Fast Quantum Fourier Transforms for a Class of Non-abelian Groups

Markus Püschel
pueschel@ira.uka.de

Martin Rötteler[1]
roettele@ira.uka.de

Thomas Beth
EISS_Office@ira.uka.de

Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe, Germany

Abstract

An algorithm is presented allowing the construction of fast Fourier transforms for any solvable group on a classical computer. The special structure of the recursion formula being the core of this algorithm makes it a good starting point to obtain systematically fast Fourier transforms for solvable groups on a quantum computer. The inherent structure of the Hilbert space imposed by the qubit architecture suggests to consider groups of order 2^n first (where n is the number of qubits). As an example, fast quantum Fourier transforms for all 4 classes of non-abelian 2-groups with cyclic normal subgroup of index 2 are explicitly constructed in terms of quantum circuits. The (quantum) complexity of the Fourier transform for these groups of size 2^n is $O(n^2)$ in all cases.

[1] supported by DFG grant GRK 209/3-98

1 Introduction

Quantum algorithms are a recent subject and of possibly central importance in physics and computer science. It has been shown that there are problems on which a putative quantum computer could outperform every classical computer. A striking example is Shor's factoring algorithm (see [22]).

Here we adress a problem used as a subroutine in almost all known quantum algorithms: The quantum Fourier transform (QFT) and its generalization to arbitrary finite groups.

In classical computation there exist elaborate methods for the construction of Fourier transforms (see Beth [3], [4], Clausen [5], [6], and Diaconis/Rockmore [9]) so it is highly interesting to adapt and modify these methods to get a quantum algorithm with a much better performance (with respect to the common quantum complexity model) as on a classical computer.

First attempts in this direction have been proposed by Beals [2] and Høyer [12]. In this paper we present an algebraic approach using representation theory which can be seen as a first step towards the realization of a large class of QFTs on a quantum computer.

2 Generalized Fourier Transforms

Fourier transforms for finite groups are an interesting and well studied topic for classical computers. We refer to [3], [6], [15], [19] as representatives for a vast number of publications. The reader not familiar with the standard notation concerning group representations should refer to these publications or to standard references as [8] or [21].

For the convenience of the reader we first recall the definition of the generalized Fourier transforms for a finite group G and explain the representation theoretical point of view we are going to take.

Each isomorphism

$$\Phi : \mathbb{C}G \longrightarrow \bigoplus_{i=1}^{k} \mathbb{C}^{d_i \times d_i}$$

between the group algebra of G and the Wedderburn components is called a *Fourier transform* for the group G. A particular isomorphism is fixed by picking a system $\rho_1, \dots, \rho_k$ of representatives of irreducible representations of G and defining Φ as the linear extension of the mapping $g \mapsto \bigoplus_{i=1}^{k} \rho_i(g),\ g \in G$ (of course $\deg(\rho_i) = d_i$). Any "fast" algorithm for the evaluation of Φ is called a *fast Fourier transform* for G.

In order to obtain a matrix representation for the linear mapping Φ one usually fixes *natural* bases L in $\mathbb{C}G$ and L' in $\bigoplus_{i=1}^{k} \mathbb{C}^{d_i \times d_i}$. This is done by choosing an ordering $L = (g_1, \dots, g_{|G|})$ on the elements of G and an ordering L' on the elementary matrices $E_{k,l}$ (1 at position (k,l), 0 else) which correspond to the coefficients of the irreducible representations appearing in the Wedderburn decomposition. The Fourier transform Φ then is represented by a matrix $M_{L,L'} \in \mathbb{C}^{|G|\times|G|}$. Since $M_{L,L'}$ is a base change between two orthonormal bases (with respect to the standard hermitean scalar product on $\mathbb{C}G$) it must be unitary.

The matrix $M_{L,L'}$ can also be characterized by another property. Let ϕ be the regular representation obtained by right multiplication of G on L. Conjugating ϕ with $M_{L,L'}$ yields (up to a permutation matrix P) a direct sum of irreducible representations with the property that equivalent irreducibles are equal, i.e.

$$\phi^{M_{L,L'}} = (\rho_1 \oplus \ldots \oplus \rho_k)^P \quad \text{fulfilling} \quad \rho_i \cong \rho_j \Rightarrow \rho_i = \rho_j.$$

On the other hand every matrix with this property corresponds to a Fourier transform with respect to natural bases.

As an example let $G = Z_n = \langle x \mid x^n = 1\rangle$ be the cyclic group of order n with irreducible representations $\rho_i = (x \mapsto \omega_n^i)$, $i = 0 \ldots n-1$, where ω_n denotes a primitive n-th root of unity. With respect to the natural bases $L = (x^i \mid i = 0 \ldots n-1)$ and $L' = (E_{1,1}, \ldots, E_{n,n})$ the matrix $M_{L,L'} = \frac{1}{\sqrt{n}}[\omega_n^{i\cdot j} \mid i,j = 0 \ldots n-1] = \mathrm{DFT}_n$ is the discrete Fourier transform well-known from signal processing.

We will refer to the notion of a fast Fourier transform as a fast algorithm for the multiplication with $M_{L,L'}$. Of course, the term “fast” depends on the chosen complexity model. Since we are primarily interested in the realization of a fast Fourier transform on a quantum computer (QFT) we first have to define the measure of complexity on this architecture.

3 The Complexity Model

In this paper we think of a quantum computer to consist of a quantum register which in turn consists of n qubits each of which provides a 2-dimensional Hilbert space. Thus the possible operations this computer can perform are given by the unitary group $\mathcal{U}(2^n)$.

To study the complexity of unitary operators on n-qubit quantum systems we introduce the following two types of building blocks:

- Local unitary operations on the qubit i are of the form

  $$U^{(i)} := \mathbf{1}_{2^{i-1}} \otimes U \otimes \mathbf{1}_{2^{n-i}},$$

  where U is an element of the unitary group $\mathcal{U}(2)$ of 2×2-matrices.

- The controlled NOT gate (also called measurement gate) between the qubits i (control) and j (target) is defined by

  $$\mathrm{CNOT}^{(i,j)} := \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & & 1 \\ & & 1 & \end{pmatrix}$$

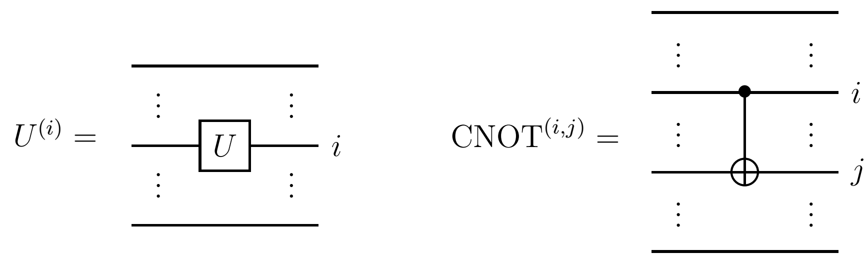

Figure 1: Elementary quantum gates

when restricted to the tensor component of the Hilbert space which is spanned by the qubits i and j. Note that the controlled not is nothing but an XOR on the resp. components.

In the graphical notation using quantum wires these transforms are written as shown in figure 1. The lines represent the qubits and the least significant bit is the lowest. We assume that these so-called elementary quantum gates can be performed with cost $O(1)$.

These two types of gates suffice to generate all unitary transformations, which is the content of the following theorem from [1].

Theorem 3.1 *The set* $\mathcal{G} = \{U^{(i)}, \mathrm{CNOT}^{(i,j)} \mid U \in \mathcal{U}(2),\ i,j = 1 \dots n,\ i \neq j\}$ *is a generating set for the unitary group* $\mathcal{U}(2^n)$.

This means that for each $U \in \mathcal{U}(2^n)$ there is a word $w_1 w_2 \dots w_k$ (where $w_i \in \mathcal{G}$ for $i = 1 \dots k$ is an elementary gate) such that U factorizes as $U = w_1 w_2 \dots w_k$.

In general only exponential upper bounds for the minimal length occuring in factorizations have been proved (see [1]) but there are many interesting classes of unitary matrices in $\mathcal{U}(2^n)$ affording only polylogarithmic word length, which means, that the minimal length k is asymptotically $O(p(n))$ where p is a polynomial.

In the following we give examples of some particular unitary transforms admitting short factorizations which will be useful in the rest of the paper.

- The symmetric group S_n is embedded in $\mathcal{U}(2^n)$ by the natural operation of S_n on the tensor components (qubits). Let $\tau \in S_n$ and Π_τ the corresponding permutation matrix on 2^n points. Then Π_τ has a $O(n)$ factorization as shown in [18]. As an example in figure 2 the permutation $(1,3,2)$ of the qubits (which corresponds to the permutation $(2,5,3)(4,6,7)$ on the register) is factored according to $(1,3,2) = (1,2)(2,3)$.

- Following the notation of [1] we denote a k-times controlled U by $\Lambda_k(U)$. Lemma 7.2 and lemma 7.5 in [1] show that for $U \in \mathcal{U}(2)$ the gate $\Lambda_{n-1}(U)$ can be realized with gate complexity $O(n^2)$ and $\Lambda_k(U)$ with $O(n)$ for $k < n-1$ on n qubits.

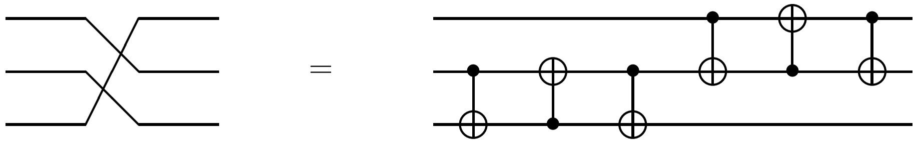

Figure 2: Factorization $(1, 3, 2) = (1, 2)(2, 3)$

- The Fourier transform DFT_{2^n} can be performed in $O(n^2)$ elementary operations on a quantum computer (see [22], [7]).

- Let $P_n \in S_{2^n}$ be the cyclic shift permutation which acts on the states of the quantum register as $x \mapsto x + 1 \bmod 2^n$. Obviously the corresponding permutation matrix is the 2^n-cycle

$$P_n = \begin{pmatrix} 0 & 1 & & & & \\ & 0 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & & 0 & 1 \\ 1 & & & & & 0 \end{pmatrix}.$$

  A factorization of P_n as a product of multiple controlled NOTs as shown in figure 3 needs $O(n^2)$ basic operations[1].

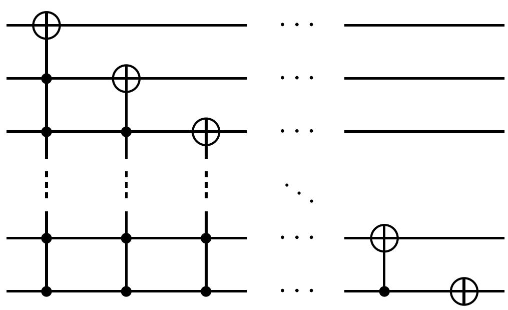

Figure 3: Quantum circuit for the 2^n-cycle P_n

- Let $U \in \mathcal{U}(2^n)$. The cost for a (single) controlled U is settled by the following lemma.

Lemma 3.2 *If $U \in \mathcal{U}(2^n)$ can be realized in $O(p(n))$ elementary operations then $\Lambda_1(U) \in \mathcal{U}(2^{n+1})$ can also be realized in $O(p(n))$ basic operations.*

[1] This quantum circuit, similar to the classical carry-look-ahead logic, has been found by Markus Grassl following discussions with Amir Fijany in May 1998

Proof: First we assume without loss of generality that U is written in elementary gates. Therefore we have to show that a doubly controlled NOT and a single controlled $U \in \mathcal{U}(2)$ can be realized with a constant increase of length. This follows from [1]. $\square$

4 Creating Fast Fourier Transforms

In section 2 we have explained that calculating a Fourier transform for a group G is the same as decomposing a regular representation ϕ of G with a matrix A into irreducibles ρ_i up to a permutation P such that equivalent irreducible summands are equal, i.e.

$$\phi^A = A^{-1} \cdot \phi \cdot A = (\rho_1 \oplus \ldots \oplus \rho_k)^P \quad \text{fulfilling} \quad \rho_i \cong \rho_j \Rightarrow \rho_i = \rho_j.$$

A "fast" Fourier transform (on a classical computer) is given by a factorization of A into a product of sparse (w.r.t. the architectural complexity measure) matrices.

In this section we present (without proof) a number of theorems and lemmata yielding an algorithm to calculate fast Fourier transforms for any solvable group G (on a classical computer). The same algorithm serves as a good starting point to obtain quantum Fourier transforms if we assume the "quantum wires" to possess a suitable number of states.

The statements in this section all are taken from the first chapter of [19] where decomposition matrices and constructive representation theory in general is investigated. There the objective is the construction of decomposition matrices for monomial representations which can be viewed as a generalization of Fourier transforms.

The following theorem provides the crucial formula needed to obtain a fast Fourier transform of G by decomposing a regular representation stepwise along a composition series of G. The formula has been known (see [3]) to yield fast Fourier transforms for solvable groups on a classical computer (counting additions and multiplications). Their tensor structure, however, also fits well to the special architecture of a quantum computer. The general constructive form of the following theorem as presented is due to [19] where furthermore an improved recursion formula for the classical architecture can be found.

We use the following convention for the induction of a representation ϕ of $H \leq G$ with transversal (i.e. a system of representatives for the right cosets) $T = (t_1, \ldots, t_n)$:

$$(\phi \uparrow_T G)(x) = \left[\dot{\phi}(t_i x t_j^{-1}) \mid i, j = 1 \ldots n\right],$$

where $\dot{\phi}(y) = \phi(y)$ for $y \in H$ and the all-zero matrix else. A regular representation ϕ is given by an induction $\phi = (1_E \uparrow_T G)$ where 1_E denotes the trivial representation of the trivial subgroup $E \leq G$.

Theorem 4.1 *Let $N \trianglelefteq G$ a normal subgroup of prime index p with (cyclic) transversal $T = (t^0, t^1, \ldots, t^{(p-1)})$ and ϕ a representation of degree d of N which has an extension $\overline{\phi}$ to G (see figure 4). Suppose that A is matrix decomposing ϕ into irreducibles, i.e. $\phi^A = \rho = \rho_1 \oplus \ldots \oplus \rho_k$ and that $\overline{\rho}$ is an extension of ρ to G. Then*

$$B = (\mathbf{1}_p \otimes A) \cdot D \cdot (\mathrm{DFT}_p \otimes \mathbf{1}_d), \quad \textit{where} \quad D = \bigoplus_{i=0}^{p-1} \overline{\rho}(t)^i,$$

is a decomposition matrix for $\phi \uparrow_T G$, more precisely

$$(\phi \uparrow_T G)^B = \bigoplus_{i=0}^{p-1} \lambda_i \cdot \overline{\rho},$$

where $\lambda_i : \ t \mapsto \omega_p^i$, $i = 0 \ldots p-1$, are the p 1-dimensional representations of G arising from the factor group G/N.

$$\begin{tikzcd}
G & \overline{\phi} & & \overline{\rho} \\
N \arrow[u] & \phi \arrow[u, "\mathrm{ext}"] \arrow[rr, "A", "\mathrm{dec}"'] & & \rho \arrow[u, "\mathrm{ext}"]
\end{tikzcd}$$

Figure 4: Situation in theorem 4.1

In the case of an abelian group G the formula yields exactly the well-known Cooley-Tukey decomposition, hence D can be viewed as a generalized Twiddle factor. In the case of G being a direct product $G \cong N \times G/N$ the Twiddle matrix D vanishes. Since we want to apply theorem 4.1 to a regular representation we need the following lemma.

Lemma 4.2 *Let $N \trianglelefteq G$ a normal subgroup of prime index p and ϕ any regular representation of N. Then ϕ (and hence all of its conjugates) has an extension $\overline{\phi}$ to G. Furthermore $\phi \cong \phi^t$ for all $t \in G$. ($\phi^t : \ x \mapsto \phi(txt^{-1})$ is called the inner conjugate of ϕ by t).*

In order to obtain an algorithm from theorem 4.1 we are faced with two problems. The first is the calculation of the Twiddle matrix D which is essentially the problem of extending ρ to $\overline{\rho}$ and evaluating it at t. Suppose we are given A and ρ which is a direct sum of irreducibles, $\rho = \rho_1 \oplus \ldots \oplus \rho_k$, with equivalent summands being equal. Because of lemma 4.2 G/N operates on the irreducibles ρ_i via inner conjugation (explained in the lemma above). According to Clifford's Theorem (see e.g. [6], pp. 88) exactly one of the following two cases applies to each summand ρ_i: Either $\rho_i \cong \rho_i^t$ and ρ_i can be extended to G or $\rho_i \ncong \rho_i^t$ and $\rho_i \uparrow_T G$ is irreducible. In the first case the extension may be calculated by Minkwitz' formula (see [17]), in the latter case the direct sum $\rho_i \oplus \rho_i^t \oplus \ldots \oplus \rho_i^{t^{(p-1)}}$ can be extended to G by $\rho_i \uparrow_T G$. We do not state Minkwitz' formula here since we will not need it in the special cases treated in section 5.

The second problem arises from the fact that the decomposition $\bigoplus_{i=0}^{p-1} \lambda_i \cdot \overline{\rho}$ in theorem 4.1 does not satisfy the property of equivalent summands being equal. This can be achieved using the following lemma.

Lemma 4.3 *Let $N \trianglelefteq G$ a normal subgroup of prime index p with transversal $T = (t^0, t^1, \ldots, t^{(p-1)})$. Suppose that ρ is an irreducible representation of degree d of N satisfying $\rho \ncong \rho^t$ and $\lambda_i : \ t \mapsto \omega_p^i$ is an irreducible representation of G arising from G/N. Then*

$$(\lambda_i \cdot (\rho \uparrow_T G))^{D \otimes \mathbf{1}_d} = \rho \uparrow_T G, \quad D = \mathrm{diag}(1, \omega_p, \ldots, \omega_p^{(p-1)})^i.$$

Now we are ready to formulate the algorithm which constructs a fast Fourier transform for G from a fast Fourier transform of a normal subgroup of prime index.

Algorithm 4.4 *Let $N \trianglelefteq G$ a normal subgroup of prime index p with transversal $T = (t^0, t^1, \ldots, t^{(p-1)})$. Suppose that ϕ is a regular representation of N with decomposition matrix A:*

$$\phi^A = \rho_1 \oplus \ldots \oplus \rho_k \quad \textit{fulfilling} \quad \rho_i \cong \rho_j \Rightarrow \rho_i = \rho_j.$$

A decomposition matrix B for the regular representation $\phi \uparrow_T G$ can be obtained as follows.

1. *Determine a permutation matrix P rearranging the ρ_i, $i = 1 \ldots k$, such that the extendable ρ_i (i.e. those satisfying $\rho_i = \rho_i^t$) come first followed by the others ordered into sequences of length p equivalent to $\rho_i, \rho_i^t, \ldots, \rho_i^{t^{(p-1)}}$. (Note: These sequences we need to equal $\rho_i, \rho_i^t, \ldots, \rho_i^{t^{(p-1)}}$ which is established in the next step).*

2. *Calculate a matrix M which is the identity on the extendables and conjugates the sequences of length p to make them equal to $\rho_i, \rho_i^t, \ldots, \rho_i^{t^{(p-1)}}$.*

3. *Note that $A \cdot P \cdot M$ is a decomposition matrix for ϕ, too, and let $\rho = \phi^{A \cdot P \cdot M}$. Extend ρ to G summandwise. For the extendable summands use Minkwitz' formula, the sequences $\rho_i, \rho_i^t, \ldots, \rho_i^{t^{(p-1)}}$ can be extended by $\rho_i \uparrow_T G$ as stated above.*

4. *Evaluate $\overline{\rho}$ at t and build $D = \bigoplus_{i=0}^{p-1} \overline{\rho}(t)^i$.*

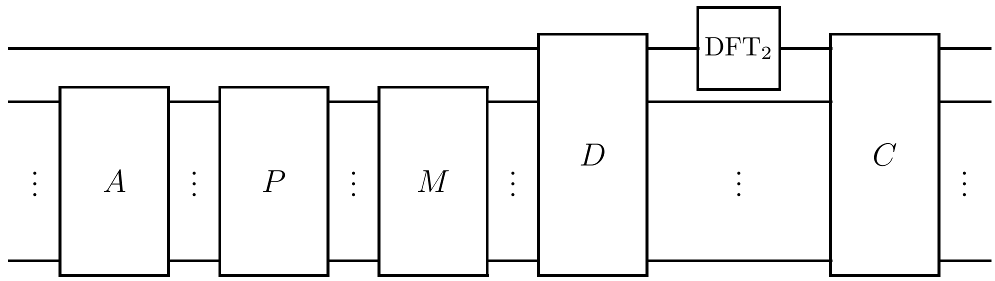

Figure 5: Coarse Quantum circuit visualizing algorithm 4.4

5. *Construct a blockdiagonal matrix C with lemma 4.3 conjugating $\bigoplus_{i=0}^{p-1} \lambda_i \cdot \overline{\rho}$ such that equivalent irreducibles are equal. C is the identity on the extended summands.*

Then

$$B = (\mathbf{1}_p \otimes A \cdot P \cdot M) \cdot D \cdot (\mathrm{DFT}_p \otimes \mathbf{1}_{|N|}) \cdot C \tag{1}$$

is a decomposition matrix for $\phi \uparrow_T G$. $\square$

It is obviously possible to construct fast Fourier transforms on a classical computer for any solvable group by recursive use of this algorithm. Note that the consideration of T-adapted representations (see [6]) here is unnecessary: The irreducibles are constructed along with the decomposition matrices.

Since we restrict ourselves to the case of a quantum computer consisting of qubits, i.e. two-level systems, we apply algorithm 4.4 to obtain QFTs for 2-groups (size is a 2-power). In this case the two tensor products occuring in (1) fit very well to yield a coarse factorization as shown in figure 5. The remaining problem, however, is the realization of the matrices A, P, M, D, C in terms of elementary building blocks as presented in section 3. At present this realization remains a creative process which might be performed by hand if an arbitrary class of groups is given. In section 5 we will apply algorithm 4.4 to a class of non-abelian 2-groups.

5 Generating QFTs for a class of 2-groups

In the case of G being an abelian 2-group the realization of a fast quantum Fourier transform has been settled by [14]. Clearly, this case is covered by the method presented here (see the remarks following theorem 4.1). In this section we will apply algorithm 4.4 to the class of non-abelian 2-groups containing a cyclic normal subgroup of index 2. Fast quantum Fourier transforms for these groups have already been constructed by Høyer in [12].

According to [13], p. 90/91 there are for $n \geq 3$ exactly four isomorphism types of non-abelian groups of order 2^{n+1} affording a cyclic normal subgroup of order 2^n:

(i) The dihedral group $D_{2^{n+1}} = \langle x, y \mid x^{2^n} = y^2 = 1,\ x^y = x^{-1} \rangle$.

(ii) The quaternion group $Q_{2^{n+1}} = \langle x, y \mid x^{2^n} = y^4 = 1,\ x^y = x^{-1} \rangle$.

(iii) The group $QP_{2^{n+1}} = \langle x, y \mid x^{2^n} = y^2 = 1,\ x^y = x^{2^{n-1}+1} \rangle$.

(iv) The quasidihedral group $QD_{2^{n+1}} = \langle x, y \mid x^{2^n} = y^2 = 1,\ x^y = x^{2^{n-1}-1} \rangle$.

Observe that the extensions (i), (iii), and (iv) of the cyclic subgroup $Z_{2^n} = \langle x \rangle$ split, i. e. the groups have the structure of a semidirect product of Z_{2^n} by Z_2. The three isomorphism types correspond to the three different embeddings of $Z_2 = \langle y \rangle$ into $(Z_{2^n})^\times \cong Z_2 \times Z_{2^{n-2}}$.

5.1 QFT for the dihedral groups $\mathbf{D_{2^{n+1}}}$

In this section we construct a QFT for the dihedral groups $D_{2^{n+1}}$ step by step according to algorithm 4.4 and explicitly state the occuring quantum circuits.

Let $G = D_{2^{n+1}} = \langle x, y \mid x^{2^n} = y^2 = 1,\ x^y = x^{-1} \rangle$ with normal subgroup $N = \langle x \rangle \trianglelefteq G$ of index 2 and transversal $T = (1, y)$. We consider the regular representation $\phi = (1_E \uparrow_S N) \uparrow_T G$ of G with $S = (1, x, \dots, x^{2^n-1})$. Obviously the regular representation $(1_E \uparrow_S N)$ of N is decomposed by $A = \mathrm{DFT}_{2^n}$ into $\rho_0 \oplus \dots \oplus \rho_{2^n-1}$ where $\rho_i = (x \mapsto \omega_{2^n}^i)$. Now we are ready to apply algorithm 4.4 to obtain a decomposition matrix B for ϕ. For convenience we denote ω_{2^n} simply as ω and the Hadamard matrix as

$$H = \mathrm{DFT}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

1. Since $\rho_i^y(x) = \rho_i(yxy^{-1}) = \rho_i(x^{-1}) = \rho_{2^n-i}(x)$ we see that there are exactly two extendable ρ_i namely for $i = 0, 2^{n-1}$. The sequences of inner conjugates are given by ρ_i, ρ_{2^n-i}, $i \neq 0, 2^{n-1}$. We need a permutation P reordering the ρ_i as

$$\underbrace{\rho_0, \rho_{2^{n-1}}}_{\text{extendables}},\ \underbrace{\rho_1, \rho_{2^n-1}, \dots, \rho_i, \rho_{2^n-i}, \dots, \rho_{2^{n-1}-1}, \rho_{2^{n-1}+1}}_{\text{pairs of inner conjugates}}.$$

   This can be accomplished by the circuit given in figure 6 since the n-cycle on the qubits which is performed first yields a decimation by two on the indices, i. e. the indices $0, \dots, 2^{n-1} - 1$ have found their correct position. The only thing which remains to do is to perform the operation $x \mapsto -x$ on the odd positions. This can be done by an inversion of all (odd) bits followed by a $x \mapsto x + 1$ shift P_{n-1} on the odd states of the register.

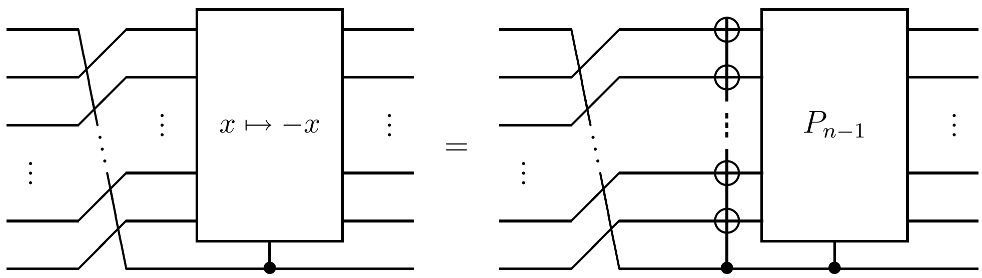

Figure 6: Ordering the irreducibles of $Z_{2^n} \trianglelefteq D_{2^{n+1}}$

2. M can be omitted since all the ρ_i are of degree 1.

3. Let $\phi^{A \cdot P} = \rho$. We extend ρ summandwise to $\overline{\rho}$:

   - $\rho_0 = 1_N$ can be extended by 1_G.
   - $\rho_{2^{n-1}}$ can be extended through $\overline{\rho}_{2^{n-1}}(y) = 1$.
   - The sequences $\rho_i \oplus \rho_{2^n - i}$, $i \neq 0, 2^{n-1}$ can be extended by $\rho_i \uparrow_T G$:

$$\rho_i \uparrow_T G : x \mapsto \begin{pmatrix} \omega^i & 0 \\ 0 & \omega^{-i} \end{pmatrix}, \; y \mapsto \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

4. Evaluation of $\overline{\rho}$ at the transversal T yields the Twiddle matrix

$$\begin{aligned} D &= \overline{\rho}(1) \oplus \overline{\rho}(y) \\ &= \mathbf{1}_{2^n} \oplus \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & & 1 & & & \\ & & 1 & & & & \\ & & & & \ddots & & \\ & & & & & & 1 \\ & & & & & 1 & \end{pmatrix}. \end{aligned}$$

   D is realized by the quantum circuit given in figure 7.

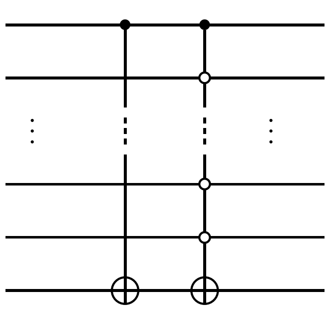

Figure 7: Twiddle matrix for $D_{2^{n+1}}$

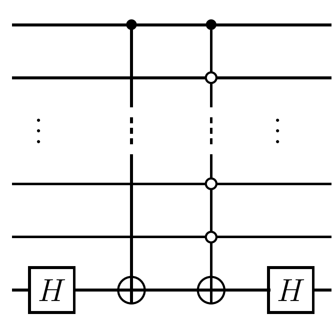

Figure 8: Equalizing inductions

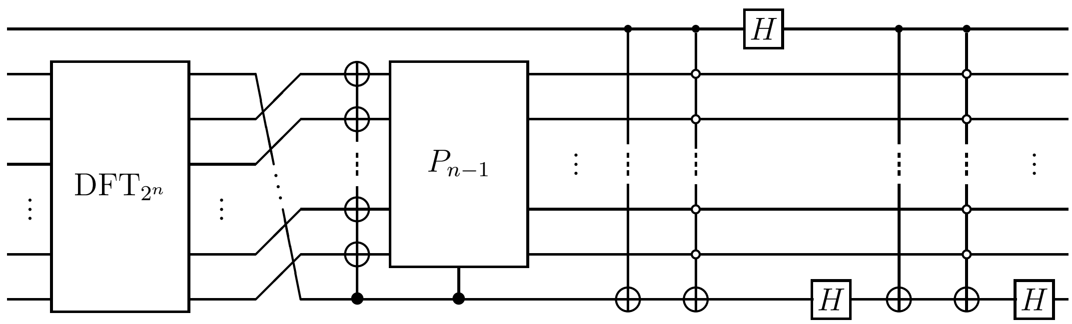

Figure 9: Complete QFT circuit for the dihedral group $D_{2^{n+1}}$

5. According to lemma 4.3 the matrix C has the following diagonal form:

$$C = \mathbf{1}_{2^n} \oplus \mathrm{diag}(1, 1, \underbrace{1, -1, \ldots, 1, -1}_{2^{n-1}-1 \text{ pairs}}),$$

which is realized by the quantum circuit given in figure 8.

Summarizing we obtain that

$$B = (\mathbf{1}_p \otimes A \cdot P \cdot M) \cdot D \cdot (\mathrm{DFT}_p \otimes \mathbf{1}_{|N|}) \cdot C$$

is a decomposition matrix for ϕ and a fast quantum Fourier transform for G. The whole circuit is shown in figure 9.

5.2 QFT for the groups $\mathbf{Q_{2^{n+1}}}$, $\mathbf{QP_{2^{n+1}}}$, and $\mathbf{QD_{2^{n+1}}}$

In the following we give the circuits for the groups $Q_{2^{n+1}}$, $QP_{2^{n+1}}$, and $QD_{2^{n+1}}$. In all cases we have $\langle x \rangle = N \trianglelefteq G$ so that algorithm 4.4 has to be performed only once for the last step. For the sake of brevity we will state only those parts of the circuit which differ from the dihedral group. Some of the essential properties of the groups are summarized in tabular 1. We use the same notation as in the last section.

Group	Inner conjugates of Z_{2^n}	No. of 1-dim irreducibles	No. of 2-dim irreducibles
$D_{2^{n+1}}$	$\rho_i, \quad \rho_{2^n-i}$	4	$2^{n-1}-1$
$Q_{2^{n+1}}$	$\rho_i, \quad \rho_{2^n-i}$	4	$2^{n-1}-1$
$QP_{2^{n+1}}$	$\rho_i, \quad \rho_{i(2^{n-1}+1) \bmod 2^n}$	2^n	2^{n-2}
$QD_{2^{n+1}}$	$\rho_i, \quad \rho_{i(2^{n-1}-1) \bmod 2^n}$	4	$2^{n-1}-1$

Table 1: A class of non-abelian 2-groups

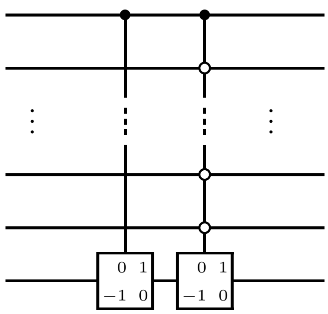

Figure 10: Twiddle matrix for $Q_{2^{n+1}}$

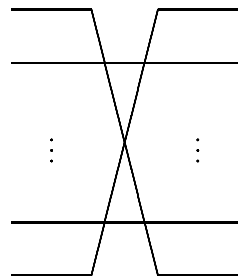

Figure 11: Permutation for $QP_{2^{n+1}}$

- $Q_{2^{n+1}}$: The irreducibles ρ_i extend or induce in the same way as in the dihedral case. Hence the QFT only differs in the Twiddle matrix D since for a not extendable ρ_i we have

$$(\rho_i \uparrow_T G)(y) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

Thus the Twiddle matrix D is given by

$$D = \mathbf{1}_{2^n} \oplus \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & & 1 & & & \\ & & -1 & & & & \\ & & & & \ddots & & \\ & & & & & & 1 \\ & & & & & -1 & \end{pmatrix}$$

and can be realized by the circuit in figure 10.

- $QP_{2^{n+1}}$: To determine which ρ_i are extendable we use $\rho_i^y(x) = \rho_i(yxy^{-1}) = \rho_i(x^{2^{n-1}+1})$. Hence

$$\rho_i = \rho_i^y \Leftrightarrow \omega^i = \omega^{i\cdot(2^{n-1}+1)} \Leftrightarrow \omega^{i\cdot 2^{n-1}} = 1 \Leftrightarrow 2 \mid i$$

and there are exactly 2^{n-1} extendable ρ_i. The reordering permutation P has the easy form shown in figure 11, and the matrix D is given by

$$D = \mathbf{1}_{2^n} \oplus \mathbf{1}_{2^{n-1}} \oplus \begin{pmatrix} & 1 & & & \\ 1 & & & & \\ & & \ddots & & \\ & & & & 1 \\ & & & 1 & \end{pmatrix}$$

which is simply a doubly controlled not as visualized in figure 12.

The matrix C then is given by figure 13.

- $QD_{2^{n+1}}$: Here we have $\rho_i^y(x) = \rho_i(x^{2^{n-1}-1})$ and

$$\rho_i = \rho_i^y \Leftrightarrow \omega^i = \omega^{i\cdot(2^{n-1}-1)} \Leftrightarrow \omega^{i\cdot(2^{n-1}-2)} = 1 \Leftrightarrow i = 0, 2^{n-1}.$$

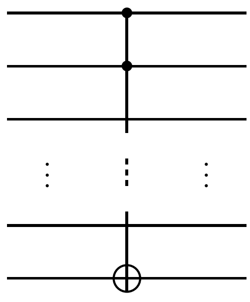

Figure 12: Twiddle matrix for $QP_{2^{n+1}}$

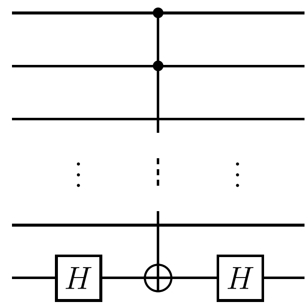

Figure 13: Equalizing for $QP_{2^{n+1}}$

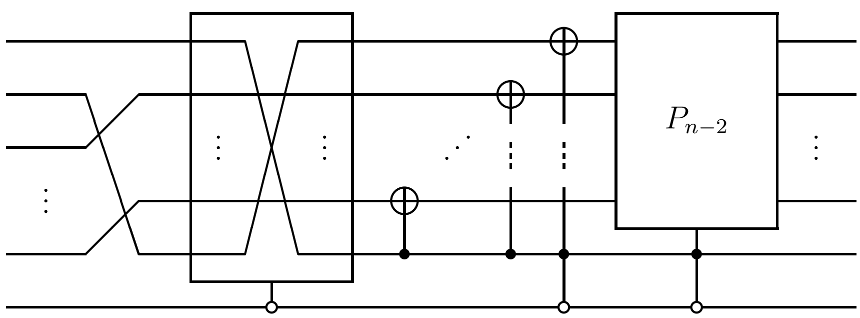

Figure 14: The permutation for the $QD_{2^{n+1}}$

> Thus everything is the same as in the dihedral case beside the ordering permutation P which takes the more complicate form shown in figure 14.

Investigation of the quantum circuits yields the following theorem.

Theorem 5.1 *The Fourier transforms for the groups $G = D_{2^n},\ Q_{2^n},\ QP_{2^n}$, and QD_{2^n} can be performed on a quantum computer in $O(\log^2 |G|)$ elementary operations.*

Proof: We can treat the four series uniformly, since the Fourier transforms all have the same decomposition pattern. First, in all cases a Fourier transform for the normal subgroup $Z_{2^{n-1}}$ is performed with cost of $O(n^2)$ basic operations. The reordering permutation P, the Twiddle matrix D, and the equalizing matrix C cost $O(n^2)$ in case of D_{2^n}, Q_{2^n}, and QD_{2^n} due to lemma 3.2 and example 3. For QP_{2^n} we need only $O(1)$ operations for P, D, and C. $\square$

All presented Fourier transforms have been implemented by the authors in the language GAP [20] using the package AREP [11] which will be available soon as a GAP share package.

6 Conclusions and Outlook

A constructive algorithm has been presented allowing to attack the problem of constructing fast Fourier transforms for 2-groups G on a quantum computer built up from qubits. For a certain class of non-abelian 2-groups the algorithm has been successfully applied. All the QFTs created are of computational complexity $O(\log^2 |G|)$ like in the case of the cyclic group Z_{2^n}. The main problem imposed by the implementation of certain permutation and block diagonal matrices has been solved efficiently.

Using the recursion formula from theorem 4.1 it should be possible to construct QFTs for other classes of groups as well as to realize certain signal transforms on a quantum computer by means of symmetry-based decomposition (see [19], [10], [16]).

We are indebted to Markus Grassl for helpful comments and discussions. Part of this work was presented and completed during the 1998 Elsag-Bailey – I.S.I. Foundation research meeting on quantum computation.

References

[1] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457–3467, November 1995. LANL preprint quant-ph/9503016.

[2] R. Beals. Quantum computation of Fourier transforms over the symmetric groups. In *Proceedings 29th Annual ACM Symposium on Theory of Computing*, El Paso, Texas, May 1997.

[3] T. Beth. *Methoden der Schnellen Fouriertransformation*. Teubner, 1984.

[4] T. Beth. On the computational complexity of the general discrete Fourier transform. *Theoretical Computer Science*, 51:331–339, 1987.

[5] M. Clausen. Fast generalized Fourier transforms. *Theoretical Computer Science*, 67:55–63, 1989.

[6] M. Clausen and U. Baum. *Fast Fourier Transforms*. BI-Verlag, 1993.

[7] D. Coppersmith. An Approximate Fourier Transform Useful for Quantum Factoring. Technical Report RC 19642, IBM Research Division, Yorktown Heights NY, December 1994.

[8] W.C. Curtis and I. Reiner. *Methods of Representation Theory*, volume 1. Interscience, 1981.

[9] P. Diaconis and D. Rockmore. Efficient computation of the Fourier transform on finite groups . *Amer. Math. Soc.*, 3(2):297–332, 1990.

[10] S. Egner. *Zur Algorithmischen Zerlegungstheorie Linearer Transformationen mit Symmetrie*. Dissertation, Universität Karlsruhe, 1997.

[11] S. Egner and M. Püschel. *AREP – A Package for Constructive Representation Theory*, 1998.

[12] P. Høyer. Efficient Quantum Transforms. LANL preprint quant-ph/9702028, February 1997.

[13] B. Huppert. *Endliche Gruppen*, volume I. Springer, 1983.

[14] A. Yu. Kitaev. Quantum Measurements and the Abelian Stabilizer Problem. LANL preprint quant-ph/9511026, November 1995.

[15] D. Maslen and D. Rockmore. Generalized FFTs – a survey of some recent results. *Proceedings of IMACS Workshop in Groups and Computation*, 28:182–238, 1995.

[16] T. Minkwitz. *Algorithmensynthese für lineare Systeme mit Symmetrie*. Dissertation, Universität Karlsruhe, 1993.

[17] T. Minkwitz. Extension of Irreducible Representations. *AAECC*, 7:391–399, 1996.

[18] C. Moore and M. Nilsson. Some notes on parallel quantum computation. LANL preprint quant-ph/9804034, April 1998.

[19] M. Püschel. *Konstruktive Darstellungstheorie und Algorithmengenerierung*. Dissertation, Universität Karlsruhe, 1998.

[20] Schönert, M. et al. *GAP — Groups, Algorithms and Programming*. Lehrstuhl D für Mathematik, Rheinisch Westfälische Technische Hochschule, Aachen, Germany, fifth edition, 1995.

[21] J. P. Serre. *Linear Representations of Finite Groups*. Springer, 1977.

[22] P. W. Shor. Algorithms for Quantum Computation: Discrete Logarithm and Factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. Institute of Electrical and Electronic Engineers Computer Society Press, November 1994.