On the Irresistible Efficiency of Signal Processing Methods in Quantum Computing

Andreas Klappenecker*[1,2], Martin Rötteler†[2]

[1]Department of Computer Science, Texas A&M University,
College Station, TX 77843-3112, USA

[2]Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe,‡
Am Fasanengarten 5, D-76 128 Karlsruhe, Germany

November 9, 2000

Abstract

We show that many well-known signal transforms allow highly efficient realizations on a quantum computer. We explain some elementary quantum circuits and review the construction of the Quantum Fourier Transform. We derive quantum circuits for the Discrete Cosine and Sine Transforms, and for the Discrete Hartley transform. We show that at most $O(\log^2 N)$ elementary quantum gates are necessary to implement any of those transforms for input sequences of length N.

§1 Introduction

Quantum computers have the potential to solve certain problems at much higher speed than any classical computer. Some evidence for this statement is given by Shor's algorithm to factor integers in polynomial time on a quantum computer. A crucial part of Shor's algorithm depends on the discrete Fourier transform. The time complexity of the quantum Fourier transform is polylogarithmic in the length of the input signal. It is natural to ask whether other signal transforms allow for similar speed-ups.

We briefly recall some properties of quantum circuits and construct the quantum Fourier transform. The main part of this paper is concerned with

*e-mail: `klappi@ira.uka.de`

†e-mail: `roettele@ira.uka.de`

‡research group Quantum Computing, Professor Thomas Beth

the construction of quantum circuits for the discrete Cosine transforms, for the discrete Sine transforms, and for the discrete Hartley transform.

§2 Elementary Quantum Circuits

The quantum computation will be done in the state space of n two-level quantum systems, which is given by a 2^n-dimensional complex vector space. The basis vectors are denoted by $|x\rangle$ where x is a binary string of length n. The basic unit of quantum information processing is a quantum bit or shortly qubit, which represents the state of a two-level quantum system.

A quantum gate on n qubits is an element in the group of unitary matrices $\mathcal{U}(2^n)$. There are two types of gates that are considered elementary: the XOR gates (also known as controlled NOTs) and the single qubit operations.

The controlled NOT gate operates on two qubits. It negates the target qubit if and only if the control qubit is 1. Suppose that $x = b_n \dots b_1$ is a string of n bits, then

$$\mathrm{CNOT}_{c,t}\,|x\rangle = \begin{cases} |y\rangle & \text{if} \quad b_c = 1 \\ |x\rangle & \text{if} \quad b_c = 0 \end{cases}$$

where y is the bitstring obtained from x by negating the bit b_t.

A single qubit gate acts on a target qubit at position t by a local unitary transformation

$$\mathbf{1}_{2^{n-t}} \otimes U \otimes \mathbf{1}_{2^{t-1}}, \qquad U \in \mathcal{U}(2).$$

It will be convenient to describe the quantum circuits with a graphical notation put forward by Feynman. The circuits are read from left to right like a musical score. The qubits are represented by lines, with the most significant bit at the top. Figure 1 shows the graphical notation of the elementary gates.

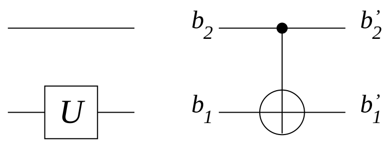

Figure 1: The Feynman notation for the single qubit gate $(\mathbf{1}_2 \otimes U)$ and for a controlled NOT operation $|b_2'\, b_1'\rangle = |b_2\, b_2 \oplus b_1\rangle$.

A multiply controlled NOT is defined as follows. Let C be a subset of $[1..n]$ not containing the target t. Then

$$\mathrm{CNOT}_{C,t}\,|x\rangle = \begin{cases} |y\rangle & \text{if} \quad b_c = 1 \text{ for all } c \in C \\ |x\rangle & \text{otherwise} \end{cases}$$

where $|y\rangle$ is defined as above. Several controlled NOT operations in a sequence allow us to implement the operation $P_n |x\rangle = |x + 1 \bmod 2^n\rangle$. Note that $O(n)$ elementary gates are sufficient to realize a multiply controlled NOT operation on n qubits, assuming that an additional scratch qubit is available. Therefore, at most $O(n^2)$ elementary gates are necessary to implement the shift operation P_n.

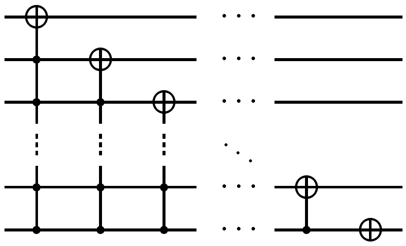

Figure 2: Shift

The state of two qubits can be exchanged with the help of three controlled NOT operations:

$$\mathrm{SWAP}_{k,h} = \mathrm{CNOT}_{h,k}\mathrm{CNOT}_{k,h}\mathrm{CNOT}_{h,k}.$$

It follows that any permutation of the n quantum wires can be realized with at most $O(n)$ elementary quantum gates.

A more detailed discussion of properties of quantum gates can be found in [1]. We will discuss the construction of the discrete Fourier transform in the next section. In particular, we will show the classical dataflow diagram and the corresponding quantum gates to further illustrate the graphical notation.

§3 Quantum Fourier Transform

The discrete Fourier transform of length $N = 2^n$ is defined by

$$F_N = \frac{1}{\sqrt{N}} \left[\omega^{jk} \right]_{j,k=0,\dots,N-1},$$

where $\omega = \exp(2\pi i/N)$ with $i^2 = -1$. Recall the recursion step used in the Cooley-Tukey decomposition:

$$F_N = \Upsilon_N (\mathbf{1}_2 \otimes F_{N/2}) \begin{pmatrix} \mathbf{1}_{N/2} & \\ & T_{N/2} \end{pmatrix} (F_2 \otimes \mathbf{1}_{N/2}) \qquad (1)$$

where $T_{N/2} := \mathrm{diag}(1, \omega, \omega^2, \ldots, \omega^{N/2-1})$ denotes the matrix of twiddle factors, and Υ_N denotes the permutation given by $\Upsilon_N |xb\rangle = |bx\rangle$ with x an $n-1$-bit integer, and b a single bit.

We note that the implementation of F_2 is a local operation on a single quantum bit. The recursion suggest four different parts of the implementation of Fourier transforms of larger length. The matrix $(F_2 \otimes \mathbf{1}_{N/2})$ is a single Hadamard operation on the most significant qubit. We would like to emphasize that this *single* quantum operation corresponds to a full butterfly diagram.

The implementation of the twiddle matrix is more complex. Notice that $T_{N/2}$ can be written as a tensor product of diagonal matrices $L_j = \mathrm{diag}(1, \omega^{2^{j-1}})$ in the form

$$T_{N/2} = L_{n-1} \otimes \ldots \otimes L_2 \otimes L_1.$$

Thus, $\mathbf{1}_{N/2} \oplus T_{N/2}$ can be realized by controlled phase shift operations. Figure 3 shows the implementation of the two operations discussed so far.

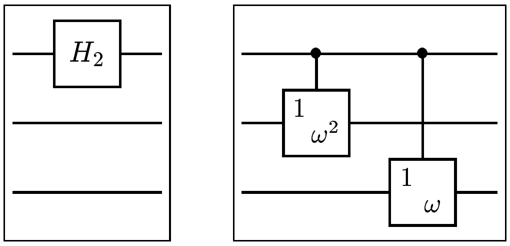

Figure 3: For length N=8, only three qubits are necessary. The circuit on the left implements $(F_2 \otimes \mathbf{1}_{N/2})$ and the other realizes the twiddle matrix $\mathbf{1}_4 \oplus \mathrm{diag}(1, \omega, \omega^2, \omega^3)$.

It remains to discuss the other two operations in (1). The operation $(\mathbf{1}_2 \otimes F_{N/2})$ means that an implementation of the discrete Fourier transform of length $N/2$ is used on the least significant $(n-1)$ bits. The operation Υ_N is a permutation of quantum wires. We can combine all the permutations

$$\Upsilon_N(\mathbf{1}_2 \otimes \Upsilon_{N/2}) \ldots (\mathbf{1}_{N-2} \otimes \Upsilon_4)$$

into a single permutation of quantum wires. The resulting permutation is the bit reversal, see Figure 4. The classical and quantum implementation of the discrete Fourier transform of length 8 are compared in Figure 5. We observe that the butterfly diagrams find simple realizations but the twiddle matrices require more elementary quantum gate operations.
The complexity of the quantum implementation can be estimated as follows. If we denote by $R(N)$ the number of gates necessary to implement the DFT

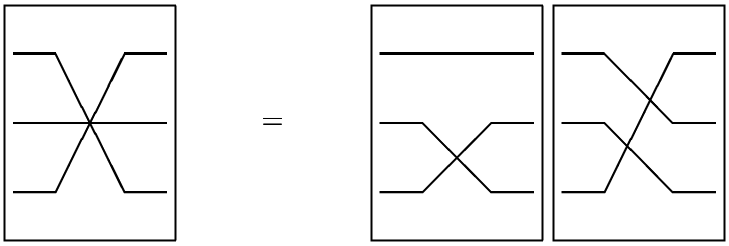

Figure 4: The bit reversal permutation is given by $\mathbf{1}_2 \otimes \Upsilon_4$ followed by Υ_8.

of length $N = 2^n$ on a quantum computer, then equation (1) implies the recurrence relation

$$R(N) = R(N/2) + O(\log N)$$

which leads to the estimate $R(N) = O(\log^2 N)$.

Shor's factoring algorithm relies on the quantum Fourier transform in a fundamental way. For more details on Fourier transforms and their generalizations to nonabelian groups, see [4, 5].

§4 Quantum Cosine and Sine Transforms

We derive quantum circuits for discrete Cosine and Sine transforms in this section. The main idea is simple: reuse the circuits for the discrete Fourier transform.

The discrete Cosine and Sine transforms are divided into various families. We follow [6] and define the following four versions of *discrete Cosine transforms*:

$$\begin{aligned}
C_N^{\mathrm{I}} &:= \left(\frac{2}{N}\right)^{1/2} \left[k_i \cos\frac{ij\pi}{N}\right]_{i,j=0,\dots,N} \\
C_N^{\mathrm{II}} &:= \left(\frac{2}{N}\right)^{1/2} \left[k_i \cos\frac{i(j+1/2)\pi}{N}\right]_{i,j=0,\dots,N-1} \\
C_N^{\mathrm{III}} &:= \left(\frac{2}{N}\right)^{1/2} \left[k_i \cos\frac{(i+1/2)j\pi}{N}\right]_{i,j=0,\dots,N-1} \\
C_N^{\mathrm{IV}} &:= \left(\frac{2}{N}\right)^{1/2} \left[k_i \cos\frac{(i+1/2)(j+1/2)\pi}{N}\right]_{i,j=0,\dots,N-1}
\end{aligned}$$

where $k_i := 1$ for $i = 1, \dots, N-1$ and $k_0 := 1/\sqrt{2}$. The numbers k_i ensure that the transforms are orthogonal. The *discrete Sine transforms* are defined

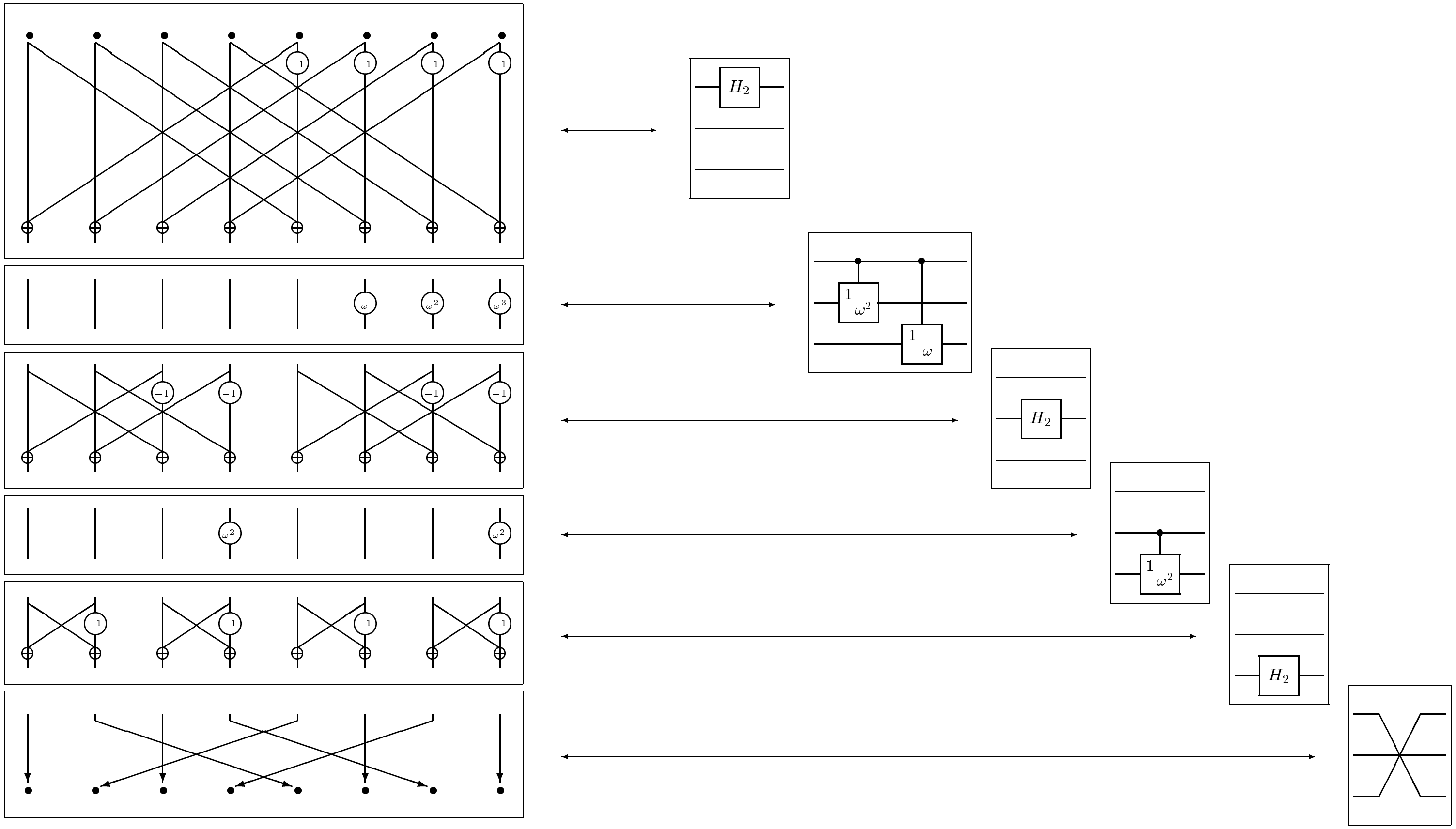

Figure 5: Dataflow graph of the DFT of length 8 and the corresponding quantum circuits.

by

$$
\begin{aligned}
S_N^{\mathrm{I}} &:= \left(\frac{2}{N}\right)^{1/2}\left[k_i \sin\frac{ij\pi}{N}\right]_{i,j=1,\ldots,N-1}\\
S_N^{\mathrm{II}} &:= \left(\frac{2}{N}\right)^{1/2}\left[k_i \sin\frac{i(j+1/2)\pi}{N}\right]_{i,j=0,\ldots,N-1}\\
S_N^{\mathrm{III}} &:= \left(\frac{2}{N}\right)^{1/2}\left[k_i \sin\frac{(i+1/2)j\pi}{N}\right]_{i,j=0,\ldots,N-1}\\
S_N^{\mathrm{IV}} &:= \left(\frac{2}{N}\right)^{1/2}\left[k_i \sin\frac{(i+1/2)(j+1/2)\pi}{N}\right]_{i,j=0,\ldots,N-1}
\end{aligned}
$$

where the constants k_i are defined as above. Notice that C_N^{III} (resp. S_N^{III}) is the transpose of C_N^{II} (resp. S_N^{II}), hence it suffices to derive circuits for the type II transforms.

It is well-known that the trigonometric transforms can be obtained by conjugating the discrete Fourier transform F_{2N} by certain sparse matrices. We refer the reader to Wickerhauser [7] for more details on the decompositions.

DCT$_{\mathrm{I}}$ and DST$_{\mathrm{I}}$. We derive the circuits for the discrete Sine and Cosine transforms of type I all at once. Indeed, the DST$_{\mathrm{I}}$ and DCT$_{\mathrm{I}}$ can be recovered from the DFT by a base change

$$T_N^\dagger \cdot F_{2N} \cdot T_N = C_N^{\mathrm{I}} \oplus iS_N^{\mathrm{I}}, \tag{2}$$

where

$$
T_N = \begin{pmatrix}
1 & & & & & & & \\
& \frac{1}{\sqrt{2}} & & & & \frac{i}{\sqrt{2}} & & \\
& & \ddots & & & & \ddots & \\
& & & \frac{1}{\sqrt{2}} & & & & \frac{i}{\sqrt{2}} \\
& & & & 1 & & & \\
& & & \frac{1}{\sqrt{2}} & & & & -\frac{i}{\sqrt{2}} \\
& & \iddots & & & & \iddots & \\
& \frac{1}{\sqrt{2}} & & & & -\frac{i}{\sqrt{2}} & &
\end{pmatrix}.
$$

It is straightforward to check that (2) holds, see Theorem 3.10 in [7]. Since we already know efficient quantum circuits for the DFT, it remains to find an efficient implementation of the base change matrix T_N.

It will be convenient to denote the basis vectors of $\mathbf{C}^{2^{n+1}}$ by $|bx\rangle$, where b is a single bit and x is an n-bit number. The two's complement of an n-bit unsigned integer x is denoted by x', that is, $x' = 2^n - x$. The action of T_N

can be described by

$$\begin{aligned} T_N\,|\mathbf{0}\mathbf{0}\rangle &= |\mathbf{0}\mathbf{0}\rangle & T_N\,|0x\rangle &= \tfrac{1}{\sqrt{2}}\,|0x\rangle + \tfrac{1}{\sqrt{2}}\,|1x'\rangle \\ T_N\,|1\mathbf{0}\rangle &= |1\mathbf{0}\rangle & T_N\,|1x\rangle &= \tfrac{i}{\sqrt{2}}\,|0x\rangle - \tfrac{i}{\sqrt{2}}\,|1x'\rangle \end{aligned}$$

for all integers x in the range $1 \leq x < 2^n$. Ignoring the two's complement in T_N, we can define an operator D by

$$\begin{aligned} D\,|\mathbf{0}\mathbf{0}\rangle &= |\mathbf{0}\mathbf{0}\rangle & D\,|0x\rangle &= \tfrac{1}{\sqrt{2}}\,|0x\rangle + \tfrac{1}{\sqrt{2}}\,|1x\rangle \\ D\,|1\mathbf{0}\rangle &= |1\mathbf{0}\rangle & D\,|1x\rangle &= \tfrac{i}{\sqrt{2}}\,|0x\rangle - \tfrac{i}{\sqrt{2}}\,|1x\rangle \end{aligned}$$

for all integers x in the range $1 \leq x < 2^n$. This operator is essentially block diagonal and easy to implement by a single qubit operation, followed by a correction. Indeed, define the matrix B by $B = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & i \\ 1 & -i \end{pmatrix}$, then Figure 5 gives an implementation of the operator D.

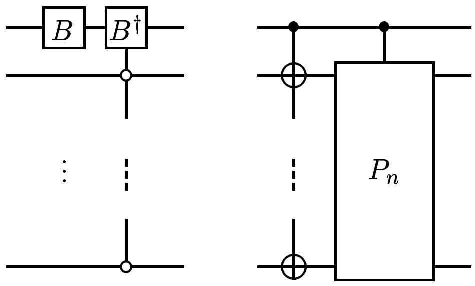

Figure 5: Circuits realizing the block matrix D and the permutation π.

Define π to be the permutation given by a two's complement conditioned on the most significant bit $\pi\,|0x\rangle = |0x\rangle$ and $\pi\,|1x\rangle = |1x'\rangle$ for all n-bit integers x. It is clear that $T_N = \pi D$. The circuits for the permutation π is shown in Figure 5.

Theorem 1 *The discrete Cosine transform C_N^{I} and the discrete Sine transform S_N^{I} can be realized with at most $O(\log^2 N)$ elementary quantum gates; the quantum circuit for these transforms is shown in Figure 6.*

Proof. Let $N = 2^n$. We note that $O(\log^2 N)$ quantum gates are sufficient to realize the DFT of length $2N$. The permutation π can be implemented with at most $O(\log^2 N)$ elementary gates. At most $O(\log N)$ quantum gates are needed to realize the operator D. This shows that the $\mathrm{DCT}_{\mathrm{I}}$ and the $\mathrm{DST}_{\mathrm{I}}$ can be realized with at most $O(\log^2 N)$ quantum gates. The preceding discussion shows that Figure 6 realizes the $\mathrm{DCT}_{\mathrm{I}}$ and $\mathrm{DST}_{\mathrm{I}}$. $\square$

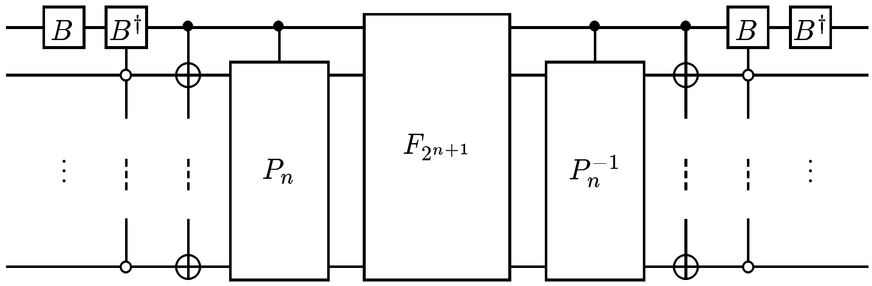

Figure 6: Complete quantum circuit for the $\mathrm{DCT_I}$

$\mathbf{DCT_{IV}}$ and $\mathbf{DST_{IV}}$. The trigonometric transforms of type IV are derived from the DFT by

$$e^{\pi i/4N} R_N^t \cdot F_{2N} \cdot R_N = C_N^{\mathrm{IV}} \oplus (-i) S_N^{\mathrm{IV}}. \tag{3}$$

Here R_N denotes the matrix

$$R_N = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & & & & -i & & & \\ & \omega & & & & -i\omega & & \\ & & \ddots & & & & \ddots & \\ & & & \omega^{N-1} & & & & -i\omega^{N-1} \\ & & & \overline{\omega}^N & & & & 1 \\ & & ⋰ & & & & ⋰ & \\ & \overline{\omega}^2 & & & & i\overline{\omega}^2 & & \\ \overline{\omega} & & & & i\overline{\omega} & & & \end{pmatrix}$$

with ω the primitive $4N$-th root of unity $\omega = \exp(2\pi i/4N)$. Equation (3) is a consequence of Theorem 3.19 in [7] obtained by complex conjugation.

Theorem 2 *The discrete Cosine transform C_N^{IV} and the discrete Sine transform S_N^{IV} can be realized with at most $O(\log^2 N)$ elementary quantum gates; the quantum circuit for these transforms is shown in Figure 7.*

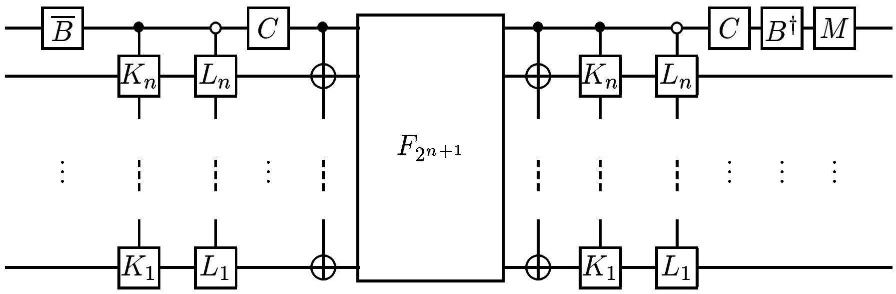

Figure 7: Complete quantum circuit for $\mathrm{DCT_{IV}}$

Proof. It remains to show that there exists an efficient quantum circuit for the matrix R_N in equation (3). A factorization of R_N can be obtained as follows. Denote by $\overline{x}$ the one's complement of an n-bit integer x. We define a permutation matrix π_1 by $\pi_1 |0x\rangle = |0x\rangle$ and $\pi_1 |1x\rangle = |1\overline{x}\rangle$ for all integers x in the range of $0 \leq x < 2^n$. Denote by D_1 the diagonal matrix

$$D_1 = \mathrm{diag}(1, \omega, \dots, \omega^{N-1}, \overline{\omega}^N, \dots, \overline{\omega}^2, \overline{\omega}).$$

Then R_N can be factored as

$$R_N \quad = \quad \pi_1 \cdot D_1 \cdot \left(\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ 1 & i \end{pmatrix} \otimes \mathbf{1}_N \right) = \pi_1 \cdot D_1 \cdot (\overline{B} \otimes \mathbf{1}_N).$$

Note that $\overline{B} \otimes \mathbf{1}_N$ is a single qubit operation, and π_1 can be realized by controlled not operations. The implementation of the diagonal matrix D_1 is more interesting. Note that the diagonal matrices of increasing (decreasing) powers can be written by tensor products

$$\begin{aligned} \Delta_1 = \mathrm{diag}(1, \omega, \dots, \omega^{N-1}) &= L_n \otimes \cdots \otimes L_2 \otimes L_1 \\ \Delta_2 = \mathrm{diag}(\overline{\omega}^{N-1}, \dots, \overline{\omega}, 1) &= K_n \otimes \cdots \otimes K_2 \otimes K_1 \end{aligned}$$

where $L_j = \mathrm{diag}(1, \omega^{2^{j-1}})$ and $K_j = \mathrm{diag}(\overline{\omega}^{2^{j-1}}, 1)$. Therefore, it is possible to write D_1 in the form $D_1 = (C \otimes \mathbf{1}_N) \cdot (\Delta_1 \oplus \Delta_2)$ with $C = \mathrm{diag}(1, \overline{\omega})$. The circuit for the diagonal matrix D_1 is shown in Figure 8.

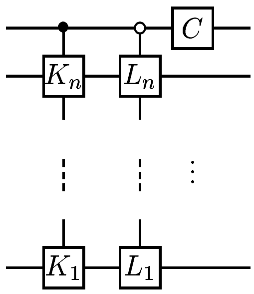

Figure 8: Quantumcircuit for the diagonal matrix D_1.

The complete quantum circuit for the $\mathrm{DCT}_{\mathrm{IV}}$ is shown in Figure 7. Note that the last three single qubit gates C, $B^\dagger$, and $M = \mathrm{diag}(e^{\pi i/4N}, e^{\pi i/4N})$ can be combined into a single gate $MB^\dagger C$. $\square$

$\mathbf{DCT_{II}}$ and $\mathbf{DST_{II}}$. The implementation of the trigonometric transforms of type II follows a similar pattern. Both transforms can be recovered from the DFT of length 2N after multiplication with certain sparse matrices, cf. Theorem 3.13 in [7]:

$$U_N^\dagger \cdot F_{2N} \cdot V_N = C_N^{\mathrm{II}} \oplus (-i) S_N^{\mathrm{II}}, \tag{4}$$

where

$$V_N = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & & & 1 & & \\ & \ddots & & & \ddots & \\ & & 1 & & & 1 \\ & & 1 & & & -1 \\ & ⋰ & & & ⋰ & \\ 1 & & & -1 & & \end{pmatrix}$$

and

$$U_N = \begin{pmatrix} 1 & & & & 0 & & & \\ & \frac{\overline{\omega}}{\sqrt{2}} & & & -\frac{i\overline{\omega}}{\sqrt{2}} & & & \\ & & \ddots & & & \ddots & & \\ & & & \frac{\overline{\omega}^{N-1}}{\sqrt{2}} & & & -\frac{i\overline{\omega}^{N-1}}{\sqrt{2}} & 0 \\ & & & 0 & & & & -1 \\ & & & \frac{\omega^{N-1}}{\sqrt{2}} & & & \frac{i\omega^{N-1}}{\sqrt{2}} & \\ & & ⋰ & & & ⋰ & & \\ 0 & \frac{\omega}{\sqrt{2}} & & & \frac{i\omega}{\sqrt{2}} & & & \end{pmatrix},$$

and ω denotes the $4N$-th primitive root of unity $\omega = \exp(2\pi i/4N)$, $i^2 = -1$.

Theorem 3 *The discrete Cosine transform C_N^{II} and the discrete Sine transform S_N^{II} can be realized with at most $O(\log^2 N)$ elementary quantum gates; the quantum circuit for these transforms is shown in Figure 9.*

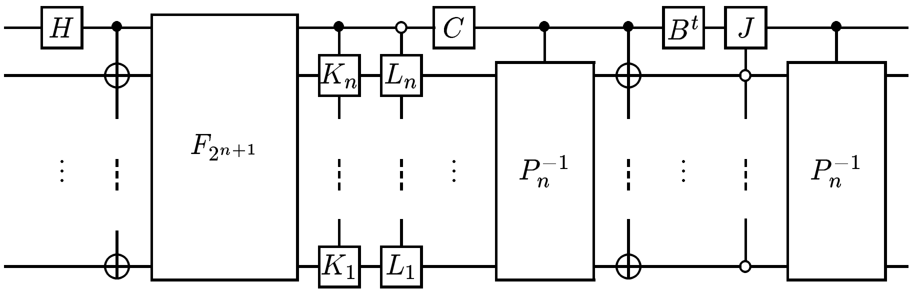

Figure 9: Complete quantum circuit for $\mathrm{DCT}_{\mathrm{II}}$

Proof. We need to derive efficient quantum circuits for the matrices V_N and U_N in equation (4). The matrix V_N has a fairly simple decomposition in terms of quantum circuits.

Lemma 4 $V_N = \pi_1(H \otimes \mathbf{1}_N)$.

Proof. It is clear that the Hadamard transform on the most significant bit $H \otimes I_N$ is – up to a permutation of rows – equivalent to V_N. The appropriate permutation of rows has been introduced in the previous section, namely $\pi_1 |0x\rangle = |1x\rangle$ and $\pi_1 |1x\rangle = |1\overline{x}\rangle$ for all $0 \le x < 2^n$. We can conclude that $V_N = \pi_1(H \otimes \mathbf{1}_N)$ as desired. $\square$

The decomposition of U_N is more elaborate. Notice that

$$\begin{array}{ll} U_N |\mathbf{00}\rangle = |\mathbf{00}\rangle & U_N |0x\rangle = \dfrac{\overline{\omega}^x}{\sqrt{2}} |0x\rangle + \dfrac{\omega^x}{\sqrt{2}} |1x'\rangle \\ U_N |\mathbf{11}\rangle = (-1) |\mathbf{10}\rangle & U_N |1y\rangle = -\dfrac{i\overline{\omega}^{y+1}}{\sqrt{2}} |0\,(y+1 \bmod 2^n)\rangle + \dfrac{i\omega^{y+1}}{\sqrt{2}} |1\overline{y}\rangle \end{array}$$

for all integers x in the range $1 \le x < 2^n$ and all integers y in $0 \le y < 2^n - 1$. Here $\mathbf{0}$ and $\mathbf{1}$ denote the n-bit integers 0 and $2^n - 1$ respectively.

Define D_0 by $D_0 |\mathbf{10}\rangle = i |\mathbf{10}\rangle$ and $D_0 |x\rangle = |x\rangle$ otherwise. We define a permutation π_2 by $\pi_2 |0x\rangle = |0x\rangle$ and $\pi_2 |1x\rangle = |1(x+1 \bmod 2^n)\rangle$ for all integers x in $0 \le x < 2^n$.

Lemma 5 $U_N = D_1^\dagger \overline{T}_N D_0 \pi_2$.

Proof. Since $D_1^\dagger |0x\rangle = \overline{\omega}^x |0x\rangle$ and $D_1^\dagger |1x\rangle = \omega^{x'} |1x\rangle$, we obtain

$$\begin{aligned} D_1^\dagger \overline{T}_N |0x\rangle &= \frac{\overline{\omega}^x}{\sqrt{2}} |0x\rangle + \frac{\omega^x}{\sqrt{2}} |1x'\rangle \\ D_1^\dagger \overline{T}_N |1x\rangle &= -\frac{i\overline{\omega}^x}{\sqrt{2}} |0x\rangle + \frac{i\omega^x}{\sqrt{2}} |1x'\rangle \end{aligned}$$

We have $D_0\pi_2 |0x\rangle = |0x\rangle$, $D_0\pi_2 |1x\rangle = |1(x+1 \bmod 2^n)\rangle$ for all integers x in $0 \le x < 2^n - 1$, and $D_0\pi_2 |\mathbf{11}\rangle = i |\mathbf{10}\rangle$. We note that $(x+1 \bmod 2^n)' = \overline{x}$, whence combining $D_1\overline{T}_N$ with $D_0\pi_2$ shows the result. $\square$

Recall that $T_N = \pi D$. It follows that

$$U_N^\dagger = \pi_2^{-1}(\overline{D}_0 D^t)\pi^{-1} D_1.$$

The implementation of D_1 has been described in the section on the $\mathrm{DCT_{IV}}$, and the implementation of π (and hence π^{-1}) is contained in the section on the $\mathrm{DCT_I}$. The implementation of π_2^{-1} is also straightforward. It remains to find an implementation of $\overline{D}_0 D^t$. We observe that

$$\begin{array}{rclrcl} \overline{D}_0 D^t |\mathbf{00}\rangle &=& |\mathbf{00}\rangle & \overline{D}_0 D^t |0x\rangle &=& \frac{1}{\sqrt{2}} |0x\rangle + \frac{i}{\sqrt{2}} |1x\rangle \\ \overline{D}_0 D^t |\mathbf{10}\rangle &=& (-i) |\mathbf{10}\rangle & \overline{D}_0 D^t |1x\rangle &=& \frac{1}{\sqrt{2}} |0x\rangle - \frac{i}{\sqrt{2}} |1x\rangle \end{array}$$

This can be accomplished by the single bit operation $B^t \otimes \mathbf{1}_N$ followed by a multiply conditioned gate $J = \dfrac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$. The full circuit is shown in Figure 9. The statement about the complexity is clear. $\square$

§5 Quantum Hartley Transforms

The discrete Hartley transform of length $N \in \mathbf{N}$ is the real $N \times N$ matrix A_N defined by

$$A_N := \frac{1}{\sqrt{N}} \Big[\mathrm{cas}\Big(\frac{2\pi i j}{N} \Big) \Big]_{i,j=0,\ldots,N-1},$$

where the function cas : $\mathbf{R} \to \mathbf{R}$ is defined by $\mathrm{cas}(x) := \cos(x) + \sin(x)$, see [2, 3] for classical implementations. The property

$$A_N = \left(\frac{1-i}{2} \right) F_N + \left(\frac{1+i}{2} \right) F_N^3$$

is easily seen from the definition. We derive a quantum circuit implementing A_N with one auxiliary quantum bit.

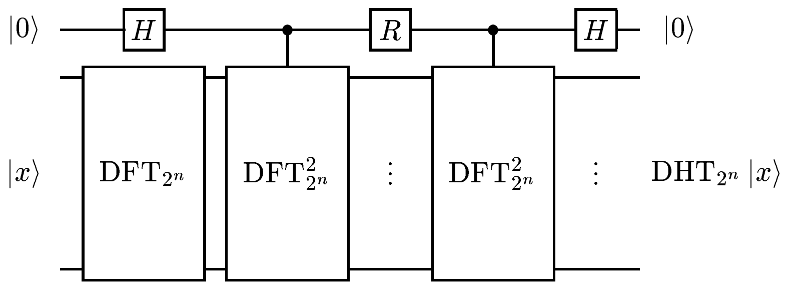

Figure 10: Circuit realising a quantum Hartley transform

Lemma 6 *The discrete Hartley transform can be factorized in the form shown in Figure 10. Here R is the unitary circulant matrix*

$$R := \frac{1}{2} \begin{pmatrix} 1-i & 1+i \\ 1+i & 1-i \end{pmatrix}$$

and H denotes the Hadamard transform.

Proof. Let $\check{F}_N$ be the transformation which effects a DFT conditioned to the first bit, i. e., written in terms of matrices we have $\check{F}_N = \mathbf{1}_N \oplus F_N$. We now show that the given circuit computes the linear transformation $|0\rangle\,|x\rangle \mapsto |0\rangle\, A_N\,|x\rangle$ for all unit vectors $x \in \mathbf{C}^n$. Proceeding from left to right in the circuit given in Figure 10 we obtain

$$|0\rangle\,|x\rangle \quad \stackrel{H}{\longmapsto} \quad \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\,|x\rangle$$

$$
\begin{aligned}
&\xmapsto{F_N} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)F_N\,|x\rangle \\
&\xmapsto{\check{F}_N^2} \frac{1}{\sqrt{2}}\,|0\rangle\, F_N\,|x\rangle + \frac{1}{\sqrt{2}}\,|0\rangle\, F_N^3\,|x\rangle \\
&\xmapsto{R} \frac{1}{\sqrt{2}}\,|0\rangle \left(\frac{1}{2}(1-i)F_N + \frac{1}{2}(1+i)F_N^3\right)|x\rangle \\
&\qquad + \frac{1}{\sqrt{2}}\,|1\rangle \left(\frac{1}{2}(1+i)F_N + \frac{1}{2}(1-i)F_N^3\right)|x\rangle \\
&= \frac{1}{\sqrt{2}}\,|0\rangle\, A_N\,|x\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\, F_N^{-2}A_N\,|x\rangle \\
&\xmapsto{\check{F}_N^2} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)A_N\,|x\rangle \\
&\xmapsto{H} |0\rangle\, A_N\,|x\rangle
\end{aligned}
$$

as desired. □

Theorem 7 *The discrete Hartley transform A_N can be computed on a quantum computer using $O(\log^2 N)$ elementary operations if we allow one additional ancilla qubit.*

Proof. Recall that the discrete Fourier transform F_N can be implemented $O(\log^2 N)$ operations as shown in Section §3. The statement follows from Lemma 6 since all transformations given there require at most $O(\log^2 N)$ elementary operations. □

§6 Conclusions

We have shown that the discrete Cosine transforms, the discrete Sine transforms, and the discrete Hartley transforms have extremely efficient realizations on a quantum computer. All implementations illustrated an important design principle: the reusability of highly optimized quantum circuits. Apart from a few sparse matrices, we only needed the circuits for the discrete Fourier transform for the implementations. A key point is that an improvement of a basic circuit, like the DFT, immediately leads to more efficient quantum circuits for the DCT, DST, and DHT.

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.

[2] Th. Beth. Generating Fast Hartley Transforms - Another Application of the Algebraic Discrete Fourier Transform. In *Proc. URSI-ISSSE '89*, pages 688–692, 1989.

[3] Bracewell. *The Hartley Transform.* Cambridge Univ. Press, 1979.

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

[5] M. Püschel, M. Rötteler, and Th. Beth. Fast Quantum Fourier Transforms for a Class of non-abelian Groups. In *Proceedings Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC-13)*, volume 1719 of *Lecture Notes in Computer Science*, pages 148–159. Springer, 1999.

[6] K. R. Rao and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, and Applications.* Academic Press, 1990.

[7] V. Wickerhauser. *Adapted Wavelet Analysis from Theory to Software.* A.K. Peters, Wellesley, 1993.