write $a_j^{(0)} = a_j$ for $j = 0, \ldots, n-1$. We actually need numbers $a_j^{(k)}$ (for $j = 0, \ldots, n-1, k = 0, \ldots, n-2$) such that

$$Z^{n+k} \equiv \sum_{j=0}^{n-1} a_j^{(k)} Z^j \pmod{f(Z)}.$$

These numbers are easy to obtain. Define an $n \times n$ matrix V as follows:

$$V = \begin{bmatrix} 0 & 0 & \cdots & 0 & a_0 \\ 1 & 0 & \cdots & 0 & a_1 \\ 0 & 1 & \cdots & 0 & a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & a_{n-1} \end{bmatrix}.$$

Then

$$[a_0^{(k)}, \ldots, a_{n-1}^{(k)}]^{\mathsf{T}} = V^k [a_0, \ldots, a_{n-1}]^{\mathsf{T}} = V^{k+1} [0, \ldots, 0, 1]^{\mathsf{T}}.$$

Finally, we can describe the matrices $B_0, \ldots, B_{n-1}$. For each $i = 0, \ldots, n-1$,

$$B_i = \text{Hankel}(\delta_{0,i}, \delta_{1,i}, \ldots, \delta_{n-1,i}, a_i^{(0)}, a_i^{(1)}, \ldots, a_i^{(n-2)}).$$

(Here, $\delta_{i,j}$ is the Kronecker-δ symbol.) A straightforward computation reveals that this choice for $B_0, \ldots, B_{n-1}$ satisfies (1). It is also not hard to show that these matrices $B_0, \ldots, B_{n-1}$ are the only matrices satisfying (1), and that each B_i is necessarily invertible.

Now, since φ: $GF(q) \to GF(p)$ is linear and not identically zero, we must have $\varphi(x) = \sum_{i=0}^{n-1} \lambda_i x_i$ for each $x \in GF(q)$ for some choice of $\lambda_0, \ldots, \lambda_{n-1} \in GF(p)$ (not all 0). At this point we see that $\varphi(xy) = \vec{x}^{\mathsf{T}} M_\varphi \vec{y}$ for $M_\varphi = \sum_{i=0}^{n-1} \lambda_i B_i$. Equivalently, we have

$$M_\varphi = \text{Hankel}\left(\lambda_0, \ldots, \lambda_{n-1}, \sum_{i=0}^{n-1} \lambda_i a_i^{(0)}, \ldots, \sum_{i=0}^{n-1} \lambda_i a_i^{(n-2)}\right). \qquad \square$$

In the previous theorem, we have ignored the issue of circuit uniformity. However, it follows from the proof that each circuit for $F_{q,\varphi}$ can be generated in polynomial time under a similar assumption on the circuits for performing F_p.

3. The Hidden Linear Structure Problem. For a prime power q, define the *hidden linear structure* problem over $GF(q)$ as follows. In the classical version, one is given a

black-box that maps $(x, y) \in GF(q) \times GF(q)$ to $(x, \pi(y + sx))$, where π is an arbitrary permutation on the elements of $GF(q)$ and $s \in GF(q)$. Analogously, in the quantum case, one is given a black-box performing the unitary transformation that maps $|x\rangle|y\rangle$ $(x, y \in GF(q))$ to $|x\rangle|\pi(y + sx)\rangle$. The goal is to determine the value of s.

In this section we give a sharp quantum versus classical query complexity separation for the hidden linear structure problem. First, in the classical case, $\Omega(\sqrt{q})$ queries are necessary to solve this problem, even with bounded error. Second, in the quantum case, a single quantum query is sufficient to solve the hidden linear structure problem exactly, provided that one can compute the QFTs $F_{q,\varphi}$ and $F_{q,\varphi}^{\dagger}$. In the case where $q = 2^n$, the QFT can be performed exactly with only $O(n^2)$ basic operations (Hadamard gates and controlled-NOT gates). The result is a single-query exact quantum algorithm to extract s with $O(n^2)$ auxiliary operations. Moreover, in this case the algorithm can be streamlined to consist of $O(n)$ Hadamard gates, the single query, and $O(n^2)$ classical post-processing after a measurement is made. In the case where q is an n-bit prime, our results are weaker, since the best procedure that we are aware of for performing the QFT exactly in that case is $O(p^2 \log p) = O(n4^n)$.

It should be noted that if the finite fields are relaxed to *finite rings* then, for the analogous hidden linear structure problem, the quantum versus classical query complexity separation may be much weaker. This is because the classical query complexity of the problem can become much smaller. For example, for the ring $\mathbb{Z}_{2^n}$, there is a simple classical procedure solving the hidden linear structure problem with only $n + 1$ queries. It begins by querying $(0, 0)$ and $(2^{n-1}, 0)$, yielding $\pi(0)$ and $\pi(s2^{n-1})$, respectively. If $\pi(0) = \pi(s2^{n-1})$, then s is even; otherwise s is odd. Thus, two queries reduce the number of possibilities for s by a factor of 2. If s is even, then the next query is $(2^{n-2}, 0)$, yielding $\pi(s2^{n-2})$, which determines whether s mod 4 is 0 or 2. If s is odd, then the next query is $(2^{n-2}, 2^n - 2^{n-2})$, yielding $\pi(2^n - 2^{n-2} + s2^{n-2})$, which determines whether s mod 4 is 1 or 3. This process can be continued to deduce s after $n + 1$ queries. For this reason, our attention is focused on the hidden linear structure problem over fields (though we do consider QFTs for some noncommutative rings in the next section).

We proceed with the classical lower bound.

THEOREM 3. *$\Omega(\sqrt{q})$ queries are necessary to solve the hidden linear structure problem over $GF(q)$ within error probability $\frac{1}{2}$.*

PROOF. The lower bound proof is similar to that for Simon's problem [18]. First, by a game-theoretic argument [19], it suffices to consider deterministic algorithms where the input data, embodied by the values of s and π, is probabilistic. Set both $s \in GF(q)$ and π (a permutation on $GF(q)$) randomly, according to the uniform distribution. Consider the information obtained about s after k queries $(x_1, y_1), \ldots, (x_k, y_k)$ (without loss of generality, the queries are all distinct). If, for some $i \neq j$, the outputs of the ith and jth queries collide in that $\pi(y_i + sx_i) = \pi(y_j + sx_j)$, then $y_i + sx_i = y_j + sx_j$, which implies that the value of s can be determined as

$$s = \frac{y_i - y_j}{x_j - x_i} \tag{3}$$

(note that $x_j - x_i \neq 0$, since this would imply that $(x_i, y_i) = (x_j, y_j)$). On the other hand, if there are no collisions among the outputs of all k queries, then all that can be

deduced about s is that

$$s \neq \frac{y_i - y_j}{x_j - x_i} \tag{4}$$

for all $1 \leq i < j \leq k$. This leaves $q - k(k-1)/2$ values for s, which are equally likely by symmetry.

Now, consider the probability of a collision occurring at the kth query given that no collisions have occurred in the previous $k-1$ queries. After the first $k-1$ queries, there remain at least $q - (k-1)(k-2)/2 > q - k^2/2$ possible values of s, equally likely by symmetry. Of these values, at most $k-1$ induce a collision between the kth query and one of the $k-1$ previous queries. Therefore, the probability of a collision occurring at the kth query is at most

$$\frac{k-1}{q - k^2/2} \leq \frac{2k}{2q - k^2}. \tag{5}$$

It follows that the probability of a collision occurring at all during the first l queries is bounded above by

$$\sum_{k=1}^{l} \frac{2k}{2q - k^2} \leq \frac{l^2}{2q - l^2}. \tag{6}$$

If this probability is to be greater than or equal to $\frac{1}{2}$, then $l^2/(2q - l^2) \geq \frac{1}{2}$, which implies that

$$l \geq \sqrt{2q/3} \in \Omega(\sqrt{q}). \tag{7}$$

□

Next, we describe the quantum algorithm.

THEOREM 4. *For a given field $GF(q)$, if $F_{q,\varphi}$ and $F_{q,\varphi}^{\dagger}$ can be performed for some nonzero linear mapping φ, then a single query is sufficient to solve the hidden linear structure problem exactly.*

PROOF. The quantum procedure is to initialize the state of two $GF(q)$-valued registers to $|0\rangle|1\rangle$ (where 0 and 1 are respectively the additive and multiplicative identities of the field) and perform the following operations (where $F = F_{q,\varphi}$):

1. Apply $F \otimes F^{\dagger}$.
2. Query the black-box.
3. Apply $F^{\dagger} \otimes F$.

Then the state of the first register is measured.

Tracing through the evolution of the state of the registers during the execution of the above algorithm, the state after each step is:

1. $(F|0\rangle)(F^{\dagger}|1\rangle)$,
2. $(F|s\rangle)(U_{\pi}F^{\dagger}|1\rangle)$,
3. $|s\rangle(FU_{\pi}F^{\dagger}|1\rangle)$.

The transformation from step 1 to step 2 follows from the control/target inversion property, as shown in Figure 1. It is clear that the output of the algorithm is s. □

As mentioned previously, the transformation $F_{2^n,\varphi}$ for any φ is particularly simple, and yields the following algorithm:

1. Initialize the state of two $GF(2^n)$-valued registers to the (classical) state $|0\rangle|M_\varphi \vec{1}\rangle$.
2. Apply a Hadamard transform to each qubit of each register.
3. Query the black-box.
4. Apply a Hadamard transform to each qubit of each register.
5. Measure the first register, yielding an n-bit string z.
6. Classically, compute $(M_\varphi^{\mathsf{T}})^{-1}\vec{z}$.

The result will be s.

4. Extension to Rings. It is natural to generalize the concept of controlled addition as we have seen it to rings in general. So, one might ask whether, for all rings, there exist operations corresponding to "quantum Fourier transforms" in the sense that they perform control/target inversion on controlled-addition gates over that ring. While we do not know the answer to this question, we will show that for any commutative ring R where such a Fourier transform exists, it is possible to define QFTs for the noncommutative ring of $m \times m$ matrices over R.

We introduce some notation. In this section all matrices are understood to be square matrices. Given an m^2 array of quantum registers $\{E_{ij}\}$ over a commutative ring R, we associate the state $|x_{ij}\rangle$ with the register E_{ij}. We also identify the $m \times m$ matrix X given by

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mm} \end{bmatrix}$$

with the product state

$$|X\rangle = \bigotimes_{i=1}^{m} \bigotimes_{j=1}^{m} |x_{ij}\rangle = |x_{11}\rangle|x_{12}\rangle \cdots |x_{1m}\rangle|x_{21}\rangle \cdots |x_{mm}\rangle$$

of the states of the registers. We then make the following definition.

DEFINITION 4.1. Let F_R be a QFT over a commutative ring R. Then we define the QFT over $R^{m\times m}$ by the following mapping for each matrix $X = (x_{ij}) \in R^{m\times m}$:

$$F_{R,m}\colon\ |X\rangle \mapsto \bigotimes_{i=1}^{m} \bigotimes_{j=1}^{m} F_R|x_{ji}\rangle.$$

That is, the QFT of $|X\rangle$ is performed by applying the Fourier transform F_R independently to all the quantum registers used to represent X, and transposing those registers (or their states) within the register array.

Multiplication in matrix rings over R will, in general, be noncommutative. Therefore, in working with matrices, we must distinguish between left and right multiplication when defining the controlled addition operators. We define left-controlled addition with parameter S (denoted by C_{S*}) and right-controlled addition with parameter S (denoted by C_{*S}) by the following action on basis states:

$$C_{S*}\colon |X\rangle|Y\rangle \mapsto |X\rangle|Y + SX\rangle, \qquad C_{*S}\colon |X\rangle|Y\rangle \mapsto |X\rangle|Y + XS\rangle.$$

As well, we introduce left- and right-controlled addition operators with the roles of the target and control registers reversed:

$$D_{S*}\colon |X\rangle|Y\rangle \mapsto |X + SY\rangle|Y\rangle, \qquad D_{*S}\colon |X\rangle|Y\rangle \mapsto |X + YS\rangle|Y\rangle.$$

As the order of multiplication becomes important for rings in general, we find it reasonable to make the following expansion of the definition of control/target inversion: a gate G performs control/target inversion on controlled addition gates over a given ring if the following equality holds:

$$(G^\dagger \otimes G)C_{S*}(G \otimes G^\dagger) = D_{*S}.$$

That is, in addition to the roles of target and control being interchanged, the manner of multiplication (left or right) is switched. In the case where the ring is commutative, this reduces to the definition given previously (see Figure 1). We now show that the QFT $F_{R,m}$ defined above has this property for $m \times m$ matrices over R, when F_R is defined and has the control/target inversion property on R.

For input matrices X and Y over R, we denote

$$|X\rangle = \bigotimes_{i=1}^{m}\bigotimes_{j=1}^{m}|x_{ij}\rangle, \qquad |Y\rangle = \bigotimes_{i=1}^{m}\bigotimes_{j=1}^{m}|y_{ij}\rangle.$$

Let E_{ij} represent the register which stores the state $|x_{ij}\rangle$, and let F_{ij} represent the register which stores the state $|y_{ij}\rangle$. Define the operator $A_{ik}^{ij}(s)$ as a controlled-ADD$_s$ gate which operates on a control register E_{ik} and a target register F_{ij}, and $B_{ik}^{ij}(s)$ as a controlled-ADD$_s$ gate which operates on a control register F_{ik} and a target register E_{ij}. Then we can decompose C_{S*} as the following product of operators:

$$C_{S*} = \prod_{i=1}^{m}\prod_{j=1}^{m}\prod_{k=1}^{m} A_{ij}^{ik}(s_{kj}).$$

This can be easily verified by testing the effect of this product on the ijth target register, where we see that the effect (for basis states) is to add the term $x_{ik}s_{kj}$ for each $1 \le k \le m$. Control/target inversion is expressed for these gates in the following manner:

$$({F_R}^{\dagger\otimes m^2} \otimes {F_R}^{\otimes m^2})A_{ij}^{ik}(s_{kj})({F_R}^{\otimes m^2} \otimes {F_R}^{\dagger\otimes m^2}) = B_{ij}^{ik}(s_{kj}).$$

Here, the QFTs cancel one another out on all registers except the ijth target register and the ikth control register, where control/target inversion occurs.

Using this decomposition, and applying QFTs to the individual registers before and after this product of gates in the same manner as above, we obtain

$$\begin{aligned}(F_R{}^{\dagger\otimes m^2} \otimes F_R{}^{\otimes m^2}) C_{S*} (F_R{}^{\otimes m^2} \otimes F_R{}^{\dagger\otimes m^2}) &= \prod_{i=1}^{m}\prod_{j=1}^{m}\prod_{k=1}^{m} B_{ij}^{ik}(s_{kj}) \\ &= \prod_{i=1}^{m}\prod_{k=1}^{m}\prod_{j=1}^{m} B_{ij}^{ik}(s_{kj}) \\ &= \prod_{i=1}^{m}\prod_{j=1}^{m}\prod_{k=1}^{m} B_{ik}^{ij}(s_{jk}) = D_{S^{\mathsf{T}}*}.\end{aligned}$$

That is, the roles of the control and target registers are reversed, and although the manner of multiplication is unchanged, the parameter matrix S is transposed.

Note that the QFT $F_{R,m}$ on $m \times m$ matrices over R can be decomposed into an application of F_R on each element of the matrix, and transposing the matrix (denoted by the operator T_m), in any order:

$$F_{R,m} = (F_R{}^{\otimes m^2}) T_m = T_m (F_R{}^{\otimes m^2}).$$

Clearly, $T_m T_m = I_m$ (the identity $m \times m$ matrix). Then we can verify that $F_{R,m}$ performs control/target inversion on controlled addition gates over $R^{m\times m}$:

$$\begin{aligned}&(F_{R,m}^{\dagger} \otimes F_{R,m}) C_{S*} (F_{R,m} \otimes F_{R,m}^{\dagger}) |X\rangle|Y\rangle \\ &\quad = (T_m \otimes T_m)(F_R^{\dagger\otimes m^2} \otimes F_R{}^{\otimes m^2}) C_{S*} (F_R{}^{\otimes m^2} \otimes F_R^{\dagger\otimes m^2})(T_m \otimes T_m)|X\rangle|Y\rangle \\ &\quad = (T_m \otimes T_m) D_{S^{\mathsf{T}}*} |X^{\mathsf{T}}\rangle|Y^{\mathsf{T}}\rangle \\ &\quad = (T_m \otimes T_m)|X^{\mathsf{T}} + S^{\mathsf{T}} Y^{\mathsf{T}}\rangle|Y^{\mathsf{T}}\rangle \\ &\quad = |X + YS\rangle|Y\rangle \\ &\quad = D_{*S}|X\rangle|Y\rangle,\end{aligned}$$

which is what we wished to show.

As for extending the hidden linear structure problem to arbitrary rings, it is not clear for which rings R an exponential separation can be achieved. The ability to perform control/target inversion for this problem when $R = GF(p^n)^{m\times m}$ (for example) indicates that the problem can be solved in one query in the quantum case, but we do not have strong classical lower bounds for this case. However, there do exist rings, such as $GF(p^n) \times GF(p^n)$, where exponential separation can be shown, building on the proof for $GF(p^n)$; thus, the strong separation in the case of finite fields is not an isolated case. Considering the proofs of the classical upper bound for $\mathbb{Z}_{p^n}$ and lower bound for $GF(p^n)$, it seems plausible that rings exhibiting a strong separation will have very few zero divisors, or little additive structure among the zero divisors. Both of these statements hold for $GF(p^n) \times GF(p^n)$, which has a ratio of $O(1/p^n)$ zero divisors among its elements, and which only has two ideals which have only a trivial intersection.

Acknowledgments. R.C. gratefully acknowledges the University of California at Berkeley and the California Institute of Technology where some of the writing and revisions to this paper occurred.

References

[1] A. Barenco, C. H. Bennett, R. Cleve, D. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. Smolin, and H. Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52:3457–3467, 1995.

[2] J. N de Beaudrap, R. Cleve, and J. Watrous, Quantum Fourier transforms for extracting hidden linear structures in finite fields. Los Alamos Preprint Archive quant-ph/0011065, 2000.

[3] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[4] D. Boneh and R. Lipton. Quantum cryptanalysis of hidden linear functions. In *Advances in Cryptology – Crypto '*95, volume 963 of Lecture Notes in Computer Science, pages 242–437. Springer-Verlag, Berlin, 1995.

[5] G. Brassard and P. Høyer. An exact quantum polynomial-time algorithm for Simon's problem. In *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*, pages 12–23, 1997.

[6] R. Cleve. The query complexity of order-finding. In *Proceedings of the* 15*th Annual IEEE Conference on Computational Complexity*, pages 54–59, 2000.

[7] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca. Quantum algorithms revisited. *Proceedings of the Royal Society*, *London*, *Series A*, 454:339–354, 1998.

[8] H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer-Verlag, New York, 1993.

[9] W. van Dam and S. Hallgren. Efficient quantum algorithms for shifted quadratic character problems. Los Alamos Preprint Archive quant-ph/0011067, 2000.

[10] D. Deutsch. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society*, *London*, *Series A*, 400:97–117, 1985.

[11] D. Deutsch and R. Jozsa. Rapid solutions of problems by quantum computation. *Proceedings of the Royal Society*, *London*, *Series A*, 439:553–558, 1992.

[12] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, 1999.

[13] L. Hales and S. Hallgren. An improved quantum Fourier transform algorithm and applications. In *Proceedings of the* 41*st Annual Symposium on Foundations of Computer Science*, pages 515–525, 2000.

[14] S. Hallgren. Personal communication, 2001.

[15] R. Lidl and H. Niederreiter. *Introduction to Finite Fields and Their Applications*, revised edition. Cambridge University Press, Cambridge, 1994.

[16] M. Mosca and A. Ekert. The hidden subgroup problem and eigenvalue estimation on a quantum computer. In *Proceedings of the* 1*st NASA International Conference on Quantum Computing and Quantum Communication*, volume 1509 of Lecture Notes in Computer Science, pages 174–188. Springer-Verlag, Berlin, 1999. Also available from the Los Alamos Preprint Archive quant-ph/9903071.

[17] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[18] D. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[19] A. C.-C. Yao. Lower bounds by probabilistic arguments. In *Proceedings of the* 24*th Annual Symposium on Foundations of Computer Science*, pages 420–428, 1983.