Phys 571 Quantum Computation **Fall 2009**

Homework Problem 3.3 **Due Thursday, November 19**
(10 points) **(at lecture)**

3.3 **The quantum Fourier transform and the fast Fourier transform.** The *quantum Fourier transform* (QFT) on N qubits is a unitary operator defined in the standard (position) basis $\{|j\rangle, j = 0, \ldots, D-1\}$ by

$$F_N|j\rangle = \frac{1}{2^{N/2}} \sum_{k=0}^{D-1} e^{2\pi i jk/D} |k\rangle \ ,$$

where $D = 2^N$ is the dimension of the N-qubit Hilbert space. Since j and k are N-bit binary numbers, i.e.,

$$j = j_1 \ldots j_N.0 = \sum_{l=1}^{N} j_l 2^{N-l} \qquad \text{and} \qquad k = k_1 \ldots k_N.0 = \sum_{l=1}^{N} k_l 2^{N-l} \ ,$$

we can write the Fourier-transform coefficient as

$$e^{2\pi i jk/D} = \prod_{l=1}^{N} e^{2\pi i k_l j/2^l} = \prod_{l=1}^{N} e^{2\pi i k_l 0.j_{N-l+1}\ldots j_N} \ . \qquad \text{(I)}$$

This allows us to put the QFT in the form

$$F_N|j\rangle = \frac{1}{2^{N/2}} \bigotimes_{l=1}^{N} \sum_{k_l=0}^{1} |k_l\rangle\, e^{2\pi i k_l 0.j_{N-l+1}\ldots j_N} = \frac{1}{2^{N/2}} \bigotimes_{l=1}^{N} \Big(|0\rangle + e^{2\pi i 0.j_{N-l+1}\ldots j_N} |1\rangle \Big) \ ,$$

which is suitable for translation into an efficient quantum circuit that does the QFT in $O(N^2)$ elementary gates.

Applying the QFT to a quantum state $|\psi\rangle = \sum_j x_j |j\rangle$ changes the amplitudes in the standard basis according to

$$y_j = \langle j|F_N|\psi\rangle = \sum_{k=0}^{D-1} \langle j|F_N|k\rangle x_k = \frac{1}{2^{N/2}} \sum_{k=0}^{D-1} e^{2\pi i jk/D} x_k \ .$$

This transformation is called the *discrete Fourier transform* (DFT). The QFT doesn't, of course, compute the DFT, i.e., compute the output amplitudes y_k; it just transforms amplitudes according to the DFT.

To compute the DFT involves taking the matrix product of a $D \times D$ matrix and a D-dimensional input vector. A naïve approach to this matrix multiplication requires evaluating $O(D^2)$ products. *Use* the identity (I) above to devise an algorithm for computing the DFT using only $O(ND)$ elementary multiplications. This algorithm is called the fast Fourier transformation (FFT); it is crucial for practical applications of Fourier-transform methods.