Scaling and Better Approximating Quantum Fourier Transform by Higher Radices

Zeljko Zilic, *Member*, *IEEE*, and Katarzyna Radecka, *Member*, *IEEE*

Abstract—Quantum Fourier Transform (QFT) plays a principal role in the development of efficient quantum algorithms. Since the number of quantum bits that can currently be built is limited, while many quantum technologies are inherently three (or more) valued, we consider extending the reach of the realistic quantum systems by building a QFT over ternary quantum digits. Compared to traditional binary QFT, the $q\text{-valued}$ transform improves approximation properties and increases the state space by a factor of $(q/2)^n$. Further, we use nonbinary QFT derivation to generalize and improve the approximation bounds for QFT.

Index Terms—Fourier transform, quantum computing, multivalued logic circuits, multivariable systems, Walsh functions.

1 Introduction

Quantum computing [15] has so far been demonstrated as being capable of speeding up algorithms for problems such as factoring integers or database searching, as well as being useful in cryptographic applications. Several working quantum-computing systems have already been demonstrated in practice [8], [17]. With such a prospect, it is natural to seek new algorithmic solutions that exploit quantum mechanical principles.

Some of the major improvements of quantum computing over the classical algorithms are due to *Quantum Fourier Transform* (QFT) [12]. While classical Fast Fourier Transform (FFT) [4] decreases running time from $O(2^{2n})$ to $O(n2^n)$, QFT offers even more dramatic improvements. Although more restrictive in scope than classical FFT, it can be performed in time $O(n^2)$ or even $O(n)$ in some cases [12]. Direct benefits of QFT are obtained in the celebrated Shor's number factoring algorithm [16]. It was further observed that the derivation of general quantum operations in nonbinary quantum logic relies on QFT [9].

Currently, the main obstacle in quantum computing is posed by the inability to realize practical quantum systems operating on a large number of quantum bits. Using multiple-valued instead of binary logic has a significant impact on the representable state space and the computational power of quantum machines. It is also worth noting that some known quantum circuit technologies, such as ion traps or quantum dots, are actually three-valued (ternary), rather than binary [6].

In this paper, we consider quantum transforms implemented with nonbinary quantum circuits as a way to extend the quantum machine utility. While we primarily focus on the implementation with ternary quantum gates, the first motivation is to extend the state space with the given number of quantum digits. Further, due to the apparent inability to realize infinite-precision quantum gates, the approximations of quantum functions are useful [10] and likely inevitable [3]. We show that the proposed transform has better approximation properties than QFT implemented with the binary gates. We further both generalize and improve upon known error bounds [5] thanks to considering more than just a binary case.

The paper first reviews the quantum computing in Section 2, followed by the QFT derivations in Section 3. In Section 4, we develop ternary quantum transforms and demonstrate their approximation capabilities.

2 Preliminaries—Quantum States and Gates

Quantum algorithms use the premises of quantum mechanics to achieve parallelism beyond the reach of classical computers. A quantum system realizing such algorithms operates in the Hilbert vector space—we can treat it as a vector space over complex numbers in which the *inner product* over vectors v and w is defined as:

$$\langle v|w\rangle = \sum_{i=1}^{n} v_i^* w_i,$$

where * denotes complex conjugate. The *norm* of a vector v is given as $\|v\| = \sqrt{\langle v|v\rangle}$. The *orthonormal basis* of the space with n quantum basis states $\{|x_1\rangle, |x_2\rangle, \cdots, |x_n\rangle\}$ provides that the inner product of each vector by itself is 1; otherwise, it is 0.

The *quantum state* is a vector, i.e., a linear combination of the basic states: $\alpha_1|x_1\rangle + \alpha_2|x_2\rangle + \cdots + \alpha_n|x_n\rangle$. The complex-valued amplitudes α_i are *wave functions* with respect to basis $|x_1\rangle, |x_2\rangle, \cdots, |x_n\rangle$. The *system evolution* in a quantum system is performed through a linear mapping that preserves the vector norm. Hence, each system evolution that transforms a state $\alpha_1|x_1\rangle + \alpha_2|x_2\rangle + \cdots + \alpha_n|x_n\rangle$ into a state

$$\alpha'_1|x_1\rangle + \alpha'_2|x_2\rangle + \cdots + \alpha'_n|x_n\rangle$$

- Z. Zilic is with McGill University, Montreal, QC H3A 2A7, Canada. E-mail: zeljko@ece.mcgill.ca.
- K. Radecka is with Concordia University, Montreal, QC Canada. E-mail: kasiar@ece.concordia.ca.

Manuscript received 11 Oct. 2005; revised 16 Mar. 2006; accepted 19 May 2006; published online 20 Dec. 2006.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TCSI-0338-1005.

0018-9340/07/$25.00 © 2007 IEEE Published by the IEEE Computer Society

TABLE 1
Standard Quantum Operation Notation

x^*	Complex conjugate: $(a+ib)^* = a-ib$
A^*	Complex conjugate of (all entries of) A
A^T	Transpose of matrix A
$A^\dagger$	Hermitian conjugate: $\begin{bmatrix} a & c \\ b & d \end{bmatrix}^\dagger = \begin{bmatrix} a^* & b^* \\ c^* & d^* \end{bmatrix}$
$\langle\alpha\|\beta\rangle$	Inner product of vectors $\|\alpha\rangle, \|\beta\rangle$
$\|\alpha\rangle \otimes \|\beta\rangle$ or $\|\alpha\rangle\|\beta\rangle$	Tensor product of vectors $\|\alpha\rangle, \|\beta\rangle$

can be expressed via unitary matrix U by $\vec{\alpha}' = U\vec{\alpha}$. As a consequence, such evolutions are always *reversible*. Basic units of information are defined as follows:

Definition 1. *A quantum bit, or qubit, is a binary quantum system, defined over the basis* $\{|0\rangle, |1\rangle\}$. *A q-ary quantum digit is a multiple-valued logic system over basis* $\{0\rangle, |1\rangle, \cdots, |q-1\rangle\}$.

In the literature, the ternary quantum digit is often referred to as a *qutrit*, and we adopt this notation. The state of a single q-valued quantum digit is a vector $c_0|0\rangle + c_1|1\rangle + \cdots + c_{q-1}|q-1\rangle$, where the vector norm is 1:

$$|c_0|^2 + |c_1|^2 + \cdots + |c_{q-1}|^2 = 1.$$

Since all operations are linear, the relevant matrix notation is summarized in Table 1. For any unitary matrix A, A^{-1} equals its Hermitian conjugate (adjoint) $A^\dagger$, i.e., $AA^\dagger = I$.

2.1 Combining the States—Entanglement

Larger quantum systems are obtained by combining individual states. For vectors x and y, the compounded state is a linear combination of the new basis states $|x_iy_j$. Such a state is *decomposable* if it can be represented as:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\alpha_{ij}|x_iy_j\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m}\alpha_i\beta_j|x_i\rangle|y_j\rangle = \sum_{i=1}^{n}\alpha_i|x_i\rangle\sum_{j=1}^{m}\beta_i|y_i\rangle.$$

Otherwise, the state is *entangled*.

Example 1.

1. Consider a system of two qubits, given as

$$\frac{1}{2}\big(|00\rangle + |01\rangle + |10\rangle + |11\rangle\big) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

This system is decomposable as the actions of each qubit are given in the separate brackets on the right-hand side.

2. The system $\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$ is entangled. It is impossible to express it in the form involving individual qubits.

In general, the states are compounded by means of the tensor (Kronecker) product of the basic state spaces. Such a combination is referred to as the *quantum register*. The speedups in quantum computation are often due to parallelism offered by the entanglement. Hence, entanglement is a special new resource in quantum computing.

2.2 Binary Quantum Gates

As with classical circuits, quantum operations can be performed by networks of gates. Among a large number of possible quantum gates, we now review only the gates used in quantum Fourier transforms.

For reversibility, each gate must have the same number of inputs and the outputs. First, single-input quantum gates are defined by 2×2 unitary matrices over complex numbers.

Example 2. Consider the Walsh-Hadamard gate:

$$W_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Its application to state $|\mathrm{x}>$ is:

$$W_2|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^x|1\rangle),$$

which is easily verified by applying it to states $|0>$ and $|1>$. This gate is equal to its own inverse since $(W_2)^2 = I$.

Binary quantum gates with n inputs and n outputs are given by unitary matrices with $2^n \times 2^n$ entries. Multiple-input, multiple output gates perform unitary operations over a tensor product of single quantum states. There are several common constructions of multiple qubit gates [15]. The *controlled gate* approach can be applied to any single-qubit gate G. The first (control) qubit is left unchanged, while the second qubit is affected by G only when the first qubit is $|1>$.

Example 3. The phase shift gate performs multiplication:

$$S_\alpha = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{bmatrix}.$$

The controlled phase shift gate R_α performs the phase shift if the value of the control qubit is 1. The effect of the circuit is given as:

$$|00\rangle \rightarrow |00\rangle, |01\rangle \rightarrow |01\rangle, |10\rangle \rightarrow |10\rangle, |11\rangle \rightarrow e^{i\alpha}|11\rangle.$$

Other multi-qubit constructions include that of n independent single-qubit operations. In that case, the function is given by the Kronecker product of the single-qubit matrices.

3 Quantum Transforms

The Fourier Transform represents elements over an arbitrary Abelian (commutative) group by an expansion over orthogonal sets of basis vectors. We will first describe the traditional development of quantum transforms in the case of multivariate binary inputs, i.e., $(Z_2)^n$, as well as discrete Fourier transforms performed over binary qubits, in which case the group is Z_2^n.

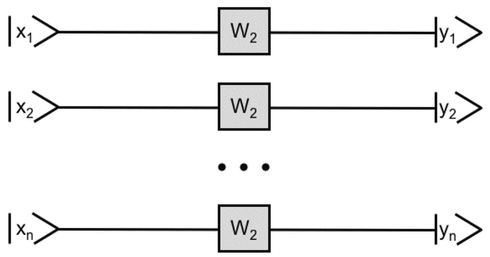

Fig. 1. Circuit for Quantum Walsh-Hadamard Transform.

3.1 Quantum Walsh-Hadamard Transform

The first construction is given in terms of n-variable quantum functions over binary digits. Then, for vectors x and y, QFT equals the quantum Walsh-Hadamard Transform,

$$W_{2^n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{xy}|y\rangle,$$

which we denote in shorthand as:

$$|x\rangle \to \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{xy}|y\rangle.$$

The transformation of n-variable functions is performed by a Kronecker product of univariate transforms. Since the n-fold Kronecker product is equivalent to the parallel application of n single-qubit functions, the overall transform is performed by only n Walsh-Hadamard gates, Fig. 1. The classical fast Walsh Transform requires $O(n2^n)$ operations.

3.2 Quantum Discrete Fourier Transform

In the general case, Quantum Fourier Transforms defined over a set of t values are somewhat more complex to perform. For numbers x and y, the transform has the univariate form:

$$|x\rangle \to \frac{1}{\sqrt{t}} \sum_{y\,\in\,\{0,1,\cdots,t-1\}} e^{\frac{-2\pi i x y}{t}}|y\rangle.$$

Unlike in previous cases, there is no apparent general way to represent the transform by means of the Kronecker product of the decomposable bases. Similarly to the developments in the traditional FFT, several factorizations are possible. We concentrate on the factorization that will allow us to develop the ternary transform in the section to follow.

3.2.1 Transforming 2^n Elements—Qubit Description

The speedup in deriving fast quantum algorithms is due to decomposable realization. The 2^n quantities by n qubits are represented as $y = y_0 + 2y_1 + 2^2y_2 + \cdots 2^{n-1}y_{n-1}$ and the decomposable states are: $|y\rangle = |y_0\rangle|y_1\rangle \cdots |y_{n-1}\rangle$. An application of QFT is then:

$$\frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} e^{-\frac{2\pi i}{2^n}xy}|y\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} e^{-\frac{2\pi i}{2^n}x \sum_{l=1}^{n} y_l 2^l}|y_0 y_1 \cdots y_{n-1}\rangle.$$

The key step, analogous to the FFT, is the decomposition

$$\sum_{y \in \{0,1\}^n} e^{-\frac{2\pi i}{2^n xy}}|y\rangle = \sum_{y' \in \{0,1\}^{n-1}} e^{-\frac{2\pi i}{2^n}x2y'}|y'0\rangle + \sum_{y' \in \{0,1\}^{n-1}} e^{-\frac{2\pi i}{2^n}x(2y'+1)}|y'1\rangle$$

by the least significant bit. Then, QFT is (up to a multiplicative constant) reduced to decomposable states:

$$\sum_{y' \in \{0,1\}^{n-1}} e^{-\frac{2\pi i}{2^n}x2y'}|y'\rangle\Big(|0\rangle + e^{-\frac{2\pi i x}{2^n}}|1\rangle\Big).$$

By using the same construction recursively, QFT reduces to:

$$(|0\rangle + e^{\frac{-\pi i x}{2^0}}|1\rangle)(|0\rangle + e^{\frac{-\pi i x}{2^1}}|1\rangle) \cdots (|0\rangle + e^{\frac{-\pi i x}{2^{n-1}}}|1\rangle). \qquad (1)$$

By dividing x with 2^l, the expression $0.x_{l-1}x_{l-2}\ldots x_0$ denotes the fraction obtained from the least significant l qubits of x as

$$0.x_{l-1}x_{l-2}\cdots x_0 = 2^{-1}x_{l-1} + 2^{-2}x_{l-2} + \cdots + 2^{-l}x_0$$

and the expansion is rewritten into:

$$\Big(|0\rangle + e^{-2\pi i 0.x_0}|1\rangle\Big)\Big(|0\rangle + e^{-2\pi i 0.x_1 x_0}|1\rangle\Big)\cdots \\ \Big(|0\rangle + e^{-2\pi i 0.x_{n-1}x_{n-2}\cdots x_0}|1\rangle\Big).$$

This representation is amenable to the efficient quantum circuit implementations. Each term, multiplied with the appropriate constant, is the unitary transform, built using the Walsh-Hadamard and the controlled rotation gates. The leftmost product term is, by definition, equal to the function performed by the Walsh-Hadamard gate

$$\Big(|0\rangle + e^{-2\pi i 0.x_0}|1\rangle\Big) = \Big(|0\rangle + e^{-\pi i x_0}|1\rangle\Big) = \Big(|0\rangle + (-1)^{x_0}|1\rangle\Big).$$

The remaining terms are realized following the construction in [5]. For the lth term, we first apply the W_2 gate, leading to $\frac{1}{\sqrt{2}}(|0\rangle + (-1)^{x_{l-1}}|1\rangle)$. For each $k < l$, we then perform a phase shift for $\exp(\frac{-\pi i x_k}{2^{l-k}})$, conditional on the kth qubit being nonzero. Recalling Example 3, this operation is performed by the controlled phase shift gate R_k. Since one phase shift gate is employed for each control qubit x_k, $k < l$, this construction is leading up to $n-1$ phase shift gates in total for each qubit. The overall transform can hence be implemented with $n^*(n+1)/2$ binary quantum gates, including n Walsh-Hadamard gates.

4 QFT Using Ternary Quantum Gates

Since the most pressing limitation in making quantum machines is in the number of quantum digits that can be physically realized, using multivalued digit representation has a certain advantage in expanding the state space. To realize in the q-valued case the same number of states as the quantum machine implemented by binary quantum gates, we achieve the reduction by a factor of $\log_2(q)$ qubits. While the proposed implementations are presented here for 3-valued qutrits, the results hold for larger q as well.

4.1 Q-Ary Quantum Gates

Although the q-valued quantum states $(q > 2)$ come quite naturally in many cases [13], there exist a few explicit constructions of q-valued gates. In developing the representation of q-valued states, we find useful the generalization of the Walsh-Hadamard gate, which we call the Chrestenson gate after the Chrestenson Transform [11].

Example 4. For the case $q = 3$, the complex third root of unity is:

$$a = e^{-\frac{2\pi i}{3}} = \cos\left(\frac{-2}{3}\pi i\right) + i\sin\left(\frac{-2}{3}\pi i\right) = -0.5 - i^{*}0.866.$$

The Chrestenson gate performs the mapping given by:

$$CH = \frac{1}{\sqrt{3}}\begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix}.$$

The adjoint of CH is given as:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & a & a^2 \\ 1 & a^2 & a \end{bmatrix}^{\dagger} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a^{*} & (a^2)^{*} \\ 1 & (a^2)^{*} & a^{*} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a^{*} & a \\ 1 & a & a^{*} \end{bmatrix}.$$

The gate is unitary, which can be verified by multiplying it by its Hermitian conjugate. By performing such a multiplication and applying identities $a^3 = 1$ and $1 + a + a^2 = 0$, we obtain the identity matrix.

Multivalued quantum gates used in 3-valued implementation include a controlled phase shift gate, which applies to the incoming signal the multiplication by a factor $e^{(i\alpha x)}$, where x is the input controlling the amount of shift. Unlike in the binary case, there are three possible amounts by which the signal is phase-shifted, depending on the value of x.

4.2 Quantum Chrestenson Transform

The Chrestenson Transform in the q-valued case is given by means of orthogonal expansion:

$$|x\rangle \rightarrow \frac{1}{\sqrt{q^n}} \sum_{y \in \{0,1,\cdots,q-1\}^n} e^{\frac{-2\pi i x y}{q}} |y\rangle.$$

The multipliers are equal to the qth complex roots of the unity. Please note that, by setting $q = 2$, we obtain the quantum Walsh-Hadamard Transform as a special case.

For practical calculation, the input and output vectors to the Chrestenson Transform are expressed in terms of their q-ary expansions:

$$x = \sum_{i=1}^{n} x_i q^i, y = \sum_{i=1}^{n} y_i q^i$$

and the decomposable states are $|x\rangle = |x_0\rangle|x_1\rangle|x_2\rangle \cdots |x_{n-1}\rangle$ and $|y\rangle = |y_0\rangle|y_1\rangle \cdots |y_{n-1}\rangle$.

The transform matrix is then the n-fold Kronecker product of univariate transforms:

$$CH_n = CH \otimes CH \otimes \cdots \otimes CH.$$

Further, the quantum Chrestenson Transform can be written in the recursive form. In the case when $q = 3$ and by recalling that the third root of unity is $a = e^{-\frac{2\pi i}{3}}$, we obtain the following matrix for the n-variable Chrestenson Transform:

$$CH_n = \begin{bmatrix} CH_{n-1} & CH_{n-1} & CH_{n-1} \\ CH_{n-1} & aCH_{n-1} & a^2CH_{n-1} \\ CH_{n-1} & a^2CH_{n-1} & aCH_{n-1} \end{bmatrix}.$$

Similarly to the Walsh-Hadamard case, the transform is performed in parallel by n Chrestenson gates. We note that the implementation primitives have been elaborated upon in [13] in terms of ion-trap quantum devices, where it was shown that the implementation using multiple-valued gates naturally fits the technology.

4.3 QFT Using Ternary Gates

We now present the Quantum Fourier Transform over 3^n elements implemented using ternary quantum circuits. Analogously to the binary case in Section 3.2.1, the QFT factorization is conducted as follows [14]:

$$\begin{aligned} QFT_n|x\rangle &= \frac{1}{\sqrt{3^n}} \sum_{y\in\{0,1,2\}^n} e^{\frac{-2\pi i x y}{3^n}} |y\rangle \\ &= \frac{1}{\sqrt{3^n}} \sum_{y\in\{0,1,2\}^n} e^{-\frac{2\pi i}{3^n} x \sum_{l=1}^{n} y_l 3^l} |y_0 y_1 \cdots y_{n-1}\rangle \\ &= \frac{1}{\sqrt{3^n}} \otimes_{l=0}^{n-1} \left[|0\rangle + e^{-2\pi i x 3^{-l}}|1\rangle + e^{-4\pi i x 3^{-l}}|2\rangle\right] \\ &= \frac{1}{\sqrt{3^n}} \left(|0\rangle + e^{-2\pi i 0.x_0}|1\rangle + e^{-4\pi i 0.x_0}|2\rangle\right) \cdots \\ &\cdots \left(|0\rangle + e^{-2\pi i 0.x_{n-1}\cdots x_1 x_0}|1\rangle + e^{-4\pi i 0.x_{n-1}\cdots x_1 x_0}|2\rangle\right). \end{aligned} \tag{2}$$

In the final form, we employ in this case the ternary fractional number notation:

$$0.x_{l-1}x_{l-2}\cdots x_0 = 3^{-1}x_{l-1} + 3^{-2}x_{l-2} + \cdots + 3^{-l}x_0.$$

The actual circuit for computing the transform is derived by realizing each bracket in (2) with ternary quantum gates. Starting with the leftmost bracket, the whole expression

$$T(x_0) = \frac{1}{\sqrt{3}}\left(|0\rangle + e^{-2\pi i 0.x_0}|1\rangle + e^{-4\pi i 0.x_0}|2\rangle\right)$$

can be realized by a single linear and unitary gate. By considering all three possible values for $|x_0 >$, we obtain exactly the action of the Chrestenson gate and its matrix representation is the same as matrix CH given in Section 4.1.

In the final step of deriving the transform, we expand x in terms of its ternary encoding. Considering the term with 3^l in the denominator, we express $exp(-2\pi i x/3^l)$ from (2):

$$\begin{aligned} &\exp\left(\frac{-2\pi i\left(3^{n-1}x_{n-1} + 3^{n-1}x_{n-1} + \cdots 3^l x_l + \cdots + 3x_1 + x_0\right)}{3^l}\right) \\ &= \exp(-2\pi i K)\exp\left(\frac{-2\pi i x_{l-1}}{3}\right)\cdots\exp\left(\frac{-2\pi i x_1}{3^{l-1}}\right) \\ &\quad \exp\left(\frac{-2\pi i x_0}{3^l}\right). \end{aligned}$$

Since K in this expression is an integer, the first term is always equal to 1. Hence, x_l and higher terms in the ternary expansion of x are omitted.

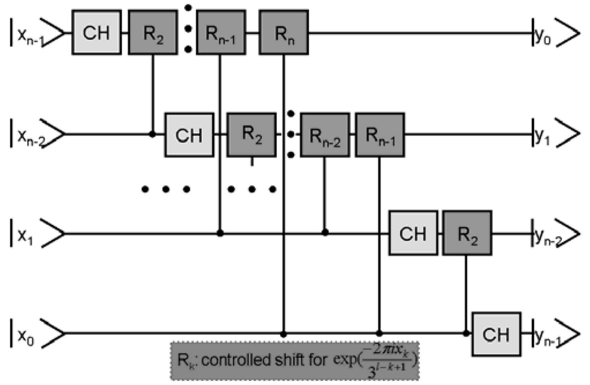

Fig. 2. Circuit for QFT by Q-ary quantum gates.

The rest of the QFT circuit is constructed by implementing each term of this expanded exponential function. Please note that, for each ternary digit (qutrit) x_r, the superposition

$$|0\rangle + e^{-2\pi i x_r 3^{-l}}|1\rangle + e^{-4\pi i x_r 3^{-l}}|2\rangle$$

is realized by the composition of a Chrestenson gate, followed by the phase shift, controlled by x_k $(k < l)$:

$$\exp\left(\frac{-2\pi i x_k}{3^{l-k+1}}\right).$$

Such controlled shifts involve the least significant digits in the fractional expansion of x.

Hence, the overall QFT is performed as in Fig. 2. This implementation consists of one Chrestenson gate per quantum digit, followed by the controlled phase shift gates R_k, for a total of $n^*(n+1)/2$ gates. Please note that the order of digits at the output is reversed.

4.4 Approximation Properties

The given realization of QFT has useful approximation properties [5]. To obtain a reduced complexity circuit, an approximate construction is derived as follows:

4.4.1 Omitting Controlled Phase Shift Gates

The controlled phase shift gates in Fig. 2 perform shifts by exponentially decreasing quantities. Therefore, for the q-valued case, omitting the least significant shift results in a multiplicative factor with error magnitude of

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

since, in the worst case, the phase shift will be the largest when the control bit is equal to $q-1$.

The magnitude of the multiplicative error can be estimated by its Taylor-Maclaurin expansion:

$$\exp\left(\frac{-2\pi i(q-1)}{q^l}\right) = 1 - i\frac{2\pi(q-1)}{q^l} - \frac{4\pi^2(q-1)^2}{2!^*q^{2l}} + i\frac{8\pi^3(q-1)^3}{3!^*q^{3l}} + \cdots,$$

which is rapidly converging to 1 as the higher-order coefficients decrease exponentially in q. Please note that the same error expression holds for the binary case, where we obtain:

$$\exp\left(\frac{-2\pi i}{2^l}\right) = 1 - i\frac{2\pi}{2^l} - \frac{4\pi^2}{2!^*2^{2l}} + i\frac{8\pi^3}{3!^*2^{3l}} + \cdots.$$

Hence, the convergence toward 1 is the slowest among all values of q and multiple-valued implementations of QFT have better approximation properties.

To estimate the phase of the error due to the omitted controlled phase shift gates, the representation of e_m by trigonometric functions is used to easily separates real and imaginary parts:

$$e_m = \cos\left(\frac{2\pi(q-1)}{q^l}\right) - i\sin\left(\frac{2\pi(q-1)}{q^l}\right).$$

Since the argument in both functions tends toward 0 as l increases, the multiplicative error factor is hence close to the real value of 1 and the phase tends to 0. In comparison with the binary case, where the denominator is equal to 2^l, the convergence toward 1 is always faster.

4.4.2 New Bounds for Phase Error

Such good approximation properties allow us to omit more than one controlled phase shift. In general, if m least significant bits are omitted, the overall error exponent is obtained as:

$$\frac{-2\pi i(q^m x_m + \cdots + q\,x_1 + x_0)}{q^l}. \tag{3}$$

The magnitude of the error phase can be determined by the technique in [5] for the approximation error bounds of binary QFT implementation:

$$\frac{2\pi}{q^l}mq^m(q-1) = \frac{2\pi m}{q^{l-m}}(q-1) < \frac{2\pi m}{q^{l-m-1}}.$$

However, as m increases, we now show that even closer bounds are obtained by noticing that the summation of the terms in the bracket of (3) yields a closed-form expression,

$$q^m x_m + \cdots + q\,x_1 + x_0 \leq (q-1)\frac{q^m-1}{q-1} = q^m - 1,$$

and the magnitude becomes bounded by $\frac{2\pi}{q^l}(q^m-1) < \frac{2\pi}{q^{l-m}}$.

This new bound is tighter by a factor of q^*m. Again, by setting $q=2$, we infer that the binary implementation has the worst approximation properties. For our ternary implementation, the error magnitude is hence reduced by factor of $(3/2)^m$ relative to the binary case.

Note that the given error bound depends on the output digit position, and is the largest for $|y_0>$. The least significant digit has the lowest weight in q-ary description of the overall output vector. The most significant digit, however, does not suffer any imprecision due to the omission of the controlled phase shift gates.

5 Conclusions and Future Work

The Quantum Fourier Transform is the key algorithm in quantum computing. In this paper, we presented the implementation of QFT using multiple-valued quantum gates that can overcome the two most serious limitations in realizing practical quantum computing machines.

First, since it is currently hard to realize physical systems with large numbers of entangled qubits, the q-valued approach reduces the requirements for number of qubits by a factor of $\log_2(q)$. Alternatively, for systems employing n qubits, it expands state space by a factor of $(q/2)^n$.

Second, the proposed implementation possesses better approximation properties than the binary one. By eliminating gates performing the smallest controlled phase shifts, the error magnitude decreases exponentially with q. We have also devised new bounds for the error in approximating QFT that are tighter than the original [5] by a factor of q^*m.

More work can be done in the design of circuits consisting of multiple-valued gates, especially following the results showing that such quantum operations might need to be expressed in terms of Fourier Transform [9]. Binary implementations optimized for the circuit depth have been proposed in [3]. It is worth investigating if such techniques are applicable to ternary quantum circuits. Further, the issue of suitable approximation of QFT in such a way that the decoherence problem is alleviated, similarly to [18], is another worthwhile addition to this study.

References

[1] T. Baba, "Development of Quantum Functional Devices for Multiple-Valued Logic Circuits," *Proc. Int'l Symp. Multiple-Valued Logic,* pp. 2-9, 1999.

[2] I.L. Chuang, N. Gershenfeld, and M. Kubinec, "Experimental Implementation of Fast Quantum Searching," *Physical Rev. Letters,* vol. 80, no. 15, pp. 3408-3411, 1998.

[3] R. Cleve and J. Watrous, "Fast Parallel Circuits for the Quantum Fourier Transform," *Proc. IEEE Symp. Theory of Computing,* pp. 526-535, 2000.

[4] W. Cooley and J. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Math. Computation,* vol. 19, pp. 297-301, 1965.

[5] D. Coppersmith, "An Approximate Fourier Transform Useful in Quantum Factoring," IBM Research Report RC 19642, 1994.

[6] X. Deng, T. Hanyu, and M. Kameyama, "Quantum Device Model Based Super Pass Gate for Multiple-Valued Digital Systems," *Proc. Int'l Symp. Multiple-Valued Logic,* pp. 130-138, 1995.

[7] D. Deutsch, "Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer," *Proc. Royal Soc. A,* vol. 400, pp. 97-117, 1985.

[8] N. Gershenfeld and I.L. Chuang, "Bulk Spin-Resonance Quantum Computing," *Nature,* vol. 404, pp. 350-356, 1997.

[9] R. Duan, Z. Ji, Y. Feng, and M. Ying, "A Relation between Quantum Operations and the Quantum Fourier Transform," *Quantum Physics Archive,* arxiv:quant-ph/0304145, Dec. 2003.

[10] L. Hales, "Quantum Fourier Transform and Extensions of the Abelian Hidden Subgroup Problem," PhD dissertation, Univ. of California, Berkeley, 2002.

[11] S.L. Hurst, D.M. Miller, and J. Muzio, *Spectral Techniques in Digital Logic.* Academic Press, 1985.

[12] R. Jozsa, "Quantum Algorithms and the Fourier Transform," *Proc. Royal Soc. London,* vol. 454, pp. 323-337, 1997.

[13] A. Muthukrishnan and C. Stroud Jr., "Quantum Fast Fourier Transform Using Multilevel Atoms," *J. Modern Optics,* vol. 49, pp. 2115-2127, 2002.

[14] Z. Zilic and K. Radecka, "The Role of Super-Fast Transforms in Speeding up Quantum Computation," *Proc. 32nd Int'l Symp. Multiple-Valued Logic,* pp. 89-95, 2002.

[15] M.L. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information.* Cambridge Univ. Press, 2000.

[16] P.W. Shor, "Polynomial Time Algorithms for Prime Factorization and Discrete Logarithm," *SIAM J. Computing,* vol. 26, no. 5, pp. 1484-1509, 1997.

[17] M. Steffen, L.M.K. Vandersypen, and I.L. Chuang, "Toward Quantum Computation: A Five-Qubit Quantum Processor," *IEEE Micro,* vol. 27, no. 1, pp. 24-34, Jan./Feb. 2001.

[18] A. Barenco, A. Ekert, K.-A. Suominen, and P. Torma, "Approximate Quantum Fourier Transform and Decoherence," *Physics Rev. A,* 1999.

Zeljko Zilic received the PhD degree from the University of Toronto in 1997. From 1997 to 1998, he was a member of the technical staff of the FPGA Division, Microelectronics Group at Lucent Technologies. Currently, he is an associate professor at McGill University, Montreal, Canada. He is researching various aspects of system design, test, and verification. He has published more than 100 research papers, holds four patents in the area of clock and power management, and coauthored the book *Verification by Error Modeling* in 2003. He has served as a member of the technical program committees of the ACM International Symposium on FPGAs, IEEE International Test Conference, Midwest Circuits and Systems Symposium, and Electronic Circuits and Systems, and is on the editorial board of the *Journal of Multiple-Valued Logic and Soft Computing* and the *International Journal of Software and Information Technologies*. Professor Zilic is a recipient of a Chercheur Strategique research chair from the province of Quebec. He received the Myril B. Reed Best Paper Award from the IEEE International Midwest Symposium on Circuits and Systems in 2001, a Best Paper Award from the Design and Verification Conference, 2005, and several honorable mention awards. For his undergraduate teaching, the National Council of Deans of Engineering and Applied Science and the Sandford Fleming Foundation awarded Dr. Zilic the Wighton Fellowship in 2006. As a vice president of research and development, he helped to start up Monroi Inc., a company given an Entrepreneur of the Year Award by the St. Laurent Chamber of Commerce in 2004. He is a member of the IEEE and the IEEE Computer Society.

Katarzyna Radecka received the PhD degree from McGill University, Montreal, Canada, in 2003. She received the MEng degree and BEng degree (honors) from the same university in 1997 and 1995. From 1996 to 1998, she was a member of the technical staff at Lucent Technologies in Allentown, Pennsylvania. Currently, she is an assistant professor at Concordia University in Montreal, Canada. Her interests include verification and testing of digital hardware, as well as reversible and quantum computing. She published a book on simulation-based verification in 2003. She is a member of the IEEE.