Approximate Quantum Fourier Transform and Applications: Simulation with Wolfram Mathematica

Alexander N. Prokopenya

Warsaw University of Life Sciences – SGGW, Poland, alexander_prokopenya@sggw.pl

The quantum Fourier transform (QFT) is a unitary transformation U_{FT} that can be written in the computational basis $|x\rangle_n \equiv |x_{n-1} \ldots x_1 x_0\rangle$, where the set of numbers $x_j = 0,1$ $(j = 0,1,\ldots,n-1)$ provides the binary representation of the integer x $(x = 0,1,...,2^{n-1})$, as

$$U_{FT}|x\rangle_n \to \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} \exp\left(\frac{2\pi i}{2^n} xy\right) |y\rangle_n \ . \tag{1}$$

Here n is a number of qubits in the memory register.

Note that the QFT can be carried out efficiently by a quantum circuit built entirely out of single-qubit and two-qubit gates. Actually, only n the Hadamard gates, $n(n-1)/2$ controlled phase shift gates R_k, and $\lfloor n/2 \rfloor$ Swap gates are required to build such a circuit (see, for example, [1]). The execution time of the QFT grows only as n^2 and, therefore, the QFT is executed exponentially faster that the classical fast Fourier transform.

However, in case of a large number of qubits the phase shift $2\pi/2^k$ becomes exponentially small, while a practical implementation of the high precision controlled phase shift gates R_k may be very difficult. So it would be very useful if the full Fourier transform (1) could be replaced by the approximate QFT, where only finite degree phase shift gates R_k are involved $(k \le m < n)$. Analysis of different quantum algorithms has shown that applying the approximate QFT can yield even better results than the full Fourier transform [2].

In the present paper we discuss an approximate QFT that was first proposed in [3] and can be represented as

$$\begin{aligned} \tilde{U}_{FT}|x_{n-1}\ldots x_0\rangle \to \frac{1}{2^{n/2}} \sum_{y_{n-1}=0}^{1} \ldots \sum_{y_0=0}^{1} \exp\Bigg(2\pi i \Bigg(x_{n-1}\frac{y_0}{2} + \ldots + \\ x_{n-m}\left(\frac{y_{m-1}}{2^1} + \ldots + \frac{y_0}{2^m}\right) + x_{n-m-1}\left(\frac{y_m}{2^1} + \ldots + \frac{y_1}{2^m}\right) + \ldots + \\ x_0\left(\frac{y_{n-1}}{2^1} + \ldots + \frac{y_{n-m}}{2^m}\right)\Bigg)\Bigg) |y_{n-1}\ldots y_0\rangle \ . \end{aligned} \tag{2}$$

Applying the approximate QFT, one can expect that an accuracy of computation decreases in comparison to the case of the full QFT, and probability of getting a correct result reduces, as well. The main purpose of the present paper is to estimate a probability of successful solution of a problem in case of the replacement of the full QFT by the approximate QFT and to demonstrate the results by simulation of some quantum algorithms, using the Wolfram Mathematica package "QuantumCircuit" (see [4, 5]). Recall that the package provides a user-friendly interface to specify a quantum circuit, to draw it, and to construct the corresponding unitary matrix for quantum computation defined by the circuit. Using this matrix, one can find the final state of the quantum memory register by its given initial state and to check the operation of the algorithm determined by the quantum circuit.

As examples we consider here two known quantum algorithms, where the QFT plays an essential role. The first one is the quantum algorithm for phase estimation based on the QFT [6]. We have obtained the lower bound on the probability of the successful phase estimation in both cases of the full and approximate QFT and shown that the decrease in the accuracy of phase estimation results in increasing the probability of the successful problem solution [7].

The second example is the Shor algorithm for order finding [8]. Our calculations show that using the approximate QFT gives the same results as in case of the full QFT even for small enough degree m of approximation. At the same time the probability to obtain correct result decreases only a little bit in comparison to the case of the full QFT. The validity of the results is demonstrated by simulation of the algorithm using the package "QuantumCircuit".

References

[1] M. Nielsen, I. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press (2000).

[2] A. Barenco, A. Ekert, K.A. Suominen, P. Törmä, *Approximate quantum Fourier transform and decoherence*, Phys. Rev. A

[3] D. Coppersmith, *An approximate Fourier transform useful in quantum factoring*, IBM Research Report RC 19642 (1994). **54**, 1, pp.139-146 (1996).

[4] V.P. Gerdt, R. Kragler, A.N. Prokopenya, *A Mathematica program for constructing quantum circuits and computing their unitary matrices*, Physics of Particles and Nuclei, Lett. **6**, 7, pp. 526-529 (2009).

[5] V.P. Gerdt, R. Kragler, A.N. Prokopenya, *A Mathematica package for simulation of quantum computation*, in *Computer Algebra in Scientific Computing / CASC'2009*, V.P. Gerdt, E.W. Mayr, E.V. Vorozhtsov (ed.), LNCS 5743, Springer-Verlag, Berlin, pp. 106-117 (2009).

[6] D.S. Abrams, S. Lloyd, *Quantum algorithm providing exponential speed increase for finding eigenvales and eigenvectors*, Phys. Rev. Lett. **83**, 24, pp. 5162-5165 (1999).

[7] A.N. Prokopenya, *Simulation of a quantum algorithm for phase estimation*, Programming and Computer Software **41**, 2, pp. 98-104 (2015).

[8] P.W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Comp. **26**, 5, pp. 1484-1509 (1997).