Quantum Algorithms for Abelian Difference Sets and Applications to Dihedral Hidden Subgroups

Martin Roetteler

Microsoft Research, Quantum Architectures and Computation Group, One Microsoft Way, Redmond, WA 98052, USA
`martinro@microsoft.com`

─── **Abstract** ───────────────────────────────────

Difference sets are basic combinatorial structures that have applications in signal processing, coding theory, and cryptography. We consider the problem of identifying a shifted version of the characteristic function of a (known) difference set and present a general algorithm that can be used to tackle any hidden shift problem for any difference set in any abelian group. We discuss special cases of this framework which include (a) Paley difference sets based on quadratic residues in finite fields which allow to recover the shifted Legendre function quantum algorithm, (b) Hadamard difference sets which allow to recover the shifted bent function quantum algorithm, and (c) Singer difference sets which allow us to define instances of the dihedral hidden subgroup problem which can be efficiently solved on a quantum computer.

1 Introduction

Many exponential speedups in quantum computing are the result of solving problems that belong to either the class of hidden subgroup problems (HSPs) or the class of hidden shift problems. For instance, the problems of factoring integers and of computing discrete logarithms in abelian groups [44] can be reformulated as solving instances of hidden subgroup problems in abelian groups [34, 26, 27, 6, 22, 23]: given a function f from an abelian group A to a set, so that f is constant on the cosets of some subgroup $H \leq A$ and takes distinct values on different cosets, the task is to find generators of H.

Successes of the hidden subgroup framework include period finding over the reals which was used by Hallgren to construct an efficient quantum algorithm for solving Pell's equation [18, 24] and more recently to the discovery of a quantum algorithm for computing unit groups of number fields of arbitrary degree [12]. Moreover, the hidden subgroup problem over symmetric and dihedral groups are related to the graph isomorphism problem [5, 2, 19, 13] and some computational lattice problems [40]. Constructing efficient algorithms for these problems are two major open questions in quantum algorithms.

As far as hidden shift problems are concerned, the shifted Legendre function problem [47], shifted sphere problems and shifts of other non-linear structures [8], problems of finding shifts of non-linear Boolean functions [43, 7] can be reformulated as solving instances of hidden shift problems in abelian groups: given a pair (f, g) of functions from an abelian group A to a set so that g is obtained from f by shifting the argument by an unknown shift

(a)   (b)   (c)   (d)
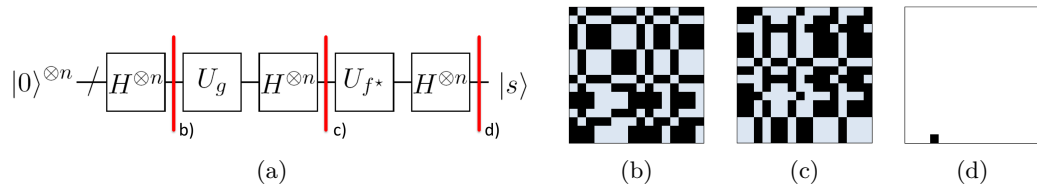
■ **Figure 1** An example for hidden shift problem $g(x) = f(x \oplus s)$ over $A = \mathbb{Z}_2^{256}$ where the instance f is given by a bent function and $f^\star$ is the dual bent function of f. Shown in (a) is the circuit for the correlation-based algorithm from [43], where the red marker denotes the state at the respective point in time during the algorithm's execution. Shown in (b), (c), and (d) are visualizations of three stages during the algorithm's execution: (b) is the state after the shifted function has been computed into the ± 1-valued phase. Here black and light blue color stand for (re-normalized) values of $+1$ and -1, respectively. Shown in (c) is the state after the Fourier transform. As bent functions have a flat spectrum in absolute value, the state is again two valued at this point. Finally, in (d) the state after the final Hadamard transform is shown, after which all amplitude is supported on the shift $|s\rangle$. Here black denotes an amplitude of 1 and white an amplitude of 0. The main contribution of this paper, Algorithm 11, can be considered a generalization of this picture to more general classes of hiding functions f. These are obtained from difference sets which, in a precise sense, generalize the notion of bent functions.

$s \in A$, the task is to find this shift. For further background on hidden subgroup and hidden shift problems see [35, 27, 25, 9, 31].

An intriguing connection exists between *injective* instances of the hidden shift problem over abelian groups A and the hidden subgroup problem for semidirect products of the form $A \rtimes \mathbb{Z}_2$ where the action of $\mathbb{Z}_2$ is given by inversion. This connection includes the special case of the hidden shift problem over the cyclic groups $A = \mathbb{Z}_N$, where N is a large integer which are related to the dihedral groups $D_N = \mathbb{Z}_N \rtimes \mathbb{Z}_2$ via this connection. Despite much effort, a fully polynomial-time quantum algorithm for the hidden subgroup problem over the dihedral groups has remained elusive. In this paper we make a step toward solving the hidden subgroup problem over the dihedral groups by exhibiting some instances that can be solved efficiently on a quantum computer. By efficient we mean that the run-time of the quantum part of the computation is bounded polynomially in the input size, which is generally assumed to be $\log A$, and the run-time of the classical post-processing part of the computation is also bounded polynomially in the input size.

1.1   Our results

Based on the combinatorial structure of difference sets we derive a class of functions that have a two-level Fourier power spectrum. We then consider the hidden shift problem for these functions, following a general algorithm principle that was used earlier to solve the hidden Legendre symbol [47] and the hidden bent function problem [43].

The basic idea underlying all these algorithms is to use the fact that the quantum computer can perform quantum Fourier transforms efficiently. This is used in correlation-based techniques which try to identify a shift by first transforming the function into frequency (Fourier) domain, then performing a point-wise multiplication with the desired target correlator, followed by an inverse Fourier transform and a measurement in the computational basis. After these steps the shift might be obtained, even without further post-processing. An example for this approach is shown in Figure 1 where the underlying group is the Boolean hypercube and the shifted function is a so-called bent function.

A rich theory of difference sets exists and many explicit constructions are known. Furthermore, several applications of difference sets exist in signal processing [38], coding theory

[32], cryptography [37], see also [4] for further examples. In this paper we focus on the case of difference sets in abelian groups and show that a correlation-based approach can be successfully applied to several families of difference sets. In one application we consider so-called Singer difference sets, which are difference sets in cyclic groups. These difference sets have parameters

$$(v, k, \lambda) = \left(\frac{q^{d+1} - 1}{q - 1}, \frac{q^d - 1}{q - 1}, \frac{q^{d-1} - 1}{q - 1} \right),$$

where q is a constant and d is a parameter that defines the input size of the problem, and v, k, and λ are characteristic parameters of the difference set. We construct instances of dihedral hidden subgroup problems that can be (query) efficiently solved on a quantum computer. There is one step in our algorithm that requires the implementation of a diagonal operator whose diagonal elements are certain generalized Gauss sums whose flatness follows from a classic result due to Turyn [46]. In general, we do not known how to implement these diagonal operators efficiently and it seems that the actual computational cost has to be determined on a case-by-case basis. However, for the case of $N = 2^n - 1$ we can leverage a result by van Dam and Seroussi [48] to implement a quantum algorithm that is fully efficient in terms of its quantum complexity as well as classical complexity. Classically, the underlying problem of this white-box problem is at least as hard as the discrete logarithm problem over a finite field.

1.2 Related work

Several papers study the dihedral hidden subgroup, however, it is an open whether quantum computers can solve this problem efficiently. There is a quantum algorithm which is fully efficient in its quantum part, which however requires an exponential-time classical post-processing [14]. Furthermore, a subexponential-time quantum algorithm for the dihedral subgroup problem is known [28, 41, 29] based on a sieving idea. The dihedral hidden subgroup problem for adversarially chosen hiding functions is believed to be intractable on a quantum computer, even we are not aware of any evidence stronger for this intuition than reductions from lattice problems [40] and subset sum type problems [40, 1]. The connection between hidden shift problems over abelian groups and hidden subgroup problems over semidirect groups of the mentioned special form is well-known and was one of the reasons why the hidden shift problem has been studied for various groups [14, 47, 15, 33, 10, 21, 9].

The study of hidden shift problems has resulted in quantum algorithms that are of independent interest and have even inspired cryptographic schemes that might be candidates for post-quantum cryptography [39]. Besides the mentioned works, problems of hidden shift type were also studied in [42, 16, 17], in the rejection sampling [36] framework, and in the context of multiregister PGM algorithms for Boolean hidden shift problems [7]. The main result of this paper is Theorem 17 which asserts that there exist instances of the hidden subgroup problem over the dihedral groups D_N that can be solved in $O(\log N)$ queries to the hiding function, $O(polylog(N))$ quantum time, $O(\log N)$ quantum space, and trivial classical post-processing. Moreover, for $N = 2^n - 1$, where $n \geq 2$, there exist instances of the hidden subgroup problem over the dihedral group $D_{2^n - 1}$ for which the hiding function is white-box and for which the entire quantum computation can be performed in $O(poly(n))$ quantum time, $O(n)$ quantum space, and trivial classical post-processing. Moreover, the classical complexity of solving these instances is at least as hard as solving the discrete logarithm problem over finite fields. To the best of our knowledge this is the first exponential size family of instances of the dihedral hidden subgroup problem that can be solved efficiently on

a quantum computer, whereas for the same class of instances no efficient classical algorithm is known[1].

In Corollary 18 we show that for D_N where $N = 2^n - 1$ this theorem implies that there are an expected number of $O(2^{n^2})$ instances of the dihedral HSP (where the hidden subgroup is a reflection) that can be solved efficiently on a quantum computer. This is a small fraction of the set of all instances of such hidden subgroup problems as the number of all instances scales doubly exponential as $O(2^{n2^n})$. In particular, it seems unlikely that the set of such constructed instances has a non-trivial intersection with the set of instances that can be obtain via Regev's reduction from gapped unique-SVP lattice problems.

The rest of this paper is organized as follows. First, in Section 2 we introduce some notation and basic definitions such as Fourier transform, convolution, and the basic combinatorial object of study in this paper, namely difference sets in finite abelian groups. Next, in Section 3 we present a quantum algorithm that can be applied to any shifted difference set problem, albeit sometimes with low probability of success. We exhibit some instances of shifted difference set problems that can be solved efficiently. These special cases include the so-called class of Singer difference sets which are then used in Section 4 to construct instances of the dihedral hidden subgroup problem that can be solved efficiently on a quantum computer. Finally, in Section 5 we offer conclusions and end with some open problems.

2   Background

2.1   Quantum Fourier transforms over abelian groups

The main tool we will use are Fourier transforms over abelian groups. In the following we state some basic definitions and properties. Recall that for any abelian group A the character group $\widehat{A} = \text{Hom}(A, \mathbb{C}^\times)$ is isomorphic to A. We denote the irreducible characters of A by $\chi : A \to \mathbb{C}^\times$.

▶ **Definition 1.** The *quantum Fourier transform* on $\mathbb{C}^d$ is a unitary transformation defined as $\text{QFT}_A := \frac{1}{\sqrt{|A|}} \sum_{a \in A} \sum_{\chi \in \widehat{A}} \chi(a)|\chi\rangle\langle a|$.

▶ **Example 2.** For $A = \mathbb{Z}_2$ the QFT_A is given by the Hadamard transform $H := \frac{1}{\sqrt{2}} \left(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix} \right)$.

▶ **Definition 3.** The *Fourier transform* of a (complex-valued) function $F : A \to \mathbb{C}$ is a function $\hat{F} : \widehat{A} \to \mathbb{C}$ defined as $\hat{F}(\chi) := \langle\chi|\text{QFT}_A|F\rangle$ where $|F\rangle := \sum_{x \in A} F(x)|x\rangle$. Here $\hat{F}(\chi)$ is called the *Fourier coefficient* of F at $\chi \in \widehat{A}$. We can write it explicitly as $\hat{F}(\chi) = \frac{1}{\sqrt{|A|}} \sum_{x \in A} \chi(x)F(x)$. The set $\{\hat{F}(\chi) : \chi \in \widehat{A}\}$ is called the *Fourier spectrum* of F.

▶ **Definition 4.** The *convolution* of functions $F, G : A \to \mathbb{C}$ is a function $(F * G) : A \to \mathbb{C}$ defined as $(F * G)(x) = \sum_{y \in A} F(y)G(x - y)$.

▶ **Fact 5.** Let $F, G, H : A \to \mathbb{C}$ denote arbitrary functions. The Fourier transform and convolution have the following basic properties:
1. The Fourier transform is linear: $\widehat{F + G} = \hat{F} + \hat{G}$.

---

[1] It is easy to see that it is possible to find instances of the dihedral hidden subgroup problem that can be solved efficiently on a classical computer, e.g., functions that identify the points of a regular N-gon that are opposites along a symmetry axis in a linear increasing fashion. On these "taco"-like instances the hidden symmetry axis, and thereby the hidden subgroup, can be found simply by a binary search.

2. When applied twice, the Fourier transform satisfies $\hat{\hat{F}}(z) = F(-z)$. In particular, for $A = \mathbb{Z}_2$ the Fourier transform is self-inverse: $\hat{\hat{F}} = F$. From this property also follows that when the Fourier transform QFT_A is applied four times then the result is the identity.

3. QFT is unitary, so the Plancherel identity $\sum_{\chi \in A} |\hat{F}(\chi)|^2 = \sum_{x \in A} |F(x)|^2$ holds.

4. The convolution is commutative: $F * G = G * F$, and associative: $(F * G) * H = F * (G * H)$.

5. The Fourier transform and convolution are related through the following identities: $(\hat{F} * \hat{G})/\sqrt{|A|} = \widehat{FG}$ and $(\widehat{F * G})/\sqrt{|A|} = \hat{F}\hat{G}$, where $FG : A \to \mathbb{C}$ is the entry-wise product of functions F and G: $(FG)(x) := F(x)G(x)$.

6. A shift of a function in time domain leads to a point-wise multiplication with a "linear phase" in Fourier domain: If there exists $s \in A$ such that for all $x \in A$ it holds $G(x) = F(x - s)$, then for all $\chi \in \hat{A}$ we have that $\hat{G}(\chi) = \chi(s)\hat{F}(\chi)$. This latter property will be crucial for the hidden shift algorithm presented later in this paper.

2.2 Difference sets

We recall the definition of difference sets in finite groups. We focus on the case of abelian groups in this paper. See also [3, 45, 30] for further information, in particular about the treatment for general, non-abelian groups.

Let A be a finite abelian group whose group operation we write additively and whose neutral element we denote with 0_A. Denote the pairwise inequivalent irreducible characters of A by $\hat{A}$. For a subset $D \subseteq A$ of A we introduce the notation $D^- := \{-d : d \in D\}$ for the set of all inverses and $\Delta D := D + D^- = \{x - y : x, y \in D\}$ for the set of all differences of pairs of elements of D.

▶ **Definition 6** (Difference set)**.** Let A be a finite abelian group of size $v = |A|$. A subset $D \subseteq A$ of size $k = |D|$ is called a (v, k, λ)-difference set, where $\lambda \geq 1$, if the following equality holds in the group algebra $\mathbb{C}[A]$ of A:

$$\Delta D = \lambda(A \setminus \{0_A\}) + k0_A. \tag{1}$$

This means that the set of all differences covers each element the same number λ of times, except for the neutral element, which is covered precisely k times. A nice feature of difference sets in abelian group is that they allow to construction functions with almost flat spectrum: the following theorem [46] asserts that all Fourier coefficients of the characteristic function of a difference set in an abelian group have the same absolute value, with a possible exception of a peak at the zero frequency:
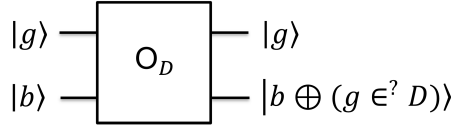
▶ **Theorem 7** (Turyn, 1965)**.** *Let A be an abelian group of order v and D be an (v, k, λ)-difference set in A. Let $\chi \in \hat{A}$ be a non-trivial character. Then*

$$|\chi(D)| := \left| \sum_{d \in D} \chi(d) \right| = \sqrt{k - \lambda} \tag{2}$$

holds. For the trivial character χ_0 we have that $|\chi_0(D)| = k$.

Proof. We include a proof as it is instructive to see how the difference set condition can be used when interpreted as the identity (1) in the group ring $\mathbb{C}[A]$. Indeed, when identifying D with $\sum_{d \in D} d \in \mathbb{C}[A]$, we obtain from eq. (1) that

$$\left(\sum_{d \in D} d \right) \left(\sum_{d \in D} -d \right) = \lambda \left(\sum_{g \in A} g \right) + (k - \lambda)0_A. \tag{3}$$

$$|g\rangle \quad \boxed{O_D} \quad |g\rangle$$
$$|b\rangle \qquad\qquad |b \oplus (g \in^? D)\rangle$$

Figure 2 The oracle for the shifted difference set problem considered in this paper. The oracle allows to test membership of a given test input $g \in A$. The value of the test $g \in^? D$ is XORed onto a bit $b \in \{0, 1\}$.

Let $\chi \in \widehat{A}$ be non-trivial. Then clearly $\chi(A) = 0$ holds which implies—by applying χ to both sides of eq. (3)—that $\chi(D)\overline{\chi(D)} = \chi(A) + (k - \lambda)\chi(0_A) = k - \lambda$. From this we obtain that $|\chi(D)| = \sqrt{k - \lambda}$ as claimed. ◀

With each difference set D we can canonically associate an incidence structure called the *development* of D, and denoted by $Dev(D)$.

▶ **Definition 8.** Let D be a (v, k, λ)-difference set in an abelian group A. Then the points of $Dev(D)$ are given by the elements of A and the blocks of $Dev(D)$ are given by $v + D := \{v + a : a \in D\}$, where $v \in A$.

It is well-known that $Dev(D)$ is a symmetric design. More precisely, we have the following result (for a proof see, e.g., [3], [45] or [30]):

▶ **Theorem 9.** *Let D be a (v, k, λ)-difference set in an abelian group A. Then $Dev(D)$ is a symmetric balanced-incomplete block design with parameters (v, k, λ).*

Theorem 9 implies that there are $|A|$ blocks, that each block has $|D|$ elements, that any two elements have precisely λ blocks in common and that in addition any two blocks intersect in precisely λ points. Also, it holds that $\lambda = k(k - 1)/(v - 1)$, see e.g. [30, Prop. 1.1], implying that λ is determined by the group order v of A and the size k of D. This equality allows us also to do a consistency check that the normalized state vector $\frac{1}{\sqrt{k}} \sum_{d \in D} |d\rangle$ is indeed mapped to a normalized vector under the Fourier transform QFT_A for the group A: using Theorem 7 we find that the length of the transformed vector is given by

$$\frac{1}{\sqrt{vk}^2} \left((v - 1)|\chi(D)|^2 + |\chi_0(D)|^2 \right) = ((v - 1)(k - \lambda) + k^2)/(vk)$$
$$= ((v - 1)k - k(k - 1) + k^2)/(vk) = 1,$$

as desired.

3 Quantum algorithm for shifted difference sets

▶ **Problem 10** (Shifted difference set problem). Let A be an abelian group and let $s \in A$. Let $D \subseteq A$ be a (known) difference set and let $s + D$ be given by a membership oracle. The problem is to find s.

Similar to [43] we can modify Problem 10 by hiding not only the characteristic function of $s + D$ via a membership oracle but also the characteristic function of D itself. In this case we assume that we have access to membership oracles for both D and $s + D$. The following quantum algorithm is a general recipe to tackle instances of the shifted difference set problem specified in Problem 10. As we will show in the following, Algorithm 11 can be

used to find the hidden shift s efficiently in several cases of difference sets for various abelian groups A. It should be noted, however, that the probability of success crucially depends on the instance (A, D) of the problem and there are instances for which the algorithm recovers s successfully is only exponentially small. The algorithm can be seen as a generalization of correlation-based algorithms for solving hidden shift problems, e.g., [47], [43], and [7].

▶ **Algorithm 11.** The input to the algorithm is a membership oracle as in Problem 10.
Step 1: Prepare the input superposition:

$$|0\rangle \mapsto \frac{1}{\sqrt{|A|}} \sum_{g \in A} |g\rangle.$$

Step 2: Query the shifted difference set. This maps the state to:

$$\frac{1}{\sqrt{|A|}} \sum_{g \in A} (-1)^{(g \in^? s+D)} |g\rangle = \frac{1}{\sqrt{|A|}} \sum_{g \in A} |g\rangle - \frac{2}{\sqrt{|A|}} \sum_{d \in (s+D)} |d\rangle.$$

Step 3: Apply the quantum Fourier transform for A. This maps the state to:

$$|\chi_0\rangle - \frac{2}{|A|} \sum_{\chi \in \widehat{A}} \chi(s+D)|\chi\rangle = \left(1 - \frac{2k}{|A|}\right)|\chi_0\rangle - \frac{2}{|A|} \sum_{\chi \neq \chi_0} \chi(D)\chi(s)|\chi\rangle.$$

Step 4: Compute $\mathrm{diag}(1, \overline{\chi(D)}/\sqrt{k-\lambda} : \chi \neq \chi_0)$ into the phase. This maps the state to:

$$\left(1 - \frac{2k}{|A|}\right)|\chi_0\rangle - \frac{2(k-\lambda)}{|A|} \sum_{\chi \neq \chi_0} \chi(s)|\chi\rangle.$$

Step 5: Apply the inverse quantum Fourier transform for A. This maps the state to:

$$\frac{1}{\sqrt{|A|}} \left(1 - \frac{2(k - \sqrt{k-\lambda})}{|A|}\right) \sum_{g \in A} |g\rangle - \frac{2\sqrt{k-\lambda}}{\sqrt{|A|}} |-s\rangle$$

Step 5: Measure in the standard basis. Obtain $-s$ with probability $p := \frac{4(k-\lambda)}{|A|}$ and all other group elements uniformly with probability $(1-p)/|A|$.

3.1 Examples

3.1.1 Paley difference sets and shifted Legendre functions

Let A be the additive group of the finite field $\mathbb{F}_q$, where $q = p^n$ is a prime power such that $q \equiv 3 \pmod 4$. Define $D := \{x : x \text{ is a non-zero square in } \mathbb{F}_q\}$. It is well-known [3, 45] that D is then a difference set in A. These difference sets are also known as Paley difference sets. The parameters of D are as follows:

$$(v, k, \lambda) = \left(q, \frac{q-1}{2}, \frac{q-3}{4}\right).$$

▶ **Example 12.** Let $q = 27$ and consider the irreducible polynomial $f(x) = x^3 + x^2 + x + 2 \in \mathbb{F}_3[x]$, defining the finite field $\mathbb{F}_{27} \cong \mathbb{F}_3[x]/(f(x))$. Denote by $\{1, \alpha, \alpha^2\}$ an $\mathbb{F}_3$-basis of $\mathbb{F}_{27}$ where $\alpha := x \bmod f(x)$, the image of x under the canonical projection. Then D given by the following 13 elements

$$\begin{aligned} D = \{ & 1, \quad \alpha, \quad 2\alpha^2 + 2\alpha + 1, \quad 2\alpha + 2, \quad \alpha + 2, \quad \alpha^2 + 2\alpha, \quad \alpha^2 + 1, \quad 2\alpha^2 + 1, \\ & \alpha^2 + \alpha + 1, \quad \alpha^2, \quad 2\alpha^2 + 2\alpha, \quad \alpha^2 + 2\alpha + 1, \quad \alpha^2 + 2\alpha + 2 \} \end{aligned}$$

defines a $(27, 13, 6)$-difference set in $\mathbb{Z}_3^3$.

▶ Remark. Applying Algorithm 11 finds the hidden shift with probability of success

$$p_{success} = \left| \frac{2 \left(\frac{q-1}{2} - \frac{q-3}{4} \right)^{1/2}}{q^{1/2}} \right|^2 \approx 1 - O(1/q).$$

This means that for large q, we can efficiently recover the hidden shift. In this case, the Algorithm 11 specializes to the algorithm given in [47]. We recover the result that an unknown shift of the Legendre symbol can be reconstructed with high probability using 1 query.

3.1.2 Hadamard difference sets and shifted bent functions

Let A be the elementary abelian 2-group $A = \mathbb{Z}_2^{2n}$, where $n \in \mathbb{N}$. Let $f : \mathbb{Z}_2^{2n} \to \mathbb{Z}_2$ be a bent function. Define $D := \{x \in \mathbb{Z}_2^{2n} : f(x) = 1\}$. It is well-known [3, 45] that D is a difference set in A. These difference sets are also known as Hadamard difference sets. The parameters of D are as follows:

$$(v, k, \lambda) = \left(2^{2n}, \ 2^{2n-1} - 2^{n-1}, \ 2^{2n-2} - 2^{n-1} \right).$$

▶ **Example 13.** Let $n = 4$ and let $f(x_1, x_2, x_3, x_4) = x_1 x_2 \oplus x_3 x_4 \oplus x_1 \in \mathbb{F}_2[x_1, x_2, x_3, x_4]$ be a bent function from the Maiorana-McFarland family [11]. Then $D = \{x \in \mathbb{F}_2^4 : f(x) = 1\}$ given by the following 6 elements

$$D \quad = \quad \{(1,1,0,0), \quad (1,1,1,0), \quad (1,1,0,1), \quad (0,0,1,1), \quad (1,0,1,1), \quad (0,1,1,1)\}$$

defines a $(16, 6, 2)$-difference set in $\mathbb{Z}_2^4$. The blocks of the development $Dev(D)$ of D are obtained by taking the characteristic function of f and shifting it under all elements of $A = \mathbb{Z}_2^4$. Hence, the incidence matrix of the $(16, 6, 2)$-design $Dev(D)$ is given by

$$\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix}.$$

Applying Algorithm 11 to the shifted difference problem for a Hadamard difference set finds the hidden shift with probability of success

$$p_{success} = \left| \frac{2 \left(2^{2n-1} - 2^{2n-2} \right)^{1/2}}{2^{2n^{1/2}}} \right|^2 = 1.$$

This means that we always recover the hidden shift s with probability 1. In this case, the Algorithm 11 specializes to the algorithm given in [43]. We recover the result that an unknown shift of a bent function can be reconstructed using 1 query.

3.1.3 Singer difference sets and shifted hyperplanes

Let q be a prime power, let $d \geq 1$ and let $\mathbb{F}_{q^{d+1}}$ be the finite field with q^{d+1} elements. The Singer difference sets are constructed from d-dimensional projective spaces over $\mathbb{F}_q$ as follows: consider the trace map tr from $\mathbb{F}_{q^{d+1}}$ to $\mathbb{F}_q$. Let T be a transversal of $\mathbb{F}_q^*$ in $\mathbb{F}_{q^{d+1}}^*$ that is chosen in such a way that tr maps T onto the values 0 and 1 in $\mathbb{F}_q$ only. We can then define a group $A := \mathbb{F}_{q^{d+1}}^{\times}/\mathbb{F}_q^{\times}$ which turns out to be cyclic. Furthermore, we can define a subset $D := \{x : x \in A | \mathrm{tr}(x) = 0\}$. It turns out [3] that D is then a difference set in $\mathbb{Z}_N$, where $N = \frac{q^{d+1}-1}{q-1}$. This difference set has parameters

$$(v, k, \lambda) = \left(\frac{q^{d+1}-1}{q-1}, \frac{q^d-1}{q-1}, \frac{q^{d-1}-1}{q-1} \right). \tag{4}$$

▶ **Example 14.** Let $P = PG(2,3)$ be the two-dimensional projective space over $\mathbb{F}_3$. Then $|P| = (3^3 - 1)/(3 - 1) = 13$. By choosing an $\mathbb{F}_3$-basis of $\mathbb{F}_{27}$ we obtain an embedding of $\mathbb{F}_{27}^{\times}$ into $\mathrm{GL}(3, \mathbb{F}_3)$. If $\alpha \in \mathbb{F}_{27}$ is a primitive element for $\mathbb{F}_{27}/\mathbb{F}_3$, then the corresponding matrix has order 26 and therefore generates a cyclic subgroup C of $\mathrm{GL}(3, \mathbb{F}_3)$ order 26. Under the canonical projection $\pi : \mathrm{GL}(3, \mathbb{F}_3) \to \mathrm{PGL}(3, \mathbb{F}_3)$, the subgroup C is mapped to a subgroup $\overline{C} = \langle \sigma \rangle$ of $\mathrm{PGL}(3, \mathbb{F}_3)$ of order 13 (see also [20, Kapitel II, Satz 7.3]). This subgroup is sometimes also called the "Singler cycle." The Singer cycle operates transitively on the points $\{(x : y : z) : x, y, z \in \mathbb{F}_3\}$ of the projective space P. By picking the particular order $[\sigma^i p_0 : i = 0, \ldots, 12]$, where p_0 is the point $(0 : 0 : 1)$, we obtain points that we can identify with $[0, 1, \ldots, 12]$. The image of the hyperplane given by all points $p \in \mathbb{F}_{27}$ with $\mathrm{tr}(p) = 0$ is given by the set $D := \{0, 1, 3, 9\}$. Then D is a $(13, 4, 1)$-difference set in the cyclic group $\mathbb{Z}_{13}$. The development $Dev(D)$ is given by:

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}.$$

If in eq. (4) we consider q to be constant and d be a parameter that corresponds to the input size of a hidden shift problem over $\mathbb{Z}_N$, we can use Algorithm 11 to solve the hidden shift problem over $\mathbb{Z}_N$ with probability of success

$$p_{success} = \left| \frac{4(q^d - q^{d-1})^{1/2}}{(q^{d+1} - 1)^{1/2}} \right|^2 = \frac{2}{q} + O(1/q^2).$$

This means that for constant q, we can efficiently recover the hidden shift from a constant number of trials. In Section 4 we show how we can use the instances of hidden difference problems of Singer type to construct efficiently solvable instances of the dihedral hidden subgroup problem.

▶ **Remark.** We note that not all shifted difference set problems can be solved efficiently by using Algorithm 11. An example is given by the projective planes $(q^2 + q + 1, q + 1, 1)$ of order q. In this case the input size is given by $\log q$ and the probability of success can be computed to be $p_{success} = |\frac{2q^{1/2}}{(q^2+q+1)^{1/2}}|^2 \approx \frac{2}{q} + O(1/q^2)$, i.e., the probability of success is exponentially small in this case. It is an open problem if cases like this can be tackled, e.g., by considering multi-register algorithms.

3.2   Injectivization

As mentioned in the introduction, it is well known that the hidden subgroup problem over semidirect products of the form $A \rtimes \mathbb{Z}_2$, where the action of $\mathbb{Z}_2$ is given by inversion, and the hidden shift problem over A are closely related. More precisely, there is a one-to-one correspondence between instances of the hidden subgroup problem in which the subgroup is a conjugate of the order 2 subgroup $H = \langle (0, 1) \rangle$ and instances of *injective* hidden shift problems over A.

This leads to the question whether it is possible to relate instances of hidden shift problems where the hiding function $f : A \to S$ is not injective to the injective case. Thankfully, as shown in [17] such a connection indeed exists. We briefly review this construction.

For given $f : A \to S$, and a set $V := \{v_1, \dots, v_m\} \subseteq A$ of m elements of A we define a new function $f_V(x) := (f(x + v_1), \dots, f(x + v_m))$. Gharibi showed in [17] that if the set V is chosen uniformly at random, then the probability that the function f_V is not injective can be upper bounded as

$$Pr_V(f_V \text{ not injective}) \leq |A|^2 (1 - \gamma_{min})^m, \tag{5}$$

where $\gamma_{min} := \min_{v \neq 0}(\gamma_v(f))$ and for all $v \in A$ the so-called influences $\gamma_v(f)$ of f at v are defined as $\gamma_v(f) := Pr_x(f(x) \neq f(x + v))$, i.e., the probability that f changes its value when the input is toggled by v.

We now show that for instances of shifted difference set problems these influences can be bounded by the parameters of the difference set alone. This in turn allows to establish a bound on the overall number of copies m that are needed to make the hiding function injective, namely a bound that grows proportional to $\log |A|$.

▶ **Lemma 15.** *Let $f : A \to \{0, 1\}$ be a hiding function corresponding to the characteristic function a (v, k, λ)-difference set in an abelian group A. Then for all $v \in A \setminus \{0\}$ we have that $\gamma_v(f) = \frac{2(k-\lambda)}{|A|}$.*

Proof. Note that

$$Pr_x(f(x) \neq f(x + v)) \quad = \quad \frac{1}{|A|} \sum_{x \in A} (f(x) - f(x + v))^2 \tag{6}$$

$$= \quad \frac{1}{|A|} (|D| + |v + D| - 2|D \cap (v + D)|) \tag{7}$$

$$= \quad \frac{1}{|A|} (2|D| - 2\lambda) = \frac{2(k - \lambda)}{|A|}, \tag{8}$$

where in the second equation we used the fact that only elements in the intersection contribute to $f(x)f(x + v)$ and in the third equation we used Theorem 9 which implies that $|D \cap (v + D)| = \lambda$ for all $v \neq 0$. ◀

We can now establish the claimed result that the number of copies only grows with the log of the group size.

▶ **Theorem 16.** *Let D be a (v, k, λ)-difference set in an abelian group A and $f : A \rightarrow \{0, 1\}$ an instance of a hidden difference set problem for D. Then $m = O(\log |A|)$ copies are enough to obtain an injective instance f_V with probability greater than $1 - \frac{1}{64}$.*

Proof. From the cited bound (5) we obtain that

$$Pr(f \text{ injective}) \geq 1 - |A|^2 (1 - \gamma_{min}(f))^m$$

It is easy to see that lower bounding the right hand side in this expression by $1 - \frac{1}{64}$ is equivalent to choosing $m \geq \frac{1}{\log(1 - \gamma_{min}(f))}(-6 - 2\log_2(|A|))$. Now, from Lemma 15 we have that $\gamma_{min}(f) = \frac{2(k-\lambda)}{|A|}$ from which we can conclude that in particular $|A| \geq 2(k - \lambda)$ holds. Using the fact that $\log_2(1 - x) \leq -x$ holds for $x \in [0, 1)$, this implies that

$$m \quad \geq \quad \left\lceil \frac{1}{\log_2\left(1 - \frac{2(k-\lambda)}{|A|}\right)}(-6 - 2\log_2|A|)\right\rceil \tag{9}$$

$$\geq \quad \left\lceil -\frac{|A|}{2(k-\lambda)}(-6 - 2\log_2|A|)\right\rceil \geq 2\log_2|A| + 6. \tag{10}$$

Hence $m = O(\log |A|)$ copies are enough to guarantee that for $V = \{v_1, \ldots v_m\}$ chosen uniformly at random, the probability of f_V being injective is at least $1 - \frac{1}{64}$.   ◀

4   Efficiently solvable dihedral hidden subgroup problems

▶ **Theorem 17.** *There exist instances of the hidden subgroup problem over the dihedral groups D_N that can be solved in $O(\log N)$ queries to the hiding function, $O(polylog(N))$ quantum time, $O(\log N)$ quantum space, and trivial classical post-processing. Moreover, for $N = 2^n - 1$, where $n \geq 2$, there exist instances of the hidden subgroup problem over the dihedral group $D_{2^n - 1}$ for which the hiding function is white-box and for which the entire quantum computation can be performed in $O(poly(n))$ quantum time, $O(n)$ quantum space, and trivial classical post-processing. Moreover, the classical complexity of solving these instances is at least as hard as solving the discrete logarithm problem over finite fields.*

Proof. To construct the instances that can be solved efficiently we proceed in three steps: (i) first, we show that a particular set of hidden shift problems over $\mathbb{Z}_N$ can be obtained from hiding functions that are indicator functions of hyperplanes and that these indicator functions can be implemented efficiently, (ii) next we show that Algorithm 11 is query, time, and space efficient for these instances; (iii) finally, we show that it is possible to construct instances of the hidden subgroup problem in D_N from the hidden shift instances constructed in (i) and that these instances are unlikely to be solvable on a classical computer, unless computing finite field discrete logarithms is possible in polynomial-time.

Step (i): We instantiate the abelian difference set quantum algorithm for the case of the cyclic group $A = \mathbb{Z}_N$, where $N = (q^{d+1} - 1)/(q - 1) = q^d + q^{q-1} + \ldots + 1$. Here q is constant and d is a parameter that corresponds to the input size of the problem. We use the explicitly (white-box) description of the function $f(x) = \text{tr}(\alpha^x)$, where tr denotes the trace map from $\mathbb{F}_q^{d+1}$ to $\mathbb{F}_q$ and where α is a primitive element in $\mathbb{F}_q$. Now, the instance of the shifted difference set problem is defined by the hiding function $g(x) = \text{tr}(\alpha^{x+s})$, where $s \in \mathbb{Z}_N$. This function can be given as a white-box function by providing the element $\beta := \alpha^s \in \mathbb{F}_{q^{d+1}}$ so that g can then be evaluated as $g(x) = \text{tr}(\alpha^x \beta)$. Note that the set $\{x \in A : \text{tr}(x) = 0\}$ defines a hyperplane and therefore a difference set D of Singer type.

Step (ii): We now go through each step of Algorithm 11 and check that the steps are time- and space-efficient. In the first step, a Fourier transform is applied to create the equal

superposition of all elements of A. As A is abelian, this can clearly be done efficiently. In the second step, we have to evaluate the function g in superposition. Again, as there is an explicit description of the trace which can be computed as sum of powers of the relative Frobenius from $\mathbb{F}_{q^{d+1}}$ to $\mathbb{F}_q$ as follows $\text{tr}(x) = x + x^q + \ldots + x^{q^d}$ we can evaluate $g(x) = \text{tr}(\alpha^x \beta)$ by first constructing a circuit for exponentiation $x \mapsto \alpha^x \in \mathbb{F}_{q^{d+1}}$ followed by scalar multiplication with β, followed by the application of the trace map. Clearly, all these operations can be efficiently implemented by means of a classical Boolean circuit whose size and depth are polynomial in d. Hence, by applying standard techniques from reversible computing, we can derive quantum circuits for the evaluation of f and g. Therefore we can compute Step 2 efficiently on a quantum computer.

Step 3 is another application of a quantum Fourier transform over the abelian group A which as in Step 1 can be done efficiently. Step 4 is the most challenging step in the entire algorithm. If we were just interested in the query complexity of the problem we would be done as we could simply apply the diagonal unitary operator $\Delta := \text{diag}(1, \overline{\chi_1(D)}, \ldots, \overline{\chi_{N-1}(D)})$, where $\chi_1, \ldots, \chi_{N-1}$ runs through all non-trivial characters of $\mathbb{Z}_N$. This argument is sufficient to establish the first claimed statement in the theorem, i.e., the query complexity result.

For the white-box statement, we are interested in the time- and space-efficiency of the algorithm, i.e., we have to show that Δ can be implemented efficiently. For this we have to assume $N = 2^n - 1$ as required by one of the subsequent steps (and we highlight where). First we use a result due to van Dam and Seroussi [48] establishing that finite field Gauss sums can be approximated efficiently on a quantum computer. The connection to our situation is that the elements of Δ are Gauss sums. We briefly review the van Dam/Seroussi algorithm and then argue that we can apply it in superposition in order to compute Δ.

Let $\mathbb{F}_q$ be a finite field where $q = p^{d+1}$ and p prime. Let $\psi := \mathbb{F}_p \to \mathbb{C}^\times$ be a non-trivial additive character and let $\chi : \mathbb{F}_q^\times \to C^\times$ be a non-trivial multiplicative character. Then the Gauss sum $G(\psi, \chi)$ is defined as

$$G(\psi, \chi) = \sum_{x \in \mathbb{F}_q^\times} \chi(x) \psi(\text{tr}(x)).$$

The additive and multiplicative characters of $\mathbb{F}_q$ have a simple description: For $n \in \mathbb{N}$ denote a primitive n-th root of unity in $\mathbb{C}^\times$ with ω_n. Then the additive characters take the form $\psi_\mu(x) := \omega_p^{\text{tr}(\mu x)}$, where $\mu \in \mathbb{F}_q$ runs through all elements of $\mathbb{F}_q$. The multiplicative characters can be described using a primitive elements $\alpha \in \mathbb{F}_{p^{d+1}}$ as follows: $\chi_\beta(\alpha^i) := \omega_{p^{d+1}-1}^{\beta i}$, where β runs through all non-zero elements of $\mathbb{F}_{p^{d+1}}$. This means that evaluation $\chi_\beta(x) = \omega^{\beta \log_\alpha(x)}$ requires the computation of a discrete log over the multiplicative group of the field.

It is known that for non-trivial ψ and χ, the absolute value of the Gauss sum $G(\psi, \chi)$ evaluates to $|G(\psi, \chi)| = \sqrt{q}$, i.e., $G(\psi, \chi) = \sqrt{q} e^{i\theta}$, where $\theta \in [0, 2\pi)$. The paper [48] established that θ can be approximated with precision ε by a quantum algorithm in time $O(\frac{1}{\varepsilon} polylog(q))$. As we are overall only looking for a quantum algorithm that can solve the hidden shift problem over $\mathbb{Z}_N$ with bounded probability of success, it will be enough to approximate the diagonal elements of Δ with constant precision, i.e., we can use the van Dam/Seroussi algorithm to estimate $G(\psi, \chi)$. A minor complication is the fact that in [48] only the case of known character χ is considered, however, by making all steps of the algorithm conditioned on the character χ it can be easily seen that the transformation $|\chi\rangle \mapsto G(\chi, \psi)/\sqrt{q}|\chi\rangle$ can also be implemented coherently, i.e., on superposition of inputs χ. The final step is to show how to relate $\chi(D)$ and $G(\chi, \psi)$. For this we make the restriction

that $p = 2$ so that our parameters always take the form $N = 2^{d+1} - 1$. We then obtain that

$$
\begin{aligned}
\chi(D) \quad &= \sum_{x:\mathrm{tr}(x)=0} \chi(x) = \sum_{x \in \mathbb{F}_q^\times} \chi(x)(1 + (-1)^{\mathrm{tr}(x)}) = \sum_{x \in \mathbb{F}_q^\times} \chi(x) + \sum_{x \in \mathbb{F}_q^\times} \chi(x)(-1)^{\mathrm{tr}(x)} \\
&= \sum_{x \in \mathbb{F}_q^\times} \chi(x)(-1)^{\mathrm{tr}(x)} = G(\psi, \chi),
\end{aligned}
$$

where ψ denotes the additive character $\psi(x) := (-1)^{\mathrm{tr}(x)}$ of $\mathbb{F}_{2^n}$ and $\chi(x)$ denotes a multiplicative character of $\mathbb{F}_{2^n}^\times$. This argument establishes that we can approximate the operator Δ efficiently on a quantum computer with constant precision ε.

The final two steps of the algorithm are easy to do: Step 5 is just another Fourier transform and Step 6 a measurement in the computational basis, both of which can be done efficiently.

Step (iii): To construct the desired instances of the hidden subgroup problem from the hidden shift problem, we apply the results from Subsection 3.2 and specialize them to the case of the Singer difference sets. We pick $m = 2(d+1) + 6$ random elements $v_1, \ldots, v_m \in \mathbb{F}_{q^{d+1}}$ and construct the hiding function $g_{v_1, \ldots, v_m}(x) := (g(x + v_1), \ldots, g(x + v_m))$ which according to Theorem 16 is injective with probability greater than $1 - \frac{1}{64}$. We then apply another standard construction [15, 28] which allows to turn an instance of an injective hidden shift problem into a hidden subgroup problem. Indeed, if $f, g : A \to S$ is an injective instance of a hidden shift problem with shift $s \in A$, then the corresponding hidden subgroup problem over $A \rtimes \mathbb{Z}_2$ is given by the hiding function $F((a, 0)) := f(a)$ and $F((a, 1)) := g(a)$, where (a, t) is an encoding of the elements, i.e., $a \in A$ and $t \in \mathbb{Z}_2$. Conversely, if $F : A \rtimes \mathbb{Z}_2 \to S$ is a defining function of a hidden subgroup problem with hidden subgroup $H = \langle (a, 1) \rangle$ of order 2, then $f(x) := F(x, 0)$ and $g(x) := F(x, 1)$ defines a hidden shift problem over A.

Overall, we established the claimed result of the existence of an efficient quantum algorithm to solve the hidden subgroup problem. The classical complexity of finding the shift s from β clearly is as least as hard as solving the discrete logarithm over a finite field. ◄

▶ **Corollary 18.** *Let $N = 2^n - 1$, where $n \geq 2$, there there exist an expected number of 2^{n^2} instances of hidden subgroup problems over D_N that can be solved efficiently on a quantum computer.*

Proof. From the proof of Theorem 17 we see that in step (iii) for each random choice of $m = O(\log |A|)$ elements, where $|A| = |\mathbb{Z}_{2^n - 1}| = 2^n - 1$, we obtain a valid injectivization of the hidden shift function. There are an expected number of $O(|A|^m) = O((2^{\log_2(|A|)})^m) = O(2^{n^2})$ such functions. ◄

5 Conclusions

We showed that the property of difference sets to give rise to functions with two level Fourier (power) spectrum which makes them useful for classical applications also allows to define hidden shift problems which can then be tackled on a quantum computer. While a solution to general hidden shift problems for arbitrary difference sets remains elusive, we showed that several interesting special cases can indeed be solved efficiently on a quantum computer. This includes the known cases of the Legendre symbol which we show to be an instantiation of our framework for the case of a Paley difference set. Furthermore, it includes the case of hidden bent functions which we show to be special cases of Hadamard difference sets. The case of Singer difference sets appears to be new and allows us to construct white-box

instances of dihedral hidden subgroup problems that can be solved fully efficiently on a quantum computer, both in the quantum and in the classical parts of the algorithm.

Open problems include whether these findings have any consequence for more general classes of instances of the dihedral hidden subgroup problem and the hidden subgroup problem in other semidirect products of a similar form. Other open problems include whether it is possible to solve the shifted difference set problem for projective planes which we mentioned cannot be solved by our main algorithm with better than exponentially small probability of success. One possible avenue for future research is to consider multi-register algorithms to tackle this problem. Another open problem is the case of hidden shift problems over abelian groups for functions that have approximately constant spectra, possibly with the exception of the zero frequency as in case of the functions arising from difference sets considered in this paper.

References

1 Dave Bacon, Andrew M. Childs, and Wim van Dam. Optimal measurements for the dihedral hidden subgroup problem. *Chicago Journal of Theoretical Computer Science*, 2006(2), Oct 2006. quant-ph/0501044. URL: `http://cjtcs.cs.uchicago.edu/articles/2006/2/contents.html`.

2 Robert Beals. Quantum computation of Fourier transforms over symmetric groups. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, STOC'97, pages 48–53, New York, NY, USA, 1997. ACM. `doi:10.1145/258533.258548`.

3 Thomas Beth, Dieter Jungnickel, and Hanfried Lenz. *Design Theory*, volume I. Cambridge University Press, 2nd edition, 1999.

4 Thomas Beth, Dieter Jungnickel, and Hanfried Lenz. *Design Theory*, volume II. Cambridge University Press, 2nd edition, 1999.

5 Dan Boneh and Richard Lipton. Quantum cryptanalysis of hidden linear functions. In *Advances in Cryptology — CRYPT0' 95*, volume 963 of *Lecture Notes in Computer Science*, pages 424–437. Springer, 1995. `doi:10.1007/3-540-44750-4_34`.

6 Gilles Brassard and Peter Høyer. An exact polynomial–time algorithm for Simon's problem. In *Proceedings of Fifth Israeli Symposium on Theory of Computing and Systems*, pages 12–33. ISTCS, IEEE Computer Society Press, 1997. arXiv quant–ph/9704027.

7 Andrew M. Childs, Robin Kothari, Maris Ozols, and Martin Roetteler. Easy and hard functions for the Boolean hidden shift problem. In *8th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2013, May 21-23, 2013, Guelph, Canada*, pages 50–79, 2013.

8 Andrew M. Childs, Leonard J. Schulman, and Umesh V. Vazirani. Quantum algorithms for hidden nonlinear structures. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 395–404, 2007.

9 Andrew M. Childs and Wim van Dam. Quantum algorithms for algebraic problems. *Rev. Mod. Phys.*, 82:1–52, Jan 2010. arXiv: 0812.0380. `doi:10.1103/RevModPhys.82.1`.

10 Andrew M. Childs and Pawel Wocjan. On the quantum hardness of solving isomorphism problems as nonabelian hidden shift problems. *Quantum Information and Computation*, 7(5):504–521, Jul 2007. quant-ph/0510185. URL: `http://www.rintonpress.com/journals/qiconline.html#v7n56`.

11 John F. Dillon. A survey of bent functions. *The NSA technical journal*, pages 191–215, 1972.

12 Kirsten Eisenträger, Sean Hallgren, Alexei Kitaev, and Fang Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 – June 03, 2014*, pages 293–302, 2014.

13 Mark Ettinger and Peter Høyer. A quantum observable for the graph isomorphism problem. quant-ph/9901029, 1999.

14 Mark Ettinger and Peter Høyer. On quantum algorithms for noncommutative hidden subgroups. *Advances in Applied Mathematics*, 25(3):239–251, 2000. quant-ph/9807029. `doi:10.1006/aama.2000.0699`.

15 Katalin Friedl, Gábor Ivanyos, Frédéric Magniez, Miklos Santha, and Pranab Sen. Hidden translation and orbit coset in quantum computing. In *Proceedings of the 35fth Annual ACM Symposium on Theory of Computing (STOC'03)*, pages 1–9. ACM, 2002. quant-ph/0211091. `doi:10.1145/780542.780544`.

16 Dmitry Gavinsky, Martin Roetteler, and Jérémie Roland. Quantum algorithm for the boolean hidden shift problem. In *Computing and Combinatorics*, volume 6842 of *Lecture Notes in Computer Science*, pages 158–167. Springer Berlin / Heidelberg, 2011. arXiv: 1103.3017. `doi:10.1007/978-3-642-22685-4_14`.

17 Mirmojtaba Gharibi. Reduction from non-injective hidden shift problem to injective hidden shift problem. *Quantum Information and Computation*, 13(3&4):212–230, 2013.

18 Sean Hallgren. Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem. *J. ACM*, 54(1):4:1–4:19, Mar 2007. `doi:10.1145/1206035.1206039`.

19 Peter Høyer. Efficient quantum transforms. quant-ph/9702028, 1997.

20 Bertram Huppert. *Endliche Gruppen I*. Springer, 1967.

21 Gábor Ivanyos. On solving systems of random linear disequations. *Quantum Information and Computation*, 8(6&7):579–594, 2008. arXiv: 0704.2988. URL: `http://www.rintonpress.com/journals/qiconline.html#v8n67`.

22 Richard Jozsa. Quantum algorithms and the Fourier transform. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1969):323–337, 1998. quant-ph/9707033. `doi:10.1098/rspa.1998.0163`.

23 Richard Jozsa. Quantum factoring, discrete logarithms, and the hidden subgroup problem. *Computing in Science Engineering*, 3(2):34–43, Mar/Apr 2001. quant-ph/0012084. `doi:10.1109/5992.909000`.

24 Richard Jozsa. Quantum computation in algebraic number theory: Hallgren's efficient quantum algorithm for solving Pell's equation. *Annals of Physics*, 306(2):241–279, 2003. quant-ph/0302134. `doi:10.1016/S0003-4916(03)00067-8`.

25 Phillip Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing*. Oxford University Press, 2007.

26 Alexei Yu. Kitaev. Quantum measurements and the abelian stabilizer problem, 1995. quant-ph/9511026.

27 Alexei Yu. Kitaev, Alexander Shen, and Mikhail N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. American Mathematical Society, 2002.

28 Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005. quant-ph/0302112. `doi:10.1137/S0097539703436345`.

29 Greg Kuperberg. Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem. arXiv: 1112.3333, 2011. URL: `http://arxiv.org/abs/1112.3333`.

30 Eric S. Ladner. *Symmetric Designs: An Algebraic Approach*, volume 74 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, 1983.

31 Chris Lomont. The hidden subgroup problem – review and open problems. quant-ph/0411037, 2004.

32 F. J. MacWilliams and N. J. A. Sloane. *The theory of error–correcting codes*. North–Holland, Amsterdam, 1977.

33 Cristopher Moore, Daniel Rockmore, Alexander Russell, and Leonard J. Schulman. The power of strong Fourier sampling: Quantum algorithms for affine groups and hidden shifts. *SIAM J. Comput.*, 37(3):938–958, Jun 2007. quant-ph/0503095. `doi:10.1137/S0097539705447177`.

34 Michele Mosca and Artur Ekert. The hidden subgroup problem and eigenvalue estimation on a quantum computer. In *Quantum Computing and Quantum Communications*, volume 1509 of *Lecture Notes in Computer Science*, pages 174–188. Springer, 1999. quant-ph/9903071. `doi:10.1007/3-540-49208-9_15`.

35 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.

36 Maris Ozols, Martin Roetteler, and Jérémie Roland. Quantum rejection sampling. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS'12, pages 290–308, New York, NY, USA, 2012. ACM. arXiv: 1103.2774.

37 Josef Pieprzyk, Thomas Hardjono, and Jennifer Seberry. *Fundamentals of computer security*. Springer, 2003.

38 Alexander Pott, Vijay Kumar, Tor Helleseth, and Dieter Jungnickel, editors. *Difference sets, sequences, and their correlations*, volume 542 of *NATO Science Series*. Kluwer, 1998.

39 Oded Regev. New lattice-based cryptographic constructions. *J. ACM*, 51(6):899–942, 2004.

40 Oded Regev. Quantum computation and lattice problems. *SIAM Journal on Computing*, 33(3):738–760, 2004.

41 Oded Regev. A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space. quant-ph/0406151, 2004.

42 Martin Roetteler. Quantum algorithms to solve the hidden shift problem for quadratics and for functions of large Gowers norm. In *Proceedings of the 34st International Symposium on Mathematical Foundations of Computer Science (MFCS'09)*, volume 5734 of *Lecture Notes in Computer Science*, pages 663–674. Springer, 2009. arXiv: 0911.4724. `doi:10.1007/978-3-642-03816-7_56`.

43 Martin Roetteler. Quantum algorithms for highly non-linear Boolean functions. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'10)*, pages 448–457, 2010. arXiv: 0811.3208. URL: `http://arxiv.org/abs/0811.3208`.

44 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. Preliminary version in FOCS 1994. `doi:10.1137/S0097539795293172`.

45 Douglas R. Stinson. *Combinatorial Designs: Constructions and Analysis*. Springer, 2003.

46 Richard J. Turyn. Character sums and difference sets. *Pacific Journal of Mathematics*, 15(1):319–346, 1965.

47 Wim van Dam, Sean Hallgren, and Lawrence Ip. Quantum algorithms for some hidden shift problems. *SIAM Journal on Computing*, 36(3):763–778, 2006. quant-ph/0211140. `doi:10.1137/S009753970343141X`.

48 Wim van Dam and Gadiel Seroussi. Quantum algorithms for estimating Gauss sums. quant-ph/0207131, 2002.