Solving Shift Problems and the Hidden Coset Problem Using the Fourier Transform

Lawrence Ip*

May 7, 2002

Abstract

We give a quantum algorithm for solving a shifted multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$ and finite fields. We show that the algorithm can be interpreted as a matrix factorization or as solving a deconvolution problem and give sufficient conditions for a shift problem to be solved efficiently by our algorithm. We also show that combining the shift problem with the hidden subgroup problem results in a hidden coset problem. This naturally captures the redundancy in the shift due to the periodic structure of multiplicative characters over $\mathbb{Z}/n\mathbb{Z}$.

1 Introduction

The Fourier transform lies at the heart of the solution of all problems that are known to admit a large quantum speedup over classical algorithms. This is due to two factors, the quantum Fourier transform can be performed exponentially faster than its classical counterpart, and the Fourier transform is particularly suited to extracting information about periodicities.

Taking a periodic superposition, performing a Fourier transform and measuring gives us the period. More generally, the Fourier transform of a superposition over coset states of a subgroup yields a superposition over the dual or "perp" of the subgroup. This is the basis of the hidden subgroup problem. The most well known example of a problem and algorithm fitting this framework is Shor's solution to the factoring problem [6].

*Computer Science Division, University of California, Berkeley, USA. Email: lip@eecs.berkeley.edu. 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.

However, information about periodicities is not the only feature that Fourier transforms can identify. The Fourier transform can also identify shifts or translations. This suggests that we look for ways in which the Fourier transform can be used to identify unknown shifts.

The shift problem may be formulated as follows.

Let g be a complex valued function defined on the group G. Let f be a shifted version of g where $f(x) = g(x + s)$ for some s in G. Find s.

We give an efficient quantum algorithm for solving the shift problem provided G is abelian and g satisfies certain conditions. The conditions are satisfied if G is the additive group of a finite field and g is a multiplicative character. So our algorithm solves the shifted multiplicative character problem on a finite field.

If the shift is not uniquely determined, it can be shown that the set of possible shifts is a coset of a subgroup of the group G. We then have the *hidden coset problem*.

Let g be a complex valued function defined on the group G. Let f be a shifted version of g where $f(x) = g(x + s)$ for some s in G. Find all s' satisfying $f(x) = g(x + s')$ for all x.

The hidden coset problem combines the features of the hidden subgroup problem and the shift problem. Our algorithm treats this problem by first solving a hidden subgroup problem to find H and then solving a shift problem on the quotient group G/H. The shifted multiplicative character problem over $\mathbb{Z}/n\mathbb{Z}$ fits this framework. The multiplicative characters of $\mathbb{Z}/n\mathbb{Z}$ are periodic with respect to the additive group and so the shift is not uniquely determined.

Our algorithm solves the shifted Legendre symbol problem and the shifted quadratic character problem considered by van Dam and Hallgren [8]. The quadratic character is an example of a multiplicative character and the Legendre symbol is the quadratic character for $\mathbb{Z}/p\mathbb{Z}$. van Dam and Hallgren also give an algorithm for the shifted Jacobi symbol problem for $\mathbb{Z}/n\mathbb{Z}$ where n is square free. The Jacobi symbol is also an example of a multiplicative character. Our treatment solves the problem for all multiplicative characters of $\mathbb{Z}/n\mathbb{Z}$ for all odd n.

2 The Shift Problem

In this section we state the shift problem, and show how it can be formulated in terms of matrix multiplication. By factoring the relevant matrix we obtain an algorithm for solving the shift problem and also sufficient conditions for the algorithm to be implemented efficiently on a quantum computer. The analysis involves Fourier transforms over finite abelian groups. For a review of characters and Fourier transforms over finite abelian groups see Appendix A.

The shift problem is the following. Let g be a complex valued function defined on the group G. Let f be a shifted version of g where $f(x) = g(x + s)$ for some s in G. Find s.

We begin by reformulating the shift problem in the following way. Let s be the unknown shift and X be the matrix with columns and rows indexed by the elements of G and the matrix element in row x and column y given by $g(x+y)$. The precise ordering of the elements

of G is irrelevant as long as we are consistent. That is

$$X = [g(x+y)]_{x,y\in G}\,.$$

We then compute

$$\begin{aligned} X|s\rangle &= \sum_{x\in G} g(x+s)|x\rangle \\ &= \sum_{x\in G} f(x)|x\rangle. \end{aligned}$$

Solving the shift problem then reduces to inverting X.

We show below that the structure of X allows us to factor X into $X = (F^T)^{-1}DF^{-1}$, where F is the Fourier transform matrix and D is a diagonal matrix. The diagonal entries of D (up to a scale factor) turn out to be the Fourier transform of g evaluated at the characters of G. Provided g satisfies certain conditions, we can efficiently compute X^{-1} by inverting F^T, D and F in turn to obtain $X^{-1} = FD^{-1}F^T$.

2.1 Matrix Factorization

We now show that X is "diagonalized"[1] by the Fourier transform matrix. Let F be the Fourier transform matrix defined by

$$F = [\psi_y(x)]_{x,y\in G}\,,$$

where x and y index the rows and columns respectively and ψ_y is a character of G indexed by y. Thus each column of F contains a character of G evaluated at all elements of G.

Computing F^TXF we find that

$$\begin{aligned} F^TXF &= [\psi_v(x)]_{v,x}\,[g(x+y)]_{x,y}\,[\psi_w(y)]_{y,w} \\ &= [\psi_v(x)]_{v,x}\left[\sum_{y\in G} g(x+y)\psi_w(y)\right]_{x,w} \\ &= [\psi_v(x)]_{v,x}\left[\sum_{y\in G} g(y)\psi_w(y-x)\right]_{x,w} \\ &= [\psi_v(x)]_{v,x}\left[\overline{\psi_w(x)}\sum_{y\in G} g(y)\psi_w(y)\right]_{x,w} \\ &= [\psi_v(x)]_{v,x}\left[\overline{\psi_w(x)}\hat{g}(\psi_w)\right]_{x,w} \end{aligned}$$

[1] Strictly speaking we are not diagonalizing X as that would mean writing $X = FDF^\dagger$ and not $X = FDF^T$.

$$= \left[\sum_{x \in G} \psi_v(x) \overline{\psi_w(x)} \hat{g}(\psi_w) \right]_{v,w}$$
$$= \left[|G| \hat{g}(\psi_w) \delta_{vw} \right]_{v,w},$$

which is a diagonal matrix. Let D denote this diagonal matrix and we have $F^T X F = D$.

2.2 Matrix Inversion

The algorithm then consists of computing $X^{-1} = F D^{-1} F^T$. If X is not full rank, then D contains some zeros along the diagonal. We then calculate the *pseudoinverse* of X (sometimes called the *Moore-Penrose generalized inverse*). The pseudoinverse of X, X^* satisfies

- $XX^*X = X$,
- $X^*XX^* = X$,
- XX^* and X^*X are Hermitian.

The pseudoinverse of a matrix always exists and is unique [3]. The pseudoinverse of a diagonal matrix is given by inverting the nonzero elements of the diagonal. We then have $X^* = F D^* F^T$.

The fraction of zeros along the diagonal of D gives the probability of error, the probability that the algorithm fails to output s.

2.3 Sufficient Conditions

To implement the algorithm efficiently we need some conditions on g.

1. The magnitude of $g(x)$ is constant for all x such that $g(x)$ is nonzero.
2. The Fourier transform of g, $\hat{g}(\psi_y)$, has constant magnitude for all y for which $\hat{g}(\psi_y)$ is nonzero.
3. $\hat{g}$ can be computed efficiently up to a multiplicative constant (on a classical computer).

The above conditions are sufficient for efficient implementation of the algorithm. The following two parameters determine the probability of the algorithm outputting the correct answer.

1. α, the fraction of x in G for which $g(x)$ is nonzero
2. β, the fraction of y in G for which the Fourier transform of g, $\hat{g}(\psi_y)$, is nonzero.

The probability of our algorithm succeeding is $\alpha\beta$.

Condition 1 is needed to be able to create the superposition with $|x\rangle$ having amplitude $g(x)$ up to a global constant,

$$\sum_{x \in G} g(x)|x\rangle.$$

This can be done efficiently with probability α.

Conditions 2 and 3 are needed because we will need to compute $\overline{\hat{g}(\psi_y)}$ to invert D.

The parameter β is describes the rank of D and thus of X. When we apply X followed by its pseudoinverse, we get a vector close enough to $|s\rangle$ so that when we measure we obtain s with probability β.

Thus the overall probability of success of the algorithm is $\alpha\beta$.

2.4 Implementation of the Algorithm

We now show how to implement the algorithm efficiently on a quantum computer.

1. Setup a superposition of the all the values of f with the amplitude of $|x\rangle$ equal to $f(x)$ to obtain

$$C \sum_{x \in G} g(x+s)|x\rangle.$$

2. Compute the Fourier transform to obtain

$$C' \sum_{y \in G} \overline{\psi_y(s)} \hat{g}(\psi_y)|y\rangle.$$

3. Now compute $\overline{\hat{g}(\psi_y)}$ into the phase to obtain

$$C'' \sum_{\substack{y \in G \\ \hat{g}(\psi_y) \neq 0}} \overline{\psi_y(s)}|y\rangle$$

4. Computing the inverse Fourier transform and measuring gives $-s$.

In step 1 we setup a superposition over the elements of G,

$$\sum_{x \in G} |x\rangle,$$

compute $f(x)$ and measure to see whether $f(x)$ is zero. If so, then the algorithm fails. If not, we are left with a superposition over all x such that $f(x) \neq 0$. The algorithm succeeds

here with probability α. We next compute $f(x)$ into the amplitude of $|x\rangle$ (up to a constant factor) . Condition 1 ensures that we can do this by computing the phase of $f(x)$ into the phase of $|x\rangle$. We can always approximate this arbitrarily closely by approximating the phase of $f(x)$ to the nearest 2^nth root of unity for some sufficiently large n.

Step 2 follows from observing that

$$\begin{aligned} C' \sum_{y \in G} \left(\sum_{x \in G} g(x+s)\psi_y(x) \right) |y\rangle &= C' \sum_{y \in G} \left(\sum_{x \in G} g(x)\psi_y(x-s) \right) |y\rangle \\ &= C' \sum_{y \in G} \overline{\psi_y(s)} \left(\sum_{x \in G} g(x)\psi_y(x) \right) |y\rangle \\ &= C' \sum_{y \in G} \overline{\psi_y(s)} \hat{g}(\psi_y) |y\rangle, \end{aligned}$$

where C' is a constant.

Step 3 can be performed because of Conditions 2 and 3.

In Step 4 we measure and obtain $-s$ with probability of β. The reason we get $-s$ instead of s is that computing the inverse Fourier transform corresponds to multiplying by $F^\dagger$ instead of F^T.

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

3 The Hidden Coset Problem

If g has a "subgroup" structure then the shift may not be unique. We can make this precise in the following way by formulating a *hidden coset problem* that combines the features of the shift problem with that of the hidden subgroup problem. The hidden coset problem is a shift problem where the shift may not be uniquely defined.

The hidden coset problem is the following. Let g be a complex valued function defined on the group G. Let f be a shifted version of g where $f(x) = g(x+s)$ for some s in G. Find all s' satisfying $f(x) = g(x+s')$ for all x.

Let H be the largest subgroup of G such that g is constant on cosets of H. Because of the structure of g, s is determined only "modulo" H. Thus to solve the hidden coset problem, we need to first identify the hidden subgroup H and then the shift s modulo H.

Assuming that we have already found H, to find s modulo H, define g' and f' as complex valued functions on the quotient group G/H in the natural way so that $g'(x+H) = g(x)$ and $f'(x+H) = f(x)$ for all x in G. Then if g' satisfies the conditions in Section 2.3 when considered over the group G/H we can apply the algorithm for the shift problem to find s modulo H.

We now show how to find H. The standard formulation of the hidden subgroup problem assumes that g is constant on cosets of H and that g takes on *distinct values on distinct cosets*. This can be solved using the "standard" algorithm

1. Prepare a superposition over all of G.
2. Computing g into a register.
3. Fourier sampling to obtain a random element of $H^{\perp}$.

The condition that g takes on distinct values on distinct cosets of H means that we sample uniformly over the elements of $H^{\perp}$. This condition can be relaxed slightly so that the standard algorithm still works. Boneh and Lipton [1] and Mosca and Ekert [5] give the condition that g' is at most m to 1 and m is less than the smallest prime factor of $|H|$, the cardinality of H. Hales and Hallgren [2] give the condition that at least a polylogarithmic number of values of g need to be changed to reduce the period of g.

However, as we already have restrictions on g' and $\hat{g}'$ (so that we can solve the shift problem) we can give another condition for the hidden subgroup problem to be solved efficiently. If β, the fraction of values of $\hat{g}'$ that is nonzero satisfies $\beta > 1/p + \operatorname{poly}\log(|H|)$, where p is the smallest prime factor of $|H|$, then the following algorithm will find H.

1. Prepare a superposition over all of G.
2. Computing g into the *phase*.
3. Fourier sampling to obtain a random element of $H^{\perp}$.

The difference from the standard algorithm for solving the hidden subgroup problem is that we compute g into the *phase*.

4 Shifted Character Problem (Finite Field)

The shifted character problem for the finite field $\mathbb{F}_q$ is as follows.

Given a finite field $\mathbb{F}_q$ (where $q = p^m$ for some prime p), a multiplicative character χ of $\mathbb{F}_q$ and a shifted version of χ, $f(x) = \chi(x + s)$. Find s.

For a review of additive and multiplicative characters in finite fields see Appendix A. Appendix B contains a discussion of Fourier transforms over the additive group of multiplicative characters.

The shifted multiplicative character problem over a finite field fits into our general framework. The group G is the additive group of the finite field. $g = \chi$ is a (non-trivial) multiplicative character of $\mathbb{F}_q$. We show that χ satisfies the sufficient conditions of Section 2.3.

1. For all nonzero x, $\chi(x)$ has unit magnitude.

2. $\hat{\chi}(\psi_0) = 0$ and as shown in Appendix B, $\hat{\chi}(\psi_{g^m}) = \overline{\chi(g^m)}\hat{\chi}(\psi_{g^0})$ and so has magnitude $|\hat{\chi}(\psi_{g^0})|$ which is constant.

3. χ and thus $\overline{\chi(g^m)}$ can be computed efficiently. So $\hat{\chi}$ can be computed efficiently (up to a constant phase).

We next calculate the probability of our algorithm succeeding.

1. $\chi(x)$ is zero only if $x = 0$ so $\alpha = 1 - 1/q$.

2. $\hat{\chi}(\psi_y)$ is zero if for $y = 0$ so $\beta = 1 - 1/q$.

So our algorithm succeeds with probability $\alpha\beta = (1 - 1/q)^2$.

5 Shifted Character Problem for $\mathbb{Z}/n\mathbb{Z}$

The shifted character problem for $\mathbb{Z}/n\mathbb{Z}$ is as follows.

Given a ring $\mathbb{Z}/n\mathbb{Z}$ with n odd, a multiplicative character χ of $\mathbb{Z}/n\mathbb{Z}$ and a shifted version of χ, $f(x) = \chi(x + s)$. Find s.

The shifted character problem over $\mathbb{Z}/n\mathbb{Z}$ has the interesting feature that the solution to the shift is not unique. This is because multiplicative characters in $\mathbb{Z}/n\mathbb{Z}$ have periodicities with respect to the additive group. See Appendix A for a discussion of periodicities of multiplicative characters. Appendix B contains a discussion of Fourier transforms of multiplicative characters.

The shifted character problem over $\mathbb{Z}/n\mathbb{Z}$ fits into the hidden coset problem framework described in Section 3 so we can apply our algorithm. If $n = p_1^{m_1} \dots p_k^{m_k}$ we have $\alpha = \beta = (1 - 1/p_1) \dots (1 - 1/p_k)$ and so our algorithm will succeed with probability $(1 - 1/p_1)^2 \dots (1 - 1/p_k)^2$ (after solving the associated hidden subgroup problem).

6 Interpretation as Deconvolution

Our algorithm for solving the shift problem can be thought of solving a deconvolution problem. To see this, let $\delta_y(x) = \delta(x - y)$ be the delta function centered at y. Then f is the convolution of δ_{-s} and g, that is

$$f = \delta_{-s} \star g.$$

So to recover s or equivalently δ_{-s}, we need to solve a deconvolution problem.

Taking Fourier transforms and observing that in the Fourier domain convolution becomes pointwise multiplication we see that

$$\hat{f} = \hat{\delta}_{-s} \cdot \hat{g},$$

where $\hat{f}$, $\hat{g}$, $\hat{\delta}_{-s}$ are the Fourier transforms of f, g and δ_{-s} respectively. We then have

$$\delta_{-s} = \mathcal{F}^{-1}\left(\hat{f}/\hat{g}\right)$$

where the division is pointwise.

For the division to be performed on efficiently on a quantum computer would require that the magnitude of $\hat{g}$ be constant and non-zero. However even if 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 non-zero and doing nothing otherwise.

Deconvolution is a well studied classical problem and perhaps this interpretation will enable us to leverage existing deconvolution techniques to broaden the class of problems amenable to our approach.

7 Conclusion

We have presented a general framework for a class of shift problems and a set of sufficient conditions for the problems to be efficiently solved on a quantum computer. However, the sufficient conditions are fairly restrictive although they include the shifted multiplicative character problem over finite fields and rings $\mathbb{Z}/n\mathbb{Z}$.

It would be of interest to investigate what other shift problems satisfy the sufficient conditions and whether a less restrictive set of sufficient conditions exists.

8 Acknowledgements

I would like to thank Umesh Vazirani for much appreciated advice and encouragement and Sean Hallgren and Wim van Dam for useful discussions.

9 Bibliography

References

[1] Dan Boneh and Richard J. Lipton. Quantum cryptanalysis of hidden linear functions (extended abstract). In *Advances in Cryptology — CRYPTO '95*, Lecture Notes in Computer Science 963, pages 424–437, 1995.

[2] Lisa Hales and Sean Hallgren. An improved quantum fourier transform and appliations. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, 2001.

[3] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge, 1985.

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

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

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

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

[8] Wim van Dam and Sean Hallgren. Efficient quantum algorithms for shifted quadratic character problems. *quant-ph/0011067*, 2000.

A Mathematical Background

We start with some definitions and background. For more details see the book by Lidl and Niederreiter [4], and the book by Tolimieri et al. [7].

A.1 Characters of a Group

A character χ of a finite abelian group G is a homomorphism from a group G to $(\mathbb{C}, \times)$, the group of complex numbers with multiplication. That is

$$\chi(g_1 g_2) = \chi(g_1)\chi(g_2)$$

for all $g_1, g_2 \in G$. If G is cyclic with generator g, the characters are

$$\chi_k(g^l) = \exp(2\pi i k l/|G|)$$

for $k = 0, \ldots, |G| - 1$.

The characters of G form a group $\hat{G}$, known as the dual group, with multiplication defined as

$$(\chi_1\chi_2)(g) = \chi_1(g)\chi_2(g)$$

for all $\chi_1, \chi_2 \in \hat{G}$, $g \in G$. $\hat{G}$ is isomorphic to G. In particular, $|\hat{G}| = |G|$, that is the number of characters of G is the same as the cardinality of G.

A.2 Fourier Transform over a Group

Given a complex valued function f over G, the Fourier transform of f over the group G is given by

$$\hat{f}(\chi) = \sum_{g \in G} f(g)\chi(g)$$

for all $\chi \in \hat{G}$.

A.3 Characters of a Field

In a field $\mathbb{F}_q$ (where $q = p^m$ for some prime p) we have two operations, multiplication and addition, with corresponding groups. Thus we can define two different groups of characters.

Characters of the multiplicative group $\mathbb{F}_q^*$ of $\mathbb{F}_q$ are called *multiplicative characters* of $\mathbb{F}_q$. Since $\mathbb{F}_q^*$ is cyclic with order $q-1$, its characters $\chi_0, \ldots \chi_{q-2}$ can be explicitly represented as

$$\chi_k(g^l) = \exp\left(\frac{2\pi i k l}{q-1}\right)$$

for all $l = 0, 1, \ldots, q-2$, where g is a generator of $\mathbb{F}_q^*$. It is often convenient to extend the definition of χ_k to include 0 by defining $\chi_k(0) = 0$. The quadratic character referred to in van Dam and Hallgren [8] is $\chi_{\frac{q-1}{2}}$.

Characters of the additive group of $\mathbb{F}_q$ are called *additive characters* of $\mathbb{F}_q$. The additive characters ψ_a (for all $a \in \mathbb{F}_q$) have the form

$$\psi_a(c) = \exp\left(\frac{2\pi i \mathrm{Tr}(ac)}{p}\right)$$

for all $c \in \mathbb{F}_q$, where

$$\mathrm{Tr}(x) = \sum_{k=0}^{m-1} x^{p^k}$$

is the trace function from $\mathbb{F}_q$ to $\mathbb{F}_p$ and $\mathbb{F}_p$ is identified with $\mathbb{Z}/p\mathbb{Z}$ for the purposes of evaluating the exponential. ψ_1 is the canonical additive character of $\mathbb{F}_q$. We have that $\psi_a(c) = \psi_1(ac)$.

A.4 Characters of $\mathbb{Z}/n\mathbb{Z}$

Similarly we can define additive and multiplicative characters in $\mathbb{Z}/p^m\mathbb{Z}$ where p is an odd[2] prime. $\mathbb{Z}/p^m\mathbb{Z}$ with addition is a cyclic group so we can define additive characters

[2]We only consider odd p because $(\mathbb{Z}/2^m\mathbb{Z})^*$ is not cyclic for $m \geq 3$. In fact, $(\mathbb{Z}/2^m\mathbb{Z})^*$ is the product of two cyclic groups [7].

$\psi_0, \ldots, \psi_{p^m-1}$

$$\psi_k(x) = \exp\left(\frac{2\pi ikx}{p^m}\right)$$

for all $x \in \mathbb{Z}/p^m\mathbb{Z}$.

$(\mathbb{Z}/p^m\mathbb{Z})^*$ is a cyclic group, so we can define multiplicative characters $\chi_0, \ldots, \chi_{(p-1)p^{m-1}-1}$, where

$$\chi_k(g^l) = \exp\left(\frac{2\pi ikl}{(p-1)p^{m-1}}\right)$$

for all $l = 0, 1, \ldots, (p-1)p^{m-1} - 1$ and g is a generator of $(\mathbb{Z}/p^m\mathbb{Z})^*$. We can extend the definition of χ_k to include all of $\mathbb{Z}/p^m\mathbb{Z}$ by defining $\chi_k(x) = 0$ if x is a multiple of p.

$\mathbb{Z}/n\mathbb{Z}$ with addition is a cyclic group so we can define additive characters $\psi_0, \ldots, \psi_{n-1}$

$$\psi_k(x) = \exp\left(\frac{2\pi ikx}{n}\right)$$

for all $x \in \mathbb{Z}/n\mathbb{Z}$.

If n is odd, we can define multiplicative characters over $\mathbb{Z}/n\mathbb{Z}$ by observing that if $n = p_1^{m_1} \ldots p_k^{m_k}$ then

$$\mathbb{Z}/n\mathbb{Z} \cong \mathbb{Z}/p_1^{m_1}\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/p_k^{m_k}\mathbb{Z}.$$

Let ϕ be an isomorphism from $\mathbb{Z}/n\mathbb{Z}$ to $\mathbb{Z}/p_1^{m_1}\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/p_k^{m_k}\mathbb{Z}$ with $\phi(x) = (\phi_1(x), \ldots, \phi_k(x))$ given by $\phi_j(x) = x \bmod p_j^{m^j}$. Define the multiplicative characters as the product of the multiplicative characters in the corresponding $\mathbb{Z}/p_j^{m_j}\mathbb{Z}$. That is

$$\chi_{l_1,\ldots,l_k}(x) = \prod_{r=1}^{k} \chi_{l_r}^{(r)}(\phi_r(x)),$$

where $\chi_{l_r}^{(r)}$ is a multiplicative character of $\mathbb{Z}/p_r^{m_r}\mathbb{Z}$. Noting that all the $\chi_{l_1,\ldots,l_k}$ are distinct for appropriate ranges of $l_1, \ldots, l_k$ and that there are exactly the right number of them we see that we have all the multiplicative characters.

A.5 Periodicities of Multiplicative Characters of $\mathbb{Z}/p^m\mathbb{Z}$

The multiplicative characters of $\mathbb{Z}/p^m\mathbb{Z}$ have a particular *additive* periodic structure. Every multiplicative character has a period that is a power of p. That is, for every k, there is a j such that

$$\chi_k(x + p^j) = \chi_k(x)$$

for all $x \in \mathbb{Z}/p^m\mathbb{Z}$.

In fact we have the following theorem [7] that explicitly gives the period.

Theorem 1 *Let g be a generator of the group of units $(\mathbb{Z}/p^m\mathbb{Z})^*$ and let χ_k be a multiplicative character of $\mathbb{Z}/p^m\mathbb{Z}$ defined*

$$\chi_k(x) = \begin{cases} 0 & \text{if } p|x, \\ \exp\left(\frac{2\pi i k l}{(p-1)p^{m-1}}\right) & \text{if } x = g^l. \end{cases}$$

Let j be such that $\gcd(p^m, k) = p^{m-j}$. Then p^j is the additive period of χ_k, the smallest T such that $\chi_k(x+T) = \chi_k(x)$ for all $x \in \mathbb{Z}/p^m\mathbb{Z}$.

Proof: Suppose $\gcd(p^m, k) = p^{m-j}$ and that the period of χ_k is T. We first show that $T|p^j$ and then that $T = p^j$.

If $p|x$ then $p|(x+p^j)$ and so $\chi_k(x+p^j) = 0 = \chi_k(x)$. If $x = g^l$ then because $p^{m-j}|k$, the value of $\chi_k(g^l)$ is determined by the value of $l \bmod (p-1)p^{j-1}$, which is in turn determined by the value of $g^l \bmod p^j$. But $g^l + p^j \equiv g^l \mod p^j$ and so $\chi_k(x+p^j) = \chi_k(x)$. This shows that $T|p^j$.

Since $T|p^j$, if $T \neq p^j$, we must have $T = p^{j'}$ for some $j' < j$. Then $\chi_k(1) = \chi_k(1 + rp^{j'})$ for all $r = 0, \ldots, p^{m-j'} - 1$. Now

$$\{1 + rp^{j'} : r = 0, \ldots, p^{m-j'} - 1\} = \{g^{(p-1)p^{j'-1}l'} : l' = 0, \ldots, p^{m-j'} - 1\}$$

since all the $g^{(p-1)p^{j'-1}l'}$ are distinct and $g^{(p-1)p^{j'-1}l'} \equiv 1 \mod p^{j'}$. But $\chi_k(g^{(p-1)p^{j'-1}l'}) = 1$ for all l' only if $k(p-1)p^{j'-1}$ is a multiple of $(p-1)p^{m-1}$ which implies that $p^{m-j'}|k$. This is a contradiction since p^{m-j} was the largest power of p dividing k. Thus $T = p^j$. □

A.6 Periodicities of Multiplicative Characters of $\mathbb{Z}/n\mathbb{Z}$

As we saw in Appendix A.4, a multiplicative character $\chi_{l_1,\ldots,l_k}$ of $\mathbb{Z}/n\mathbb{Z}$ can be decomposed into a product of multiplicative characters $\chi^{(1)}_{l_1} \ldots \chi^{(k)}_{l_k}$ of $\mathbb{Z}/p_1^{m_1}\mathbb{Z}, \ldots, \mathbb{Z}/p_k^{m_k}\mathbb{Z}$. The period of $\chi_{l_1,\ldots,l_k}$ is then the product of the periods of $\chi^{(1)}_{l_1} \ldots \chi^{(k)}_{l_k}$.

B Fourier Transforms of Multiplicative Characters

B.1 Finite Field Case

Consider the natural representation of $\mathbb{F}_q^*$, the multiplicative group. It is well known that this is cyclic. If g is a generator then the elements can be represented as $\{1, g, g^2, g^3, \cdots, g^{q-2}\}$. The multiplicative characters are then exponentials in k

$$\chi_l(g^k) = \exp\left(\frac{2\pi i k l}{q-1}\right).$$

The non-trivial additive characters can be written

$$\psi_{g^m}(g^k) = \exp\left(\frac{2\pi i \mathrm{Tr}(g^m g^k)}{p}\right).$$

Note that these are all translates of the canonical additive character

$$\psi_{g^0}(g^k) = \exp\left(\frac{2\pi i \mathrm{Tr}(g^k)}{p}\right).$$

When we take the Fourier transform of a multiplicative character over the additive group we are expressing an exponential in terms of a basis where the basis functions are translates of the canonical additive character. To compute the change of basis we compute the inner product of the exponential with all the translates of the canonical additive character. This is the same as the inner product of the canonical additive characters with the exponential translated in the opposite direction. But a translated exponential is just the original exponential with a phase shift that is given by the exponential of the size of the translation. Thus

$$\begin{aligned}
\hat{\chi}_l(\psi_{g^m}) &= \sum_{k=0}^{q-2} \psi_{g^m}(g^k)\chi_l(g^k) \\
&= \sum_{k=0}^{q-2} \exp\left(\frac{2\pi i \mathrm{Tr}(g^m g^k)}{p}\right) \exp\left(\frac{2\pi i k l}{q-1}\right) \\
&= \sum_{k=0}^{q-2} \exp\left(\frac{2\pi i (k-m) l}{q-1}\right) \exp\left(\frac{2\pi i \mathrm{Tr}(g^k)}{p}\right) \\
&= \exp\left(-\frac{2\pi i m l}{q-1}\right) \hat{\chi}_l(\psi_{g^0}) \\
&= \overline{\chi_l(g^m)} \hat{\chi}_l(\psi_{g^0}).
\end{aligned}$$

B.2 $\mathbb{Z}/p^m\mathbb{Z}$ Case

If a multiplicative character χ_l of $\mathbb{Z}/p^m\mathbb{Z}$ has no periodicity, then $p \nmid l$. An extension of the argument used for the finite field case shows that

$$\hat{\chi}_l(\psi_y) = \overline{\chi_l(y)} \hat{\chi}_l(\psi_1).$$

See Tolimieri et al. for details [7]. If $\gcd(p^m, l) = p^{m-j}$, χ_l has period p^l and the previous argument does not work because p^m, the size of the additive group, and l are not relatively prime. However, by projecting $\mathbb{Z}/p^m\mathbb{Z}$ onto $\mathbb{Z}/p^j\mathbb{Z}$ by sending x to $x \bmod p^j$ we transform

χ_l to a multiplicative character of $\mathbb{Z}/p^j\mathbb{Z}$ with no periodicity. Thus the Fourier transform of χ_l over $\mathbb{Z}/p^m\mathbb{Z}$ must be

$$\hat{\chi}_l(\psi_y) = \begin{cases} K\overline{\chi_l(y/p^{m-j})} & \text{if } p^{m-j}|y, \\ 0 & \text{if } p^{m-j} \nmid y, \end{cases}$$

for some constant K.

B.3 $\mathbb{Z}/n\mathbb{Z}$ Case

Given a multiplicative character of $\mathbb{Z}/n\mathbb{Z}$, $\chi_{l_1,\ldots,l_k}$ the vector with component x equal to $\chi_{l_1,\ldots,l_k}(x)$ is equal to the tensor product of the corresponding vectors of the $\chi_{l_1}^{(1)}, \ldots, \chi_{l_k}^{(k)}$.

The Fourier transform of $\chi_{l_1,\ldots,l_k}$ will then be the tensor product of the Fourier transforms of the $\chi_{l_1}^{(1)}, \ldots, \chi_{l_k}^{(k)}$. So if the period of $\chi_{l_1,\ldots,l_k}$ is T, the Fourier transform of $\chi_{l_1,\ldots,l_k}$ is

$$\hat{\chi}_{l_1,\ldots,l_k}(\psi_y) = \begin{cases} K\overline{\chi_{l_1,\ldots,l_k}(y/T)} & \text{if } T|y, \\ 0 & \text{if } T \nmid y, \end{cases}$$

for some constant K.