Quantum Algorithms for some Hidden Shift Problems

Wim van Dam*
HP, MSRI, U.C. Berkeley
vandam@cs.berkeley.edu

Sean Hallgren†
Caltech
hallgren@cs.caltech.edu

Lawrence Ip‡
U.C. Berkeley
lip@eecs.berkeley.edu

Abstract

Almost all of the most successful quantum algorithms discovered to date exploit the ability of the Fourier transform to recover subgroup structure of functions, especially periodicity. The fact that Fourier transforms can also be used to capture shift structure has received far less attention in the context of quantum computation.

In this paper, we present three examples of "unknown shift" problems that can be solved efficiently on a quantum computer using the quantum Fourier transform. We also define the hidden coset problem, which generalizes the hidden shift problem and the hidden subgroup problem. This framework provides a unified way of viewing the ability of the Fourier transform to capture subgroup and shift structure.

1 Introduction

The first problem to demonstrate a superpolynomial separation between random and quantum polynomial time was the Recursive Fourier Sampling problem [6]. Exponential separations were subsequently discovered by Simon [32], who gave an oracle problem, and by Shor [31], who found polynomial time quantum algorithms for factoring and discrete log. We now understand that the natural generalization of Simon's problem and the factoring and discrete log problems is the hidden subgroup problem (HSP), and that when the underlying group is Abelian and finitely generated, we can solve the HSP efficiently on a quantum computer. While recent results have continued to study important generalizations of the HSP (for example, [17, 23, 19, 34, 25, 22]), only the Recursive Fourier Sampling problem remains outside the HSP framework.

In this paper, we give quantum algorithms for several hidden shift problems. In a hidden shift problem we are given two functions f, g such that there is a shift s for which $f(x) = g(x+s)$ for all x. The problem is then to find s. We show how to solve this problem for several classes of functions, but perhaps the most interesting example is the shifted Legendre symbol problem, where g is the Legendre symbol[1] with respect to a prime size finite field, and the problem is then: "Given the function $f(x) = \left(\frac{x+s}{p}\right)$ as an oracle, find s".

The oracle problem our algorithms solve can be viewed as the problem of predicting a pseudo-random function f. Such tasks play an important role in cryptography and have been studied extensively under various assumptions about how one is allowed to query the function (nonadaptive versus adaptive, deterministic versus randomized, et cetera) [7, 29]. In this paper we consider the case where the function is queried in a quantum mechanical superposition of different values x. We show that if $f(x)$ is an s-shifted multiplicative character $\chi(x+s)$, then a polynomial-time quantum algorithm making such queries can determine the hidden shift s, breaking the pseudo-randomness of f. We conjecture that classically the shifted Legendre symbol is a pseudo-random function, that is, it is impossible to efficiently predict the value of the function after a polynomial number of queries if one is only allowed a classical algorithm with oracle access to f. Partial evidence for this conjecture has been given by Damgård [15] who proposed the related task: "Given a part of the Legendre sequence $\left(\frac{s}{p}\right), \left(\frac{s+1}{p}\right), \ldots, \left(\frac{s+\ell}{p}\right)$, where ℓ is $O(\log p)$, predict the next value $\left(\frac{s+\ell+1}{p}\right)$", as a hard problem with applications in cryptography.

Using the quantum algorithms presented in this paper, we can break certain algebraically homomorphic

*HP Labs, Palo Alto; Mathematical Sciences Research Institute; and Computer Science Division, University of California, Berkeley. Supported by an HP - MSRI postdoc fellowship.

†Supported in part by an NSF Mathematical Sciences Postdoctoral Fellowship and in part by the NSF through the Institute for Quantum Information at the California Institute of Technology. Most of this work done while the author was at the Mathematical Sciences Research Institute and the University of California, Berkeley, with partial support from DARPA QUIST Grant F30602-01-2-0524.

‡Computer Science Division, University of California, Berkeley. Supported by NSF Grant CCR-0049092, DARPA Grant F30602-00-2-0601 and DARPA QUIST Grant F30602-01-2-2054. Part of this work was done while the author was a visitor at the Institute for Quantum Information at the California Institute of Technology.

[1] The Legendre symbol $\left(\frac{x}{p}\right)$ is defined to be 0 if p divides x, 1 if x is a quadratic residue mod p and -1 if x is not a quadratic residue mod p.

cryptosystems by a reduction to the shifted Legendre symbol problem. The best known classical algorithm [9] for breaking these cryptosystems is subexponential and is based on a smoothness assumption. These cryptosystems can also be broken by Shor's algorithm for period finding, but the two attacks on the cryptosystems appear to use completely different ideas.

While current quantum algorithms solve problems based on an underlying group and the Fourier transform over that group, we initiate the study of problems where there is an underlying ring or field. The Fourier transform over the additive group of the ring is defined using the characters of the additive group, the additive characters of the ring. Similarly, the multiplicative group of units induces multiplicative characters of the ring. The interplay between additive and multiplicative characters is well understood [28, 33], and we show that this connection can be exploited in quantum algorithms. In particular, we put a multiplicative character into the phase of the registers and compute the Fourier transform over the additive group. The resulting phases are the inner products between the multiplicative character and each of the additive characters, a Gauss sum. We hope the new tools presented here will lead to other quantum algorithms.

We give algorithms for three types of hidden shift problems:

In the first problem, g is a multiplicative character of a finite field. Given f, a shifted version of g, the shift is uniquely determined from f and g. An example of a multiplicative character of $\mathbb{Z}/p\mathbb{Z}$ is the Legendre symbol. Our algorithm uses the Fourier transform over the additive group of a finite field.

In the second problem, g is a multiplicative character of the ring $\mathbb{Z}/n\mathbb{Z}$. This problem has the feature that the shift is not uniquely determined by f and g and our algorithm identifies all possible shifts. An example of a multiplicative character of $\mathbb{Z}/n\mathbb{Z}$ is the Jacobi symbol[2].

In the third problem we have the same setup as in the second problem with the additional twist that n is unknown.

We also define the *hidden coset problem*, which is a generalization of the hidden shift problem and the hidden subgroup problem. This definition provides a unified way of viewing the quantum Fourier transform's ability to capture subgroup and shift structure.

Some of our hidden shift problems can be reduced to the HSP, although efficient algorithms for these HSP instances are not known. Assuming Conjecture 2.1 from [9], the shifted Legendre symbol problem over $\mathbb{Z}/p\mathbb{Z}$ can be reduced to an instance of the HSP over the dihedral group $D_p = \mathbb{Z}/p\mathbb{Z} \rtimes \mathbb{Z}/2\mathbb{Z}$ in the following way. Let $f(x,0) = (\left(\frac{x}{p}\right), \left(\frac{x+1}{p}\right), \ldots, \left(\frac{x+\ell}{p}\right))$ and $f(x,1) = (\left(\frac{x+s}{p}\right), \left(\frac{x+s+1}{p}\right), \ldots, \left(\frac{x+s+\ell}{p}\right))$, where s is unknown and $\ell > 2\log^2 p$. Then the hidden subgroup is $H = \{(0,0),(s,1)\}$. This conjecture is necessary to ensure that f will be distinct on distinct cosets of H. For the general shifted multiplicative character problem, the analogous reduction to the HSP may fail because f may not be distinct on distinct cosets. However, we can efficiently generate random coset states, that is, superpositions of the form $|x,0\rangle + |x+s,1\rangle$, although it is unknown how to use these to efficiently find s [16]. The issue of nondistinctness on cosets in the HSP has been studied for some groups [8, 21, 20, 18].

The existence of a time efficient quantum algorithm for the shifted Legendre symbol problem was posed as an open question in [12]. The Fourier transform over the additive group of a finite field was independently proposed for the solution of a different problem in [4]. The current paper subsumes [13] and [24]. Building on the ideas in this paper, a quantum algorithm for estimating Gauss sums is described in [14].

This paper is organized as follows. Section 2 contains some definitions and facts. In Section 3, we give some intuition for the ideas behind the algorithms. In Section 4, we present an algorithm for the shifted multiplicative problem over finite fields, of which the shifted Legendre symbol problem is a special case, and show how we can use this algorithm to break certain algebraically homomorphic cryptosystems. In Section 5, we extend our algorithm to the shifted multiplicative problem over rings $\mathbb{Z}/n\mathbb{Z}$. This has the feature that unlike in the case of the finite field, the possible shifts may not be unique. We then show that this algorithm can be extended to the situation where n is unknown. In Section 6, we show that all these problems lie within the general framework of the hidden coset problem. We give an efficient algorithm for the hidden coset problem provided g satisfies certain conditions. We also show how our algorithm can be interpreted as solving a deconvolution problem using Fourier transforms.

2 Background

2.1 Notation and Conventions We use the following notation: ω_n is the nth root of unity $\exp(2\pi i/n)$, and $\hat{f}$ denotes the Fourier transform of the function f. An algorithm computing in $\mathbb{F}_q$, $\mathbb{Z}/n\mathbb{Z}$ or G runs in polynomial time if it runs in time polynomial in $\log q$, $\log n$ or $\log |G|$.

In a ring $\mathbb{Z}/n\mathbb{Z}$ or a field $\mathbb{F}_q$, additive characters ψ ($\mathbb{Z}/n\mathbb{Z} \to \mathbb{C}^*$ or $\mathbb{F}_q \to \mathbb{C}^*$) are characters of the

[2] The Jacobi symbol $\left(\frac{a}{b}\right)$ is defined so that it satisfies the relation $\left(\frac{a}{bc}\right) = \left(\frac{a}{b}\right)\left(\frac{a}{c}\right)$ and reduces to the Legendre symbol when the lower parameter is prime.

additive group, that is, $\psi(x+y) = \psi(x)\psi(y)$, and multiplicative characters χ ($(\mathbb{Z}/n\mathbb{Z})^* \to \mathbb{C}^*$ or $\mathbb{F}_q^* \to \mathbb{C}^*$) are characters of the multiplicative group of units, that is, $\chi(xy) = \chi(x)\chi(y)$ for all x and y. We extend the definition of a multiplicative character to the entire ring or field by assigning the value zero to elements outside the unit group. All nonzero $\chi(x)$ values have unit norm and so $\chi(x^{-1}) = \overline{\chi(x)}$.

We ignore the normalization term in front of a superposition unless we need to explicitly calculate the probability of measuring a particular value.

2.2 Computing Superpositions We will need to compute the superposition $\sum_x f(x)|x\rangle$ where $f(x)$ is in the *amplitude*.

LEMMA 2.1. (COMPUTING SUPERPOSITIONS) *Let $f : G \to \mathbb{C}$ be a complex-valued function defined on the set G such that $f(x)$ has unit magnitude whenever $f(x)$ is nonzero. Then there is an efficient algorithm for creating the superposition $\sum_x f(x)|x\rangle$ with success probability equal to the fraction of x such that $f(x)$ is nonzero and that uses only two queries to the function f.*

Proof. Start with the superposition over all x, $\sum_x |x\rangle$. Compute $f(x)$ into the second register and measure to see whether $f(x)$ is nonzero. This succeeds with probability equal to the fraction of x such that $f(x)$ is nonzero. Then we are left with a superposition over all x such that $f(x)$ is nonzero. Compute the phase of $f(x)$ into the phase of $|x\rangle$. This phase computation can be approximated arbitrarily closely by approximating the phase of $f(x)$ to the nearest 2^nth root of unity for sufficiently large n. Use a second query to f to reversibly uncompute the $f(x)$ from the second register.

2.3 Approximate Fourier Sampling It is not known how to efficiently compute the quantum Fourier transform over $\mathbb{Z}/n\mathbb{Z}$ exactly. However, efficient approximations are known [26, 27, 11, 21]. We can even compute an efficient approximation to the distribution induced when n is unknown as long as we have an upper bound on n [21]. We will need to approximately Fourier sample to solve the unknown n case of the shifted character problem in Section 5.2.

To Fourier sample a state $|\phi\rangle$, we form the state $|\tilde{\phi}\rangle$ that is the result of repeating $|\phi\rangle$ many times. We then Fourier sample from $|\tilde{\phi}\rangle$ and use continued fractions to reduce the expanded range of values. This expansion into $|\tilde{\phi}\rangle$ allows us to perform the Fourier sampling step over a length from which we *can* exactly Fourier sample.

More formally, let $|\phi\rangle = \sum_{x=0}^{n-1} \phi_x|x\rangle$ be an arbitrary superposition, and $\hat{\mathcal{D}}_{|\phi\rangle}$ be the distribution induced by Fourier sampling $|\phi\rangle$ over $\mathbb{Z}_n$. Let the superposition $|\tilde{\phi}\rangle = \sum_{x=0}^{m-1} \phi_{x \bmod n}|x\rangle$ be $|\phi\rangle$ repeated until some arbitrary integer m, not necessarily a multiple of n. Let $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ be the distribution induced by Fourier sampling $|\tilde{\phi}\rangle$ over $\mathbb{Z}_q$ rather than $\mathbb{Z}_m$ (where $q > m$ and $\phi_x = 0$ if $x \geq m$). Notice that $\hat{\mathcal{D}}_{|\phi\rangle}$ is a distribution on $\mathbb{Z}_n$ and $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ is a distribution on $\mathbb{Z}_q$.

We can now define the two distributions we will compare. Let $\hat{\mathcal{D}}^{\text{RF}}_{|\phi\rangle}$ be the distribution induced on the reduced fractions of $\hat{\mathcal{D}}_{|\phi\rangle}$, that is, if x is a sample from $\hat{\mathcal{D}}_{|\phi\rangle}$, we return the fraction x/n in lowest terms. In particular, define $\hat{\mathcal{D}}^{\text{RF}}_{|\phi\rangle}(j,k) = \hat{\mathcal{D}}_{|\phi\rangle}(jm)$ if $mk = n$. Let $\hat{\mathcal{D}}^{\text{CF}}_{|\tilde{\phi}\rangle}$ be the distribution induced on fractions from sampling $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ to obtain x, and then using continued fractions to compute the closest approximation to x/q with denominator at most n. If $m = \Omega(\frac{n^2}{\epsilon^2})$ and $q = \Omega(\frac{m}{\epsilon})$, then $|\hat{\mathcal{D}}^{\text{RF}}_{|\phi\rangle} - \hat{\mathcal{D}}^{\text{CF}}_{|\tilde{\phi}\rangle}|_1 < \epsilon$.

2.4 Finite Fields The elements of a finite field $\mathbb{F}_q$ (where $q = p^r$ for some prime p) can be represented as polynomials in $\mathbb{F}_p[X]$ modulo a degree r irreducible polynomial in $\mathbb{F}_p[X]$. In this representation, addition, subtraction, multiplication and division can all be performed in $O((\log q)^2)$ time [2].

We will need to compute the Fourier transform over the additive group of a finite field, which is isomorphic to $(\mathbb{Z}/p\mathbb{Z})^r$. The additive characters are of the form $\psi_y(x) = \omega_p^{\text{Tr}(xy)}$, where $\text{Tr} : \mathbb{F}_q \to \mathbb{F}_p$ is the trace of the finite field $\text{Tr}(x) = \sum_{j=0}^{r-1} x^{p^j}$, and $y \in \mathbb{F}_q$ [28]. We can efficiently compute the Fourier transform over the additive group of a finite field.

LEMMA 2.2. (FOURIER TRANSFORM OVER $\mathbb{F}_q$) *The Fourier transform $|x\rangle \mapsto \frac{1}{\sqrt{q}}\sum_{y\in\mathbb{F}_q} \omega_p^{\text{Tr}(xy)}|y\rangle$ can be approximated to within error ϵ in time polynomial in $\log q$ and $\log 1/\epsilon$.*

Proof. See [13]. (Independently, the efficiency of this transform was also shown in [4].)

For clarity of exposition we assume throughout the rest of the paper that this Fourier transform can be performed exactly, as we can make the errors due to the approximation exponentially small with only polynomial overhead.

2.5 Multiplicative Characters and their Fourier Transforms The multiplicative group $\mathbb{F}_q^*$ of a finite field $\mathbb{F}_q$ is cyclic. Let g be a generator of $\mathbb{F}_q^*$. Then the multiplicative characters of $\mathbb{F}_q$ are of the form $\chi(g^\ell) = \omega_{q-1}^{k\ell}$ for all $\ell \in \{0, \ldots, q-2\}$ where the $q-1$ different multiplicative characters are indexed by

$k \in \{0, \ldots, q-2\}$. The trivial character is the character with $k = 0$. We can extend the definition of χ to $\mathbb{F}_q$ by defining $\chi(0) = 0$. On a quantum computer we can efficiently compute $\chi(x)$ because the value is determined by the discrete logarithm $\log_g(x)$, which can be computed efficiently using Shor's algorithm [31]. The Fourier transform of a multiplicative character χ of the finite field $\mathbb{F}_q$ is given by $\hat{\chi}(y) = \overline{\chi(y)}\hat{\chi}(1)$ [28, 33].

Let $n = p_1^{m_1} \ldots p_k^{m_k}$ be the prime factorization of n. Then by the Chinese Remainder Theorem, $(\mathbb{Z}/n\mathbb{Z})^* \cong (\mathbb{Z}/p_1^{m_1}\mathbb{Z})^* \times \cdots \times (\mathbb{Z}/p_k^{m_k}\mathbb{Z})^*$. Every multiplicative character χ of $\mathbb{Z}/n\mathbb{Z}$ can be written as the product $\chi(x) = \chi_1(x_1) \ldots \chi_k(x_k)$, where χ_i is a multiplicative character of $\mathbb{Z}/p_i^{m_i}\mathbb{Z}$ and $x_i \equiv x \bmod p_i^{m_i}$. We say χ is *completely nontrivial* if each of the χ_i is nontrivial. We extend the definition of χ to all of $\mathbb{Z}/n\mathbb{Z}$ by defining $\chi(y) = 0$ if $\gcd(y, n) \neq 1$. The character χ is aperiodic on $\{0, \ldots, n-1\}$ if and only if all its χ_i factors are aperiodic over their respective domains $\{0, \ldots, p_i^{m_i} - 1\}$. We call χ a *primitive character* if it is completely nontrivial and aperiodic. Hence, χ is primitive if and only if all its χ_i terms are primitive.

It is well known that the Fourier transform of a primitive χ is $\hat{\chi}(y) = \overline{\chi(y)}\hat{\chi}(1)$. If χ is completely nontrivial but periodic with period ℓ, then its Fourier transform obeys $\hat{\chi}(yn/\ell) = \overline{\chi'(y)}\hat{\chi}'(1)$, where χ' is the primitive character obtained by restricting χ to $\{0, \ldots, \ell - 1\}$. See the book by Tolimieri et al. for details [33].

3 The Intuition Behind the Algorithms for the Hidden Shift Problem

We give some intuition for the ideas behind our algorithms for the hidden shift problem. We use the shifted Legendre symbol problem as our running example, but the approach works more generally. In the shifted Legendre symbol problem we are given a function $f_s : \mathbb{Z}_p \to \{0, \pm 1\}$ such that $f(x) = \left(\frac{x+s}{p}\right)$, and are asked to find s. The Legendre symbol $\left(\frac{\cdot}{p}\right) : \mathbb{F}_p \to \{0, \pm 1\}$ is the quadratic multiplicative character of $\mathbb{F}_p$ defined: $\left(\frac{x}{p}\right)$ is 1 if x is a square modulo p, -1 if it is not a square, and 0 if $x \equiv 0$.

The algorithm starts by putting the function value in the phase to get $|f_s\rangle = \sum_x f_s(x)|x\rangle = \sum_x \left(\frac{x+s}{p}\right)|x\rangle$. Assume the functions f_z are mutually (near) orthogonal for different z, so that the inner product $\langle f_z|f_s\rangle$ approximates the delta function value $\delta_s(z)$. Using this assumption, define the (near) unitary matrix C, where the zth row is $|f_z\rangle$. Our quantum state $|f_s\rangle$ is one of the rows, hence $C|f_s\rangle = |s\rangle$. The problem then reduces to: how do we efficiently implement C? By definition, C is a circulant matrix ($c_{x,y} = c_{x+1,y+1}$). Since the Fourier transform matrix diagonalizes a circulant matrix, we can write $C = \mathcal{F}(\mathcal{F}^{-1}C\mathcal{F})\mathcal{F}^{-1} = \mathcal{F}D\mathcal{F}^{-1}$, where D is diagonal. Thus we can implement C if we can implement D. The vector on the diagonal of D is the vector $\mathcal{F}^{-1}|f_0\rangle = \mathcal{F}^{-1}\sum_x \left(\frac{x}{p}\right)|x\rangle$, the inverse Fourier transform of the Legendre symbol. The Legendre symbol is an eigenvector of the Fourier transform, so the diagonal matrix contains the values of the Legendre symbol times a global constant that can be ignored. Because the Legendre symbol can be computed efficiently classically, it can be computed into the phase, so C can be implemented efficiently.

In summary, to implement C for the hidden shift problem for the Legendre symbol, compute the Fourier transform, compute $\left(\frac{x}{p}\right)$ into the phase at $|x\rangle$, and then compute the Fourier transform again (it is not important whether we use $\mathcal{F}$ or $\mathcal{F}^{-1}$).

Figure 1 shows a circuit diagram outlining the algorithm for the hidden shift problem in general. Contrast this with the circuit for the hidden subgroup problem shown in Figure 2.

4 Shifted Multiplicative Characters of Finite Fields

In this section we show how to solve the hidden shift problem for any nontrivial multiplicative character of a finite field. The Fourier transform we use is the Fourier transform over the additive group of the finite field.

Definition 4.1. (Shifted Multiplicative Character Problem over Finite Fields) *Given a nontrivial multiplicative character χ of a finite field $\mathbb{F}_q$ (where $q = p^r$ for some prime p), and a function f for which there is an s such that $f(x) = \chi(x+s)$ for all x. Find s.*

Algorithm 4.1. (Shifted Multiplicative Character Problem over Finite Fields)

1. Create $\sum_{x \in \mathbb{F}_q} \chi(x+s)|x\rangle$.
2. Compute the Fourier transform to obtain $\sum_{y \in \mathbb{F}_q} \omega_p^{\mathrm{Tr}(-sy)}\hat{\chi}(y)|y\rangle$.
3. For all $y \neq 0$, compute $\chi(y)$ into the phase to obtain $\hat{\chi}(1)\sum_{y \in \mathbb{F}_q^*} \omega_p^{\mathrm{Tr}(-sy)}|y\rangle$.
4. Compute the inverse Fourier transform and measure the outcome $-s$.

Theorem 4.1. *For any finite field and any nontrivial multiplicative character, Algorithm 4.1 solves the shifted multiplicative character problem over finite fields with probability $(1 - 1/q)^2$.*

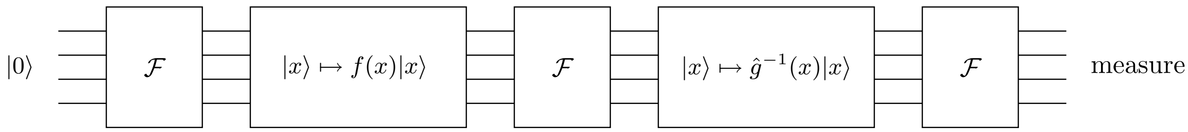

Figure 1: Circuit for hidden shift problem. Notice how we compute f and $\hat{g}^{-1}$ into the *phase*.

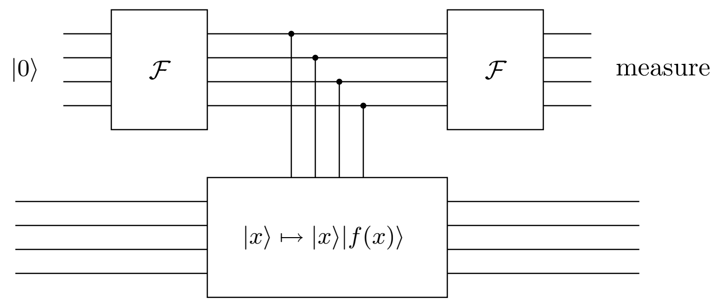

Figure 2: Circuit for hidden subgroup problem. Here f is computed into a *register*.

Proof. 1. Since $\chi(x) = 0$ only at $x = 0$, by Lemma 2.1 we can create the superposition with probability $1 - 1/q$.

2. By Lemma 2.2 we can compute the Fourier transform efficiently. The Fourier transform moves the shift s into the phase as described.

3. Because $\hat{\chi}(y) = \overline{\chi(y)}\hat{\chi}(1)$ for every nonzero y, the phase change $|y\rangle \mapsto \chi(y)|y\rangle$ establishes the required transformation.

4. The amplitude of $|-s\rangle$ is $\frac{1}{\sqrt{q}}\frac{1}{\sqrt{q-1}}\sum_{y\in\mathbb{F}_q^*}\omega_p^{\mathrm{Tr}(-sy)}\omega_p^{\mathrm{Tr}(sy)} = \frac{1}{\sqrt{q}}\frac{1}{\sqrt{q-1}}\sum_{y\in\mathbb{F}_q^*}1$ $= \sqrt{\frac{q-1}{q}}$, so the probability of measuring $-s$ is $1 - 1/q$.

4.1 Example: The Legendre Symbol and Homomorphic Encryption

The Legendre symbol $\left(\frac{\cdot}{p}\right) : \mathbb{F}_p \to \{0, \pm 1\}$ is a quadratic multiplicative character of $\mathbb{F}_p$ defined: $\left(\frac{x}{p}\right)$ is $+1$ if x is a square modulo p, -1 if it is not a square, and 0 if $x = 0$. The quantum algorithm of the previous section showed us how we can determine the shift $s \in \mathbb{F}_p$ given the function $f_s(x) = \left(\frac{x+s}{p}\right)$. We now show how this algorithm enables us to break schemes for 'algebraically homomorphic encryption'.

A cryptosystem is *algebraically homomorphic* if given the encryption of two plaintexts $E(x)$, $E(y)$ with $x, y \in \mathbb{F}_p$, an untrusted party can construct the encryption of the plaintexts $E(x+y)$ and $E(xy)$ in polynomial-time. More formally, we have the secret encryption and decryption functions $E : \mathbb{F}_p \to S$ and $D : S \to \mathbb{F}_p$, in combination with the public add and multiplication transformations $A : S^2 \to S$ and $M : S^2 \to S$ such that $D(A(E(x), E(y))) = x + y$ and $D(M(E(x), E(y))) = xy$ for all $x, y \in \mathbb{F}_p$. We assume that the functions E, D, A and M are deterministic. The decryption function may be many-to-one. As a result the encryption of a given number can vary depending on how the number is constructed. For example, $A(E(4), E(2))$ may not be equal to $M(E(2), E(3))$. In addition to the public A and M functions, we also assume the existence of a zero-tester $Z : S \to \{0, 1\}$, with $Z(E(x)) = 0$ if $x = 0$, and $Z(E(x)) = 1$ otherwise.

An algebraically homomorphic cryptosystem is a cryptographic primitive that enables two players to perform noninteractive secure function evaluation. It is an open problem whether or not such a cryptosystem can be constructed. We say we can break such a cryptosystem if, given $E(s)$, we can recover s in time polylog(p) with the help of the public functions A, M and Z. The best known classical attack, due to Boneh and Lipton [9], has expected running time $O\left(\exp\left(c\sqrt{\log p \log\log p}\right)\right)$ for the field $\mathbb{F}_p$ and is based on a smoothness assumption.

Suppose we are given the ciphertext $E(s)$. Test $E(s)$ using the Z function. If s is not zero, create the encryption $E(1)$ via the identity $x^{p-1} \equiv 1 \bmod p$, which holds for all nonzero x. In particular, using $E(s)$ and the M function, we can use repeated squaring and compute $E(s)^{p-1} = E(1)$ in $\log p$ steps.

Clearly, from $E(1)$ and the A function we can construct $E(x)$ for every $x \in \mathbb{F}_p$. Then, given such an

$E(x)$, we can compute $f(x) = \left(\frac{x+s}{p}\right)$ in the following way. Add $E(s)$ and $E(x)$, yielding $E(x+s)$, and then compute the encrypted $(p-1)/2$th power[3] of $x+s$, giving $E(\left(\frac{x+s}{p}\right))$. Next, add $E(0)$, $E(-1)$ or $E(1)$ and test if it is an encryption of zero, and return 0, 1 or -1 accordingly. Applying this method on a superposition of $|x\rangle$ states, we can create (after reversibly uncomputing the garbage of the algorithm) the state $\frac{1}{\sqrt{p-1}}\sum_x f_s(x)|x\rangle$. We can then recover s by using Algorithm 4.1.

Corollary 4.1. *Given an efficient test to decide if a value is an encryption of zero, Algorithm 4.1 can be used to break any algebraically homomorphic encryption system.*

We can also break algebraically homomorphic cryptosystems using Shor's discrete log algorithm as follows. Suppose g is a generator for $\mathbb{F}_p^*$ and that we are given the unknown ciphertext $E(g^s)$. Create the superposition $\sum_{i,j}|i,j,E(g^{si+j})\rangle$ and then append the state $|\psi_{si+j}\rangle = \sum_t \left(\frac{g^{si+j}+t}{p}\right)|t\rangle$ to the superposition in i,j by the procedure described above. Next, uncompute the value $E(g^{si+j})$, which gives $\sum_{i,j}|i,j\rangle|\psi_{si+j}\rangle$. Rewriting this as $\sum_{i,r}|i,r-si\rangle|\psi_r\rangle$ and observing that the ψ_r are almost orthogonal, we see that we can apply the methods used in Shor's discrete log algorithm to recover s and thus g^s.

5 Shifted Multiplicative Characters of Finite Rings

In this section we show how to solve the shifted multiplicative character problem for $\mathbb{Z}/n\mathbb{Z}$ for any completely nontrivial multiplicative character of the ring $\mathbb{Z}/n\mathbb{Z}$ and extend this to the case when n is unknown. Unlike in the case for finite fields, the characters may be periodic. Thus the shift may not be unique. The Fourier transform is now the familiar Fourier transform over the additive group $\mathbb{Z}/n\mathbb{Z}$.

5.1 Shifted Multiplicative Characters of $\mathbb{Z}/n\mathbb{Z}$ for Known n

Definition 5.1. (Shifted Multiplicative Character Problem over $\mathbb{Z}/n\mathbb{Z}$) *Given χ, a completely nontrivial multiplicative character of $\mathbb{Z}/n\mathbb{Z}$, and a function f for which there is an s such that $f(x) = \chi(x+s)$ for all x. Find all t satisfying $f(x) = \chi(x+t)$ for all x.*

Multiplicative characters of $\mathbb{Z}/n\mathbb{Z}$ may be periodic, so to solve the shifted multiplicative character problem we first find the period and then we find the shift. If the period is ℓ then the possible shifts will be $\{s, s+\ell, s+2\ell, \dots\}$.

Algorithm 5.1. (Shifted Multiplicative Character Problem over $\mathbb{Z}/n\mathbb{Z}$)

1. Find the period ℓ of χ. Let χ' be χ restricted to $\{0,\dots,\ell-1\}$.
   (a) Create $\sum_{x=0}^{n-1}\chi(x+s)|x\rangle$.
   (b) Compute the Fourier transform over $\mathbb{Z}/n\mathbb{Z}$ to obtain $\sum_{y=0}^{\ell-1}\omega_\ell^{-sy}\hat{\chi}'(y)|yn/\ell\rangle$.
   (c) Measure $|yn/\ell\rangle$. Compute $n/\ell = \gcd(n, yn/\ell)$.
2. Find s using the period ℓ and χ':
   (a) Create $\sum_{x=0}^{\ell-1}\chi'(x+s)|x\rangle$.
   (b) Compute the Fourier transform over $\mathbb{Z}/\ell\mathbb{Z}$ to obtain $\sum_y \omega_\ell^{-sy}\hat{\chi}'(y)|y\rangle$.
   (c) For all y coprime to ℓ, $\hat{\chi}'(y)^{-1}$ into the phase to obtain $\sum_{y:\hat{\chi}'(y)\neq 0}\omega_\ell^{-sy}|y\rangle$.
   (d) Compute the inverse Fourier transform and measure.

Theorem 5.1. *Algorithm 5.1 solves the shifted multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$ for completely nontrivial multiplicative characters of $\mathbb{Z}/n\mathbb{Z}$ in polynomial time with probability at least $(\phi(n)/n)^3 = \Omega((\frac{1}{\log\log n})^3)$.*

Proof. Note: because χ is completely nontrivial, χ' is a primitive character of $\mathbb{Z}/\ell\mathbb{Z}$.

1. (a) $\chi(x+s)$ is nonzero exactly when $\gcd(x+s,n)=1$ so by Lemma 2.1 we can create the superposition with probability $\phi(n)/n$.
   (b) Since χ has period ℓ, the Fourier transform is nonzero only on multiples of n/ℓ.
   (c) Since $\hat{\chi}'(y) = \overline{\chi'(y)}\hat{\chi}'(1)$, and $\chi'(y)$ is nonzero precisely when $\gcd(y,n)=1$, when we measure yn/ℓ we have $n/\ell = \gcd(n, yn/\ell)$.
2. (a) Similar to the argument above, we can create the superposition with probability $\phi(\ell)/\ell$.
   (b) The Fourier transform moves the shift s into the phase.
   (c) As in the case for the finite field, this can be done by computing the phase of $\chi'(y)$ into the phase of $|y\rangle$.

[3] The Legendre symbol satisfies $\left(\frac{x}{p}\right) = x^{(p-1)/2}$.

(d) Let $A = \{y \in \mathbb{Z}/\ell\mathbb{Z} : \hat{\chi}'(y) \neq 0\}$. $A = (\mathbb{Z}/\ell\mathbb{Z})^*$ so $|A| = \phi(\ell)$. Then the amplitude of $|-s\rangle$ after the Fourier transform is $\frac{1}{\sqrt{\phi(\ell)}}\frac{1}{\sqrt{\ell}}\left(\sum_{y \in A} \omega_\ell^{-ys}\omega_\ell^{ys}\right) = \frac{1}{\sqrt{\phi(\ell)}}\frac{1}{\sqrt{\ell}}\left(\sum_{y \in A} 1\right) = \sqrt{\frac{\phi(\ell)}{\ell}}$. So the probability of measuring $|-s\rangle$ is $\phi(\ell)/\ell$.

Thus the algorithm succeeds with probability $(\phi(n)/n)(\phi(\ell)/\ell)^2 \geq (\phi(n)/n)^3$, which in turn is lower bounded by $\Omega((\frac{1}{\log\log n})^3)$.

5.2 Shifted Multiplicative Characters of $\mathbb{Z}/n\mathbb{Z}$ for Unknown n We now consider the case when n is unknown.

Definition 5.2. (Shifted Multiplicative Character Problem over $\mathbb{Z}/n\mathbb{Z}$ with Unknown n) *Given a completely nontrivial multiplicative character $\chi : \mathbb{Z}/n\mathbb{Z} \to \mathbb{C}$, for some unknown n, there is an s such that $f(x) = \chi(x+s)$ for all x. Find all t satisfying $f(x) = \chi(x+t)$ for all x.*

Theorem 5.2. *Given a lower bound on the size of the period of f, we can efficiently solve the shifted multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$ for unknown n on a quantum computer.*

Proof. Let ℓ be the period of f and χ' be χ restricted to $\mathbb{Z}/\ell\mathbb{Z}$. Using the Fourier sampling algorithm described in Section 2.3, we can approximately Fourier sample f over $\mathbb{Z}/\ell\mathbb{Z}$. Because $\chi'(y)$ is nonzero precisely when $\gcd(y, \ell) = 1$, this Fourier sampling algorithm returns y/ℓ with high probability, where y is coprime to ℓ. Thus we can find ℓ with high probability. Next, apply Algorithm 5.1 to find $s \bmod \ell$.

6 The Hidden Coset Problem

In this section we define the hidden coset problem and give an algorithm for solving the problem for Abelian groups under certain conditions. The algorithm consists of two parts, identifying the hidden subgroup and finding a coset representative. Finding a coset representative can be interpreted as solving a deconvolution problem.

The algorithms for hidden shift problems and hidden subgroup problems can be viewed as exploiting different facets of the power of the quantum Fourier transform. After computing a Fourier transform, the subgroup structure is captured in the magnitude whereas the shift structure is captured in the phase. In the hidden subgroup problem we measure after computing the Fourier transform and so discard information about shifts. Our algorithms for hidden shift problems do additional processing to take advantage of the information encoded in the phase. Thus the solution to the hidden coset problem requires fully utilizing the abilities of the Fourier transform.

Definition 6.1. (Hidden Coset Problem) *Given functions f and g defined on a group G such that for some $s \in G$, $f(x) = g(x+s)$ for all x in G, find the set of all t satisfying $f(x) = g(x+t)$ for all x in G. f is given as an oracle, and g is known but not necessarily efficiently computable.*

Lemma 6.1. *The answer to the hidden coset problem is a coset of some subgroup H of G, and g is constant on cosets of H.*

Proof. Let $S = \{t \in G : f(x) = g(x+t) \text{ for all } x \in G\}$ be the set of all solutions and let H be the largest subgroup of G such that g is constant on cosets of H. Clearly this is well defined (note H may be the trivial subgroup as in the Shifted Legendre Symbol Problem). Suppose t_1, t_2 are in S. Then we have $g(x+(-t_2+t_1)) = g((x-t_2)+t_1) = f(x-t_2) = g((x-t_2)+t_2) = g(x)$ for all x in G, so $-t_2+t_1$ is in H. This shows S is a contained in a coset of H. Since s is in S we must have that S is contained in $s+H$. Conversely, suppose $s+h$ is in $s+H$ (where h is in H). Then $g(x+s+h) = g(x+s) = f(x)$ for all x in G, hence $s+h$ is in S. It follows that $S = s+H$. While this proof was written with additive notation, it carries through if the group is nonabelian.

6.1 Identifying the Hidden Subgroup We start by finding the subgroup H. We give two different algorithms for determining H, the "standard" algorithm for the hidden subgroup problem, and the algorithm we used in Section 5.

In the standard algorithm for the hidden subgroup problem we form a superposition over all inputs, compute $g(x)$ into a register, measure the function value, compute the Fourier transform and then sample. The standard algorithm may fail when g is not distinct on different cosets of H. In such cases, we need other restrictions on g to be able to find the hidden subgroup H using the standard algorithm. Boneh and Lipton [8], Mosca and Ekert [30], and Hales and Hallgren [21] have all given criteria under which the standard hidden subgroup algorithm outputs H even when g is not distinct on different cosets of H.

In Section 5 we used a different algorithm to determine H because the function we were considering did not satisfy the conditions mentioned above. In this algorithm we compute the value of g into the *amplitude*, Fourier transform and then sample, whereas in the standard hidden subgroup algorithm we compute the value of g into a *register*. In general, this algorithm works when the fraction of values for which $\hat{g}$ is zero is sufficiently small and the nonzero values of $\hat{g}$ have constant magnitude.

6.2 Finding a Coset Representative as a Deconvolution Problem Once we have identified H, we can find a coset representative by solving the associated hidden coset problem for f' and g' where f' and g' are defined on the quotient group G/H and are consistent in the natural way with f and g. For notational convenience we assume that f and g are defined on G and that H is trivial, that is, the shift is uniquely defined.

The hidden shift problem may be interpreted as a *deconvolution* problem. In a deconvolution problem, we are given functions g and $f = g \star h$ (the convolution of g with some unknown function h) and asked to find this h. Let $\delta_y(x) = \delta(x-y)$ be the delta function centered at y. In the hidden shift problem, f is the convolution of δ_{-s} and g, that is, $f = g \star \delta_{-s}$. Finding s, or equivalently finding δ_{-s}, given f and g, is therefore a deconvolution problem.

Recall that under the Fourier transform convolution becomes pointwise multiplication. Thus, taking Fourier transforms, we have $\hat{f} = \hat{g} \cdot \hat{\delta}_{-s}$ and hence $\hat{\delta}_{-s} = \hat{g}^{-1} \cdot \hat{f}$ provided $\hat{g}$ is everywhere nonzero. For the multiplication by $\hat{g}^{-1}$ to be performed efficiently on a quantum computer would require $\hat{g}$ to have constant magnitude and be everywhere nonzero. However, even if only a fraction of the values of $\hat{g}$ are zero we can still approximate division of $\hat{g}$ by only dividing when $\hat{g}$ is nonzero and doing nothing otherwise. The zeros of $\hat{g}$ correspond to loss of information about δ_{-s}.

Algorithm 6.1.

1. Create $\sum_{x \in G} g(x+s)|x\rangle$.
2. Compute the Fourier transform to obtain $\sum_{y \in G} \overline{\psi_y(s)} \hat{g}(\psi_y)|y\rangle$, where ψ_y are the characters of the group G.
3. For all ψ_y for which $\hat{g}(\psi_y)$ is nonzero compute $\hat{g}(\psi_y)^{-1}$ into the phase to obtain $\sum_{y, \hat{g}(\psi_y) \neq 0} \overline{\psi_y(s)}|y\rangle$.
4. Compute the inverse Fourier transform and measure to obtain $-s$.

Theorem 6.1. *Suppose f and $\hat{g}$ are efficiently computable, the magnitude of $f(x)$ is constant for all values of x in G for which $f(x)$ is nonzero, and the magnitude of $\hat{g}(\psi_y)$ is constant for all values of ψ_y in $\hat{G}$ for which $\hat{g}(\psi_y)$ is nonzero. Let α be the fraction of x in G for which $f(x)$ is nonzero and β be the fraction of ψ_y in $\hat{G}$ for which $\hat{g}(\psi_y)$ is nonzero. Then the above algorithm outputs $-s$ with probability $\alpha\beta$.*

Proof. 1. By Lemma 2.1 we can create the superposition with probability α.

2. The Fourier transform moves the shift s into the phase.
3. Because $\hat{g}$ has constant magnitude, for values where $\hat{g}$ is nonzero, $\hat{g}(\psi_y)^{-1} = C\overline{\hat{g}(\psi_y)}$ for some constant C. So we can perform this step by computing the phase of $\overline{\hat{g}}$ into the phase. For the values where $\hat{g}$ is zero we can just leave the phase unchanged as those terms are not present in the superposition.
4. Let $A = \{y \in G : \hat{g}(\psi_y) \neq 0\}$. Then the amplitude of $|-s\rangle$ is

$$\frac{1}{\sqrt{|A|}} \frac{1}{\sqrt{|G|}} \left(\sum_{y \in A} \overline{\psi_y(s)}\, \overline{\psi_y(-s)} \right)$$
$$= \frac{1}{\sqrt{|A|}} \frac{1}{\sqrt{|G|}} \left(\sum_{y \in A} 1 \right) = \sqrt{\frac{|A|}{|G|}} = \sqrt{\beta},$$

so we measure $|-s\rangle$ with probability β.

Thus the algorithm succeeds in identifying s with probability $\alpha\beta$ and only requires one query of f and one query of $\hat{g}$.

6.3 Examples We show how the hidden shift problems we considered earlier fit into the framework of the hidden coset problem. In the shifted multiplicative character problem over finite fields, G is the additive group of $\mathbb{F}_q$, $g = \chi$ and H is trivial since the shift is unique for nontrivial χ. In the shifted multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$, G is the additive group of $\mathbb{Z}/n\mathbb{Z}$, $g = \chi$ and H is the subgroup $\{0, \ell, \ldots, n/\ell\}$, where ℓ (which is a factor of n) is the period of χ. In the shifted period multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$ for unknown n, G is the additive group of $\mathbb{Z}$, $g = \chi$ and H is the infinite subgroup $\ell\mathbb{Z}$.

7 Acknowledgements

We would like to thank the anonymous referee who pointed out the application of shifted Legendre symbol problem to algebraically homomorphic cryptosystems and Umesh Vazirani, whose many suggestions greatly improved this paper. We also thank Dylan Thurston and an anonymous referee for pointing out that algebraically homomorphic cryptosystems can be broken using Shor's algorithm for discrete log. Thanks to Lisa Hales for helpful last minute suggestions.

References

[1] Martin Abadi and Joan Feigenbaum. Secure circuit evaluation. A protocol based on hiding information from an oracle. *Journal of Cryptology*, 2(1):1–12, 1990.

[2] Eric Bach and Jeffrey Shallit. *Algorithmic Number Theory | E–cient Algorithms*, volume I. MIT Press, Cambridge, USA, 1996.

[3] Robert Beals. Quantum computation of Fourier transforms over symmetric groups. *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 48–53, 1997.

[4] J. Niel de Beaudrap, Richard Cleve, and John Watrous. Sharp quantum vs. classical query complexity separations. quant-ph archive no. 0011065, 2001. Journal version to appear in Algorithmica.

[5] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

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

[7] Manuel Blum and Silvio Micali, How to generate cryptographically strong sequences of pseudo-random bits, *SIAM Journal on Computing*, 13(4):851–863, 1984.

[8] Dan Boneh and Richard J. Lipton. Quantum cryptanalysis of hidden linear functions. *Lecture Notes in Computer Science*, 963:424–437, 1995.

[9] Dan Boneh and Richard J. Lipton. Algorithms for black-box fields and their application to cryptography. *Lecture Notes in Computer Science*, 1109:283–297, 1996.

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

[11] Richard Cleve and John Watrous. Fast parallel circuits for the quantum Fourier transform. *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 526–536, 2000.

[12] Wim van Dam. Quantum algorithms for weighing matrices and quadratic residues. quant-ph archive no. 0008059, 2000. Journal version to appear in Algorithmica.

[13] Wim van Dam and Sean Hallgren. Efficient Quantum Algorithms for Shifted Quadratic Character Problems. quant-ph archive no. 0011067, 2000.

[14] Wim van Dam and Gadiel Seroussi. Efficient Quantum Algorithms for Estimating Gauss Sums. quant-ph archive no. 0207131, 2002.

[15] Ivan B. Damgård. On the randomness of Legendre and Jacobi sequences, *Lecture Notes in Computer Science*, 403:163–172, 1988.

[16] Mark Ettinger and Peter Høyer. On Quantum Algorithms for Noncommutative Hidden Subgroups. quant-ph archive no. 9807029, 1998.

[17] Mark Ettinger, Peter Høyer and Emanuel Knill. Hidden subgroup states are almost orthogonal. quant-ph archive no. 9901034, 1999.

[18] Katalin Friedl, Frédéric Magniez, Miklos Santha and Pranab Sen. Quantum testers for hidden group properties quant-ph archive no. 0208184, 2002.

[19] Michaelangelo Grigni, Leonard Schulman, Monica Vazirani and Umesh Vazirani. Quantum mechanical algorithms for the nonabelian hidden subgroup problem. *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pages 68–74, 2001.

[20] Lisa Hales. The quantum Fourier transform and extensions of the abelian hidden subgroup problem. Ph.D. Thesis, U.C. Berkeley, 2002.

[21] Lisa Hales and Sean Hallgren. An improved quantum Fourier transform and appliations. *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 515–525, 2000.

[22] Sean Hallgren. Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem. *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, pages 653–658, 2001.

[23] Sean Hallgren, Alexander Russell, and Amnon Ta-Shma. Normal subgroup reconstruction and quantum computation using group representations. *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 627–635, 2000.

[24] Lawrence Ip. Solving Shift Problems and the Hidden Coset Problem Using the Fourier Transform. quant-ph archive no. 0205034, 2002.

[25] Gábor Ivanyos, Frédéric Magniez, and Miklos Santha. Efficient quantum algorithms for some instances of the non-abelian hidden subgroup problem. *Proceedings of the Thirteenth Annual ACM Symposium on Parallel Algorithms and Architectures*, pages 263–270, 2001.

[26] Alexei Yu. Kitaev. Quantum measurements and the Abelian stabilizer problem. quant-ph archive no. 9511026, 1995.

[27] Alexei Yu. Kitaev. Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191–1249, 1997.

[28] Rudolf Lidl and Harald Niederreiter. *Finite Fields*, volume 20 of *Encyclopedia of Mathematics and Its Applications*. Cambridge, second edition, 1997.

[29] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.

[30] Michele Mosca and Artur Ekert. The hidden subgroup problem and eigenvalue estimation on a quantum computer. *Proceedings of the 1st NASA International Conference on Quantum Computing and Quantum Communication*, Lecture Notes in Computer Science 1509, pages 174–188, 1999.

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

[32] Daniel R. Simon. On the power of quantum computa-

tion. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[33] Richard Tolimieri, Myoung An, and Chao Lu. *Algorithms for Discrete Fourier Transform and Convolution*. Springer-Verlag, 1989.

[34] John Watrous. Quantum algorithms for solvable groups. *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pages 60–67, 2001.