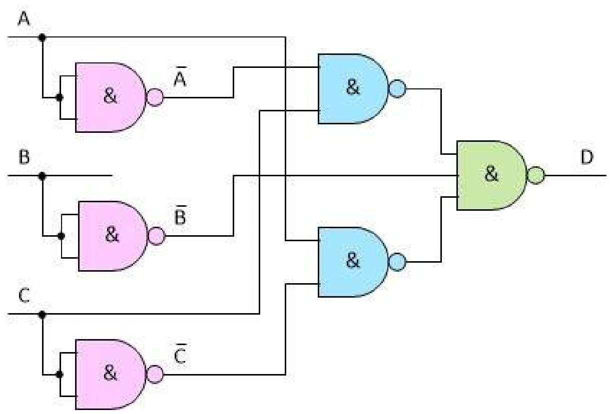
CSE491/596 Lecture Mon. 11/27/23: Quantum Circuits and Their Analysis

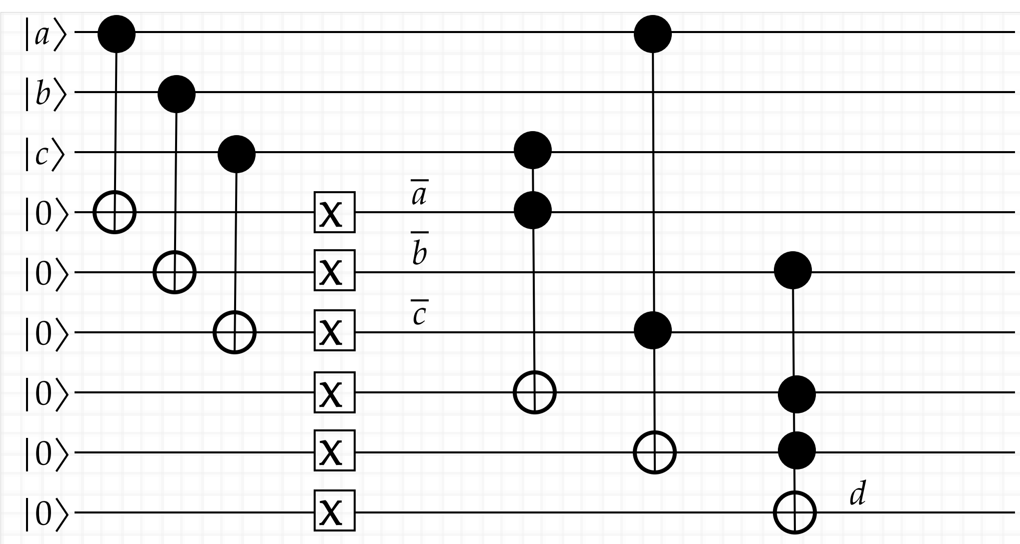
Theorem (cf. theorem 5.2 in section 5.3): Classical Boolean circuits can be efficiently simulated by quantum circuits that don't even do any superposition or entanglement.

The proof is basically that the Toffoli gate simulates NAND via $\mathrm{Tof}(x, y, 1) = (\bar{x} \vee \bar{y})$ and NAND is a universal gate. The extra lines for the constant 1 inputs also make the whole computation **reversible**.

Here is a sizable example of this theorem. Consider the following circuit of NAND gates from the [blog article](#) "Implementing Logic Functions Using Only NAND or NOR Gates" by Max Maxfield:



Here is the corresponding quantum circuit:



(We will later mirror the gates except the last one giving the function value d in order to reset the ancilla qubits 4--8 to $|0\rangle$, so that all qubits except the last keep their given basis values.)

Note also that the initial three CNOT gates effectively copy the Boolean values a, b, c so that they can be negated as $\bar{a}, \bar{b}, \bar{c}$ on the next three qubit lines. This is covered in section 6.2, and the last three qubit lines exemplify the trick in section 6.1 of using NOT gates to effectively initialize them to $|1\rangle$ rather than $|0\rangle$. *Caveat:* You can't copy an arbitrary quantum state using CNOT---the **No-Cloning**

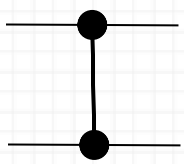
Theorem mentioned in section 6.2 shows there is no way to do this in general. But particular states in a known basis can be copied this way.

The "quantum extra", beginning with using the Hadamard gate to create superpositions, is what promises to take us beyond classical computing.

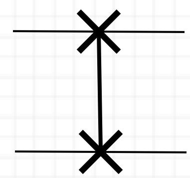
The Quantum Fourier Transform

Two other 2-qubit gates and their matrix and circuit representations are:

$$\mathbf{CZ} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

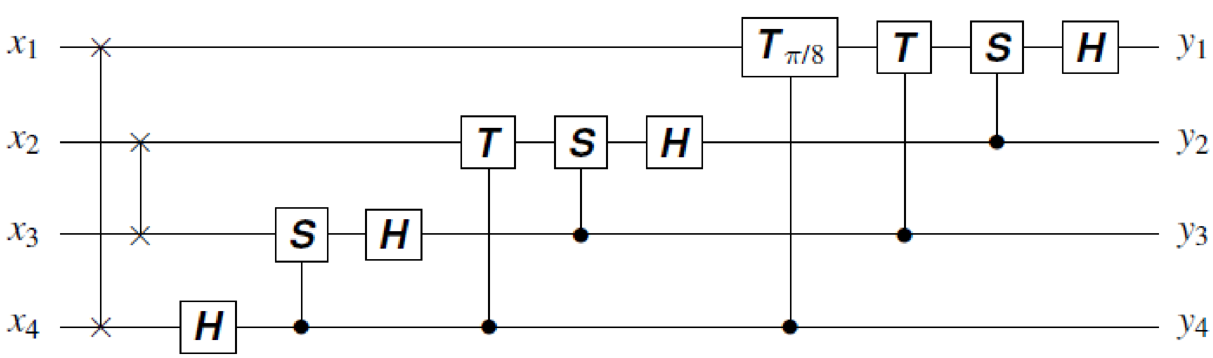


$$\mathbf{SWAP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



The $\mathbf{CZ}$ gate is symmetric: note that its results on $|01\rangle$ and on $|10\rangle$ are the same. So are the $\mathbf{CS}$ and $\mathbf{CT}$ gates, which have i and $\omega = e^{i\pi/4} = \sqrt{i}$ in place of the -1. For a general $r \times r$ matrix A, $\mathbf{C}A$ is the $2r \times 2r$ matrix given in block form as $\begin{bmatrix} I & 0 \\ 0 & A \end{bmatrix}$. The circuit convention is to put a black dot on the **control** qubit line and a vertical line extending to A in a box the **target** line(s). Most applets make you do that with $\mathbf{CZ}$ as well as $\mathbf{CS}$ and $\mathbf{CT}$, but it is good to remember that these three (and further ones with roots of ω at bottom right) are symmetric.

Continuing the idea of the progression $\mathbf{CZ}$, $\mathbf{CS}$, $\mathbf{CT}$,... to finer angles leads into the general construction of $O(n^2)$-sized circuits of basic gates for the n-qubit **Quantum Fourier Transform** (**QFT**). The usual recursive way to build it via $O(n^2)$ unary and binary gates uses controlled rotations by exponentially tiny angles. This is already evident from the four-qubit illustration in the textbook (where the two gates on the left are :



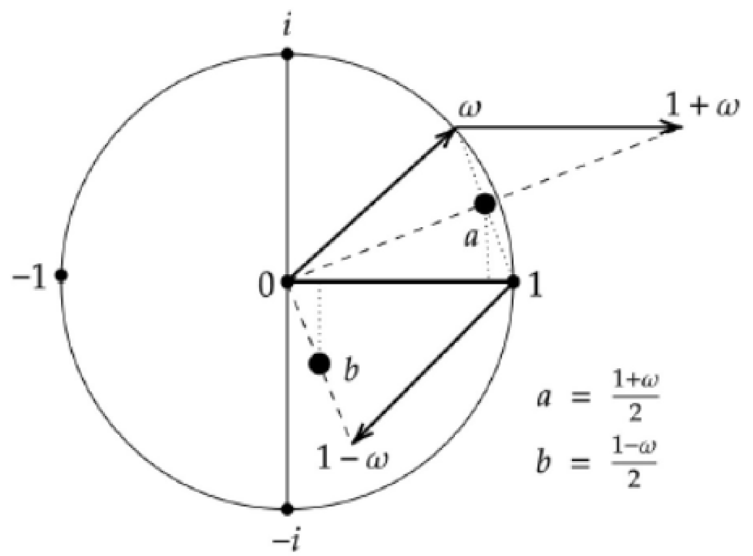
Here $T_{\pi/8} = \begin{bmatrix} 1 & 0 \\ 0 & \omega' \end{bmatrix}$ with $\omega' = e^{i\pi/8}$ not $\omega = e^{i\pi/4}$ as with the T-gate. So ω' has a phase angle one-sixteenth of a circle. For $n = 5$ the next bank uses $1/32$, then $1/64$, and soon the angles would be physically impossible so the gates could never be engineered.

Those super-tiny angles are in the definition of the QFT itself. For any n, it takes $\omega_n = e^{2\pi i/N}$ where $N = 2^n$. With $n = 3$, the matrix together with its quantum coordinates is:

$$\begin{array}{c|cccccccc}
 & 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \\
\hline
000 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
001 & 1 & \omega & i & i\omega & -1 & -\omega & -i & -i\omega \\
010 & 1 & i & -1 & -i & 1 & i & -1 & -i \\
011 & 1 & i\omega & -i & \omega & -1 & -i\omega & i & -\omega \\
100 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
101 & 1 & -\omega & i & -i\omega & -1 & \omega & -i & i\omega \\
110 & 1 & -i & -1 & i & 1 & -i & -1 & i \\
111 & 1 & -i\omega & -i & -\omega & -1 & i\omega & i & \omega
\end{array}
=
\begin{array}{c|cccccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & \omega & \omega^2 & \omega^3 & -1 & \omega^5 & \omega^6 & \omega^7 \\
2 & 1 & \omega^2 & \omega^4 & \omega^6 & 1 & \omega^2 & \omega^4 & \omega^6 \\
3 & 1 & \omega^3 & \omega^6 & \omega & -1 & \omega^7 & \omega^2 & \omega^5 \\
4 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
5 & 1 & \omega^5 & \omega^2 & \omega^7 & -1 & \omega & \omega^6 & \omega^3 \\
6 & 1 & \omega^6 & \omega^4 & \omega^2 & 1 & \omega^6 & \omega^4 & \omega^2 \\
7 & 1 & \omega^7 & \omega^6 & \omega^5 & -1 & \omega^3 & \omega^2 & \omega
\end{array}$$

$$\mathbf{QFT}[i,j] = \omega^{ij}$$

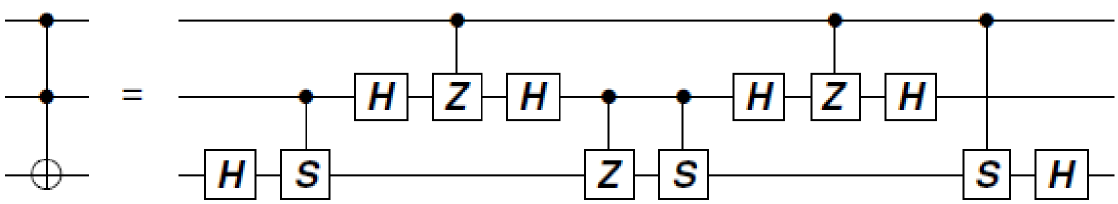
For $\mathbf{QFT}_N$ we raise ω_N with its tiny phase to exponentially many different powers. How can this possibly be feasible? Leonid Levin among others raised this objection. Here are several answers:



- Basic gates can fabricate quantum states having finer phases. This is already hinted by the diagram in the case of $\mathbf{HTH}$. Try composing $\mathbf{HTHT^*H}$ and $\mathbf{HTHT^*HTHT^*H}$. The *Solovay-Kitaev theorem* enables approximating operators with exponentially fine angles by polynomially many gates of phases that are multiples of ω (using **CNOT** to extend this to multiple-qubit operators).
- The Toffoli and Hadamard gates by themselves, which have phases only $+1$ and -1, can simulate the real parts and imaginary parts of quantum computations separately via binary code, in a way that allows re-creating all measurement probabilities. (This is undertaken in exercises 7.8--7.14 with a preview in the solved exercise 3.8.)
- The **CNOT** and Hadamard gates do not suffice for this, even when the so-called "phase gate"

$\mathbf{S} = \mathbf{T}^2 = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$ is added. The Pauli $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ gates and also $\mathbf{CZ}$ can be built from these, but quantum circuits of these gates can be simulated in deterministic ("classical") polynomial time. However, $\mathbf{CS}$ suffices to build the Toffoli gate, per the diagram below (which is also a presentation option). So Hadamard + $\mathbf{CS}$ is a universal set using only quarter phases.

- The signature application of the QFT, which is *Shor's algorithm* showing that factoring belongs to BQP, may only require coarsed-grained approximations to $\mathbf{QFT}_N$. Indeed, the above theorem about Hadamard and Toffoli gates implies that they can efficiently represent the *acceptance outcomes* of any quantum circuit---though not its complex amplitudes. This extends to the replacement of Toffoli by Controlled-$\mathbf{S}$ owing to the equation we have seen:



For these reasons, $\mathbf{QFT}_N$ is considered feasible even though $N = 2^n$ is exponential. Not every $N \times N$ unitary matrix U is feasible---the Solovay-Kitaev theorem relies on U having a small exact formulation to begin with. But if we fix a finite **universal gate set** (such as $\mathbf{H} + \mathbf{T} + \mathbf{CNOT}$, $\mathbf{H} + \mathbf{Tof}$, or $\mathbf{H} + \mathbf{CS}$ above) and use only matrices that are compositions and tensor products of these gates, then we can use the simple gate-counting metric as the main complexity measure.

Outputs and Measurements

There are various conventions about what it means for a family $[C_n]$ of quantum circuits to compute a function f on $\{0,1\}^*$, where f is an ensemble of functions f_n on $\{0,1\}^n$ and each C_n computes f_n. I like supposing that $f(x)$ is coded in $\{0,1\}^r$ where r depends only on n and giving C_n r-many output qubits separate from the n input qubits, plus some number m of ancilla qubits. (It is traditional, IMHO weirdly, to consider that the primordial input is always $\mathbf{0}^n$ and that for any other x, **NOT** gates are prepended onto the circuit for those lines i where $x_i = 1$.)

For *languages*, this means that the yes/no verdict comes on qubit $n+1$. Many references say to measure line 1 instead. (Using a swap gate between lines 1 and $n+1$ can show these conventions to be equivalent, but I prefer reserving lines 1 to n for *potential* use of the "copy-uncompute" trick, which is covered in section 6.3 and is a presentation option.) Even for languages, however, one evidently cannot get the most power if you need always to rig the circuit so that on any input $x \in \{0,1\}^n$, the output line always has a (standard-)basis value, i.e., is $\mathbf{0}$ with certainty or is $\mathbf{1}$ with certainty. Instead, one must **measure** it, whereupon the value $\mathbf{0}$ is given with some probability p, $\mathbf{1}$ with probability $1 - p$.

The math of measurements (at least of the kind of *pure states* we get in completely-specified circuits) is simple. At the end we have a quantum state Ψ of $n + r + m$ qubits, counting the output and any ancilla lines. It "is" a vector $(v_1, v_2, \dots v_Q) \in \mathbb{C}^Q$ where $Q = 2^{n+r+m}$. Numbering $\{0,1\}^{n+r+m}$ in canonical order as $z_1, \dots, z_S$, an **all-qubits measurement** gives any z_j with probability $|v_j|^2$. If we focus on just the r output lines, then any $y \in \{0,1\}^r$ occurs with probability

$$\sum_{j:\, z_j \text{ agrees with } y \text{ on the } r \text{ output lines}} |v_j|^2.$$

When $r = 1$ and $y = 0$ the sum is over all binary strings z_j that have a 0 in the corresponding places. It is a postulate of quantum mechanics that we could do the measurement in such a way that the new state Ψ' stays "coherent" on qubit lines outside the r lines that were measured, but we will not care about this---we will be OK doing an all-qubits measurement (which "collapses" the system down to $|z_j\rangle$ for whatever basis state z_j is yielded) and then re-starting the whole circuit to do multiple trials, if necessary. What can make them necessary is the simple "unamplified" definition of BQP along lines of the definition given for BPP. To simplify the notartion, let p_x denote the probability of measuring 1 on the output qubit line. The notion of uniformity is similar to that for ordinary Boolean circuits: it means that C_n can be written down in $n^{O(1)}$ (classical) time.

Definition: A language L belongs to BQP if there is a uniform family $[C_n]$ of polynomial-sized quantum circuits such that for all n and inputs $x \in \{0,1\}^n$,

$$\begin{aligned} x \in L &\implies p_x \geq 3/4; \\ x \notin L &\implies p_x \leq 1/4. \end{aligned}$$

With the help of ideas grouped under the term "**principle of deferred measurement**" (mentioned in section 6.6), the idea of amplifying the difference in probabilities by repeated trials and majority vote of the outcomes can be internalized within the circuits. This needs polynomially more ancilla qubits but allows doing only one measurement, which will then be guaranteed to give the correct answer with probability supremely close to 1 rather than probability $3/4$. However, it is (IMHO) more helpful to think instead of quantum circuits as objects that can be ***sampled***, and that a final classical post-processing routine gives the final answer as a function of the results of the samples. This is how Simon's algorithm, Shor's algorithm, and (general forms of) Grover's algorithm are usually conceived. The same approach of assembling a value $g(x)$ from multiple sample results can likewise be used for defining how functions g are computed.

With that said, the idea of computing a function $f(x) = y$ with y represented literally within a quantum (basis) state is often applied a different way. Given a circuit C computing y on lines $n+1, \dots, n+r$ that way---and using "copy-uncompute" to restore x on lines $1, \dots, n$---make C' by prepending $\mathbf{H}^{\otimes n}$ on the first n lines. Give C' $|0^n\rangle$ as the actual input. The resulting state is

$$s_f = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle|f(x)\rangle.$$

Although each individual term $|x\rangle|f(x)\rangle$ is separable---indeed, it is the basis state $\mathbf{e_x} \otimes \mathbf{e_y} = \mathbf{e_{xy}}$ where $y = f(x)$---the sum is usually majorly entangled. Our text calls this the **functional superposition** of f over the domain $\{0, 1\}^n$. In **Shor's algorithm** for a product $M = pq$ of two primes, first a seed $a < M$ is chosen randomly from the $\rho = (p-1)(q-1)$ numbers that are not multiples of p or q. Then $f(x)$ is the function $a^x \bmod M$, where x is redundantly allowed to go as high as $Q-1$ with Q being a power of 2 between M^2 and $2M^2$. That makes enough room for the periodicity of the powering mod M to make enough waves for the QFT to do what Joseph Fourier knew it would 198 years ago: it transforms the waves' period, which divides ρ, into a peak. Repeated runs and measurements eventually give enough information about ρ to infer p and q.

Thus Shor's algorithm invokes *both* the "input x, output $f(x)$" view of what a quantum circuit does and the randomized sampling view. The latter is the external algorithm, and its input is not "x" but rather C, which in turn comes from the factoring problem instance M and the random seed a. In lieu of covering the full details in chapters 11 and 12, we can state:

Shor's Theorem: FACTORING is in BQP.

At present, I accept that s_f is feasible to build and the QFT is feasible to apply---at least with sufficient approximation for Shor's algorithm to work. However, I am chary of the account given under the Many Worlds Hypothesis. As told by David Deutsch and others, each Hadamard gate branches into two universes. If the n Hadamards stayed separate to make n pairs that might be reasonable, but building s_f seems to entail piggy-backing them to make 2^n universes, all harnessed together by the QFT.