NOTES ON QUANTUM COMPUTING

LUÍS R. A. FINOTTI

Contents

1. Qubits

Notation 1.1. The following notation is commonly used:

(1) *Qubit*:

$$
\begin{aligned}
|0\rangle &\overset{\text{def}}{=} \begin{pmatrix} 1 \\ 0 \end{pmatrix}, & |1\rangle &\overset{\text{def}}{=} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\
|+\rangle &\overset{\text{def}}{=} \frac{\sqrt{2}}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, & |-\rangle &\overset{\text{def}}{=} \frac{\sqrt{2}}{2}\begin{pmatrix} 1 \\ -1 \end{pmatrix} \\
|\mathrm{i}\rangle &\overset{\text{def}}{=} \frac{\sqrt{2}}{2}\begin{pmatrix} 1 \\ \mathrm{i} \end{pmatrix}, & |-\mathrm{i}\rangle &\overset{\text{def}}{=} \frac{\sqrt{2}}{2}\begin{pmatrix} 1 \\ -\mathrm{i} \end{pmatrix}.
\end{aligned}
$$

(2) *Quantum State*:

$$|\psi\rangle = \begin{pmatrix}\alpha \\ \beta\end{pmatrix}, \qquad \text{with } |\alpha|^2 + |\beta|^2 = 1,$$

where $|\alpha|^2$ is the probability of the qubit measuring as the 0 state and $|\beta|^2$ is the probability of the qubit measuring as the 1 state. In other words

$$\mathbb{P}\left(0 \mid \psi\right) = |\alpha|^2 \quad \text{and} \quad \mathbb{P}\left(1 \mid \psi\right) = |\beta|^2.$$

(3) *Dirac Notation* (and the *ket vector*):

$$|\psi\rangle = \begin{pmatrix}\alpha \\ \beta\end{pmatrix} = \alpha\,|0\rangle + \beta\,|1\rangle\,.$$

(4) We denote by $\oplus$ the addition in $\mathbb{F}_2$. E.g., if $x \in \mathbb{F}_2$, we have that

$$|x \oplus 1\rangle = \begin{cases} 1, & \text{if } x = 0; \\ 0, & \text{if } x = 1. \end{cases}$$

(So, this is example is like the *not* gate.)

(5)

$$|\psi(\theta,\phi)\rangle \overset{\text{def}}{=} \cos(\theta/2)\,|0\rangle + \mathrm{e}^{\mathrm{i}\phi}\sin(\theta/2)\,|1\rangle\,.$$

(See Section 4 below.)

(6) If $|\phi\rangle = \lambda\,|\psi\rangle$ with $|\lambda| = 1$, then $|\phi\rangle$ and $|\psi\rangle$ are *physically equivalent*. So we can disregard the overall phase.

2. Tensor Products

Notation 2.1. We denote, e.g.,

$$|010011\rangle \overset{\text{def}}{=} |0\rangle \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle \otimes |1\rangle \otimes |1\rangle\,.$$

Also, as usual,

$$|0\rangle^{\otimes 5} \overset{\text{def}}{=} |00000\rangle\,.$$

Remark. If we have $\alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$, then

$$\begin{aligned} |\alpha|^2 &= \text{probability of measuring } |00\rangle, \\ |\beta|^2 &= \text{probability of measuring } |01\rangle, \\ |\gamma|^2 &= \text{probability of measuring } |10\rangle, \\ |\delta|^2 &= \text{probability of measuring } |11\rangle. \end{aligned}$$

Definition 2.2. The *computational basis* is simply the induced basis of $(\mathbb{C}^2)^{\otimes n}$: $\{|i_1 i_2 \dots i_n\rangle \ : \ i_j \in \mathbb{F}_2\}$. The order in Qiskit is done via the binary representation (with unit digit coming *last*), e.g.,

$$\{|000\rangle\,, |001\rangle\,, |010\rangle\,, |011\rangle\,, |100\rangle\,, |101\rangle\,, |110\rangle\,, |111\rangle\}.$$

One can sometimes represent this basis (in this same order) as

$$\{|0\rangle_3\,, |1\rangle_3\,, |2\rangle_3\,, |3\rangle_3\,, \dots, |7\rangle_3\}.$$

If $|\psi\rangle = \sum_{x \in \mathbb{F}_2^n} \psi_x\,|x\rangle$, then we have:

$$\mathbb{P}\,(x \mid \psi) = |\psi_x|^2 \quad \text{and} \quad \sum_{x \in \mathbb{F}_2^n} |\psi_x|^2 = 1.$$

Notation 2.3. Note that two qubit-strings $|\psi_1\rangle$ and $|\psi_2\rangle$ are such that $|\psi_2\rangle = \lambda\,|\psi_1\rangle$, with $|\lambda| = 1$, then they are physically equivalent (and hence indistinguishable). We shall write:

$$|\psi_1\rangle \equiv |\psi_2\rangle\,.$$

3. Inner Product

We have the *inner product* in $(\mathbb{C}^2)^{\otimes n}$: if $|\psi\rangle = \sum_{x \in \mathbb{F}_2^n} \psi_x\,|x\rangle$ and $|\phi\rangle = \sum_{x \in \mathbb{F}_2^n} \phi_x\,|x\rangle$, then

$$\langle \psi \mid \phi \rangle \overset{\text{def}}{=} \sum_{x \in \mathbb{F}_2^n} \overline{\psi}_x \phi_x.$$

(So, it is simply the inner product that makes the computational basis *orthonormal.*)

Of course, for $x, y \in \mathbb{F}_2^n$, we have that $\langle x \mid y \rangle = \delta_{x,y}$.

Note that we denote by $\langle \psi |$ (the *bra vector* of ψ) the dual element of $| \psi \rangle$:

$$\langle \psi | = \sum_{x \in \mathbb{F}_2^n} \overline{\psi}_x \langle x | .$$

Hence,

$$\langle \psi | \, | \phi \rangle = \langle \psi \mid \phi \rangle.$$

4. Bloch Sphere

Reference: Bloch Sphere | Visualizing Qubits and Spin | Quantum Information

Note that if $|\rho| = 1$, then $|\psi\rangle \equiv \rho \, |\psi\rangle$, as we don't care about the *overall* phase. So, if

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha \, |0\rangle + \beta \, |1\rangle ,$$

then $|\alpha|^2 + |\beta|^2 = 1$, so

$$\begin{aligned} \alpha &= \cos(\gamma) \mathrm{e}^{\mathrm{i}\delta}, \\ \beta &= \sin(\gamma) \mathrm{e}^{\mathrm{i}\epsilon}, \end{aligned}$$

with $\gamma, \delta, \epsilon \in \mathbb{R}$. Now,

$$\begin{aligned} |\psi\rangle &= \cos(\gamma) \mathrm{e}^{\mathrm{i}\delta} \, |0\rangle + \sin(\gamma) \mathrm{e}^{\mathrm{i}\epsilon} \, |1\rangle \\ &= \mathrm{e}^{\mathrm{i}\delta} \left(\cos(\gamma) \, |0\rangle + \mathrm{e}^{\mathrm{i}(\epsilon - \delta)} \sin(\gamma) \, |1\rangle \right) \\ &\equiv \cos(\gamma) \, |0\rangle + \mathrm{e}^{\mathrm{i}(\epsilon - \delta)} \sin(\gamma) \, |1\rangle . \end{aligned}$$

So, we have that

$$|\psi\rangle \equiv |\psi(\theta, \phi)\rangle \overset{\text{def}}{=} \cos(\theta/2) \, |0\rangle + \mathrm{e}^{\mathrm{i}\phi} \sin(\theta/2) \, |1\rangle , \qquad \theta \in [0, \pi], \ \phi \in [0, 2\pi]. \tag{1}$$

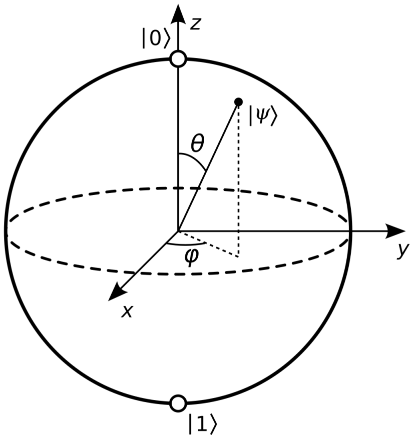


Figure 1. Bloch Sphere

Note that $\epsilon - \delta$ is the *relative phase*. Clearly, the relative phase does not change the probabilities.

Considering:

- θ the angle in $\mathbb{R}^3$ with the z-axis;
- ϕ the angle in $\mathbb{R}^3$ with the x-axis around the z-axis;

we get a sphere (with spherical coordinates and radius 1), the *Bloch sphere* (in Fig. 1).

Let

$$\hat{\eta} = \begin{pmatrix} \sin(\theta)\cos(\phi) \\ \sin(\theta)\sin(\phi) \\ \cos(\theta) \end{pmatrix}$$

be a direction/point on the sphere (in spherical coordinates). Then, the spin operator in the direction of $\hat{\eta}$ (with angles θ and ϕ as above) the Bloch state $|\psi(\theta, \phi)\rangle$ is an eigenfunction with positive eigenvalue $\hbar/2$. (FIXME! Need details!)

5. 1-Qubit Gates

Notation 5.1. (1) If A is a square complex (or real) matrix, then we denote by $A^\dagger$ the *adjoint*, i.e., the complex conjugate of the transpose of A).

(2) An $n \times n$ matrix is *unitary* if $A^\dagger = A^{-1}$. (So, it must be invertible.)

(3) We shall denote the set of $n \times n$ unitary complex matrices by $\mathrm{U}_n(\mathbb{C})$.

Remark. Note that unitary matrices preserve inner product, and therefore norms.

A physically closed system, quantum systems evolve through unitary operations.

Definition 5.2. A 1-qubit *gate* is simply an element of $\mathrm{U}_2(\mathbb{C})$ acting on a single qubit. More gerenerally an n-qubit gate is an element of $\mathrm{U}_{2^n}(\mathbb{C})$.

In a *physical* level, implementing 1-qubit gates is "easy", but n-qubit gates are usually hard. So, we need a small set of simple (physically, so n-qubit gates with n small, i.e., mostly $n = 1$, and a few with $n = 2$, and maybe $n = 3$, and also easy to understand and use) and universal (meaning that we can produce any n-gate from them) set of gates.

5.1. Pauli Gates.

Notation 5.3. (1) *X Gate* or *σ_1 Gate*: multiplication by:

$$\sigma_1 = X \overset{\text{def}}{=} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

It is basically the NOT gate, $|x\rangle \mapsto |x \oplus 1\rangle$. So, $\alpha\,|0\rangle + \beta\,|1\rangle \mapsto \beta\,|0\rangle + \alpha\,|1\rangle$.

(2) *Y Gate* or *σ_2 Gate*: multiplication by:

$$\sigma_2 = Y \overset{\text{def}}{=} \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}$$

So, $\alpha\,|0\rangle + \beta\,|1\rangle \mapsto -\mathrm{i}\beta\,|0\rangle + \alpha\mathrm{i}\,|1\rangle \equiv \beta\,|0\rangle - \alpha\,|1\rangle$.

(3) *Z Gate* or σ_3 *Gate*: multiplication by:

$$\sigma_3 = Z \overset{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

So, $\alpha\,|0\rangle + \beta\,|1\rangle \mapsto \alpha\,|0\rangle - \beta\,|1\rangle$, or $Z\,|x\rangle = (-1)^x\,|x\rangle$, for $x \in \mathbb{F}_2$. Note then that the Z gate is a *phase flip*.

Remarks. (1) The X, Y, and Z are referred to as *Pauli gates*.

(2) Note that $Y = iXZ \equiv XZ$.

(3) $XY = -\mathrm{i}Z \equiv Z$, $YZ = \mathrm{i}X \equiv X$, and $XZ = \mathrm{i}Y \equiv Y$.

(4) Note that all the matrices of these gates are unitary and their own inverses.

(5) Note that

$$Y \equiv \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}.$$

But, if we have *controlled gates* in a circuit, we cannot replace one with the other.

5.2. **Rotations.** We also have *rotations* around the X, Y, and Z angles:

$$R_X(\theta) \overset{\text{def}}{=} \exp\left(-\mathrm{i}\frac{\theta}{2}X\right) = \begin{pmatrix} \cos(\theta/2) & -\mathrm{i}\sin(\theta/2) \\ -\mathrm{i}\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R_Y(\theta) \overset{\text{def}}{=} \exp\left(-\mathrm{i}\frac{\theta}{2}Y\right) = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R_Z(\theta) \overset{\text{def}}{=} \exp\left(-\mathrm{i}\frac{\theta}{2}Z\right) = \begin{pmatrix} \mathrm{e}^{-\mathrm{i}\theta/2} & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\theta/2} \end{pmatrix}$$

Note that if R is one of these, then:

- $R(0) = \mathbb{I}$;
- $R(\theta_1 + \theta_2) = R(\theta_1)R(\theta_2)$;
- $R(2\pi) = -\mathbb{I} \equiv \mathbb{I}$.

Theorem 5.4. *If $U \in \mathrm{U}_2(\mathbb{C})$, then*

$$U = \mathrm{e}^{\mathrm{i}\chi} R_X(\theta_1) R_Y(\theta_2) R_X(\theta_3),$$

for some $\chi, \theta_1, \theta_2, \theta_3 \in \mathbb{R}$. Moreover the X and Y can be replaces by any two of X, Y, and Z.

Corollary 5.5. *If $U \in \mathrm{U}_2(\mathbb{C})$, then*

$$U = \mathrm{e}^{\chi\xi} AXBXC,$$

for some $\chi \in \mathbb{R}$ and $A, B, C \in \mathrm{U}_2(\mathbb{C})$.

Proof. As in Theorem 5.4, let $U = \mathrm{e}^{\mathrm{i}\chi} R_Z(\theta_1) R_Y(\theta_2) R_Z(\theta_3)$, and set

$$\begin{aligned} A &= R_Z(\theta_1) R_Y(\theta_2/2), \\ B &= R_Y(-\theta_2/2) R_Y(-(\theta_1+\theta_3)/2), \\ C &= R_Z((\theta_3-\theta_1)/2). \end{aligned}$$

Then, it is clear that $ABC = \mathbb{I}$. One can also check that

$$\begin{aligned} XBX &= X R_Y\left(-\frac{\theta_2}{2}\right) X X R_Z\left(-\frac{\theta_1+\theta_2}{2}\right) X \\ &= R_Y\left(\frac{\theta_2}{2}\right) R_Z\left(\frac{\theta_1+\theta_3}{2}\right). \end{aligned}$$

Then,

$$\begin{aligned} AXBXC &= R_Z(\theta_1) R_Y\left(\frac{\theta_2}{2}\right) R_Y\left(\frac{\theta_2}{2}\right) R_Z\left(\frac{\theta_1+\theta_3}{2}\right) R_Z\left(\frac{\theta_3-\theta_1}{2}\right) \\ &= R_Z(\theta_1) R_Y(\theta_2) R_Z(\theta_3), \end{aligned}$$

which finishes the proof. □

5.3. **Hadamard, S, T, and Phase Gates.**

Notation 5.6. (1) *Hadamard Gate*: multiplication by

$$H \overset{\text{def}}{=} \frac{\sqrt{2}}{2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

(2) We also have the *S gate* and *T gate*:

$$S \overset{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\pi/2} \end{pmatrix}, \qquad T \overset{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\pi/4} \end{pmatrix}$$

(3) The *phase gate* is given by

$$P(\phi) \overset{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\phi} \end{pmatrix} = \mathrm{e}^{-\mathrm{i}\phi/2} R_Z(\phi).$$

Remark. Note that if we have two gates U_1 and U_2, with $U_1 \equiv U_2$, if they are affecting a single qubit, they can be exchanged without affecting the physical result, as they only add a global phase. On the other hand, if they come with a *controlled gate*, as seen in Section 7.6, then they introduce a relative phase and *cannot* be switched!

Remarks. (1) $H^2 = \mathbb{I}$.

(2) $T^2 = S$ and $S^2 = Z$.

(3) $S = P(\pi/2)$ and $T = P(\pi/4)$.

(4) Note that, with the Hadamard gate we have

$$|0\rangle \mapsto |+\rangle, \qquad |+\rangle \mapsto |0\rangle,$$
$$|1\rangle \mapsto |-\rangle, \qquad |-\rangle \mapsto |1\rangle.$$

So, it is a change of basis between $\{|0\rangle, |1\rangle\}$ and $\{|+\rangle, |-\rangle\}$ and back.

(5) We have that

$$|s\rangle \overset{\text{def}}{=} H^{\otimes n} |0\rangle^{\otimes n} = \sum_{x \in \mathbb{F}_2^n} \frac{1}{2^{n/2}} |x\rangle,$$

and hence it changes $|00\ldots0\rangle$ to one of equal probabilities, i.e., an equal (unbiased) superposition of all computational basis elements.

(6) Note that all the matrices of these gates are their own inverses.

(7) Note that $|+\rangle$ and $|-\rangle$ have the same probabilities for 0 and 1 (half for each), but after applying H (getting $|0\rangle$ and $|1\rangle$ respectively), they do not!

(8) Note that S and T introduce relative phases of $\pi/2$ and $\mathrm{i}/4$.

(9) Note that $R_Z(\pi/2) = \mathrm{e}^{-\mathrm{i}\pi/4} S \equiv S$ and $R_Z(\pi/4) = \mathrm{e}^{-\mathrm{i}\pi/8} \equiv T$.

Theorem 5.7. *If $U \in \mathrm{U}_2(\mathbb{C})$, then*

$$U = \mathrm{e}^{\mathrm{i}\chi} \begin{pmatrix} \cos(\theta/2) & -\mathrm{e}^{\mathrm{i}\lambda} \sin(\theta/2) \\ \mathrm{e}^{\mathrm{i}\phi} \sin(\theta/2) & \mathrm{e}^{\mathrm{i}(\phi+\lambda)} \cos(\theta/2) \end{pmatrix},$$

for some $\chi, \theta, \phi, \lambda \in \mathbb{R}$. (Note that χ only affects the overall phase, so it is irrelevant.)

Remark. Most quantum algorithms start with $|s\rangle = H^{\otimes n} |0\rangle^{\otimes n}$ (equal probabilities) and then amplify the coefficient of the answer. Then, measuring will most likely give you the correct answer.

5.4. **Computations with the Hadamard Gate.** Note that we have

$$H |a\rangle = \frac{1}{\sqrt{2}} |0\rangle + (-1)^a |1\rangle = \frac{1}{\sqrt{2}} \sum_{b \in \mathbb{F}_2} (-1)^{ab} |b\rangle .$$

Then,

$$\begin{aligned} H^{\otimes n} |x\rangle_n &= H^{\otimes n} |x_{n-1} x_{n-2} \cdots x_0\rangle \\ &= (H |x_{n-1}\rangle) \otimes \cdots \otimes (H |x_0\rangle) \\ &= \left(\frac{1}{\sqrt{2^n}} \sum_{y_{n-1} \in \mathbb{F}_2} (-1)^{x_{n-1} y_{n-1}} |y_{n-1}\rangle \right) \otimes \cdots \otimes \left(\frac{1}{\sqrt{2^n}} \sum_{y_0 \in F_2} (-1)^{x_0 y_0} |y_0\rangle \right) \\ &= \frac{1}{\sqrt{2^n}} \sum_{y \in \mathbb{F}_2^n} (-1)^{x \cdot y} |y\rangle_n , \end{aligned}$$

i.e.,

$$H^{\otimes n} |x\rangle_n = \frac{1}{\sqrt{2^n}} \sum_{y \in \mathbb{F}_2^n} (-1)^{x \cdot y} |y\rangle_n , \tag{2}$$

where the operation in $x \cdot y$ is the dot product of $\mathbb{F}_2^n$.

5.5. **Query Gate.**

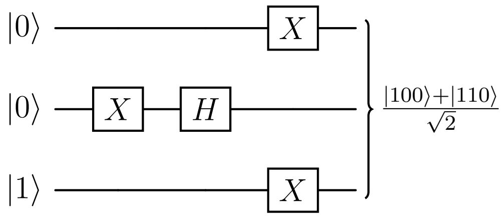
Definition 5.8. Given a function $f : \mathbb{F}_2^n \to \mathbb{F}_2^m$, the *query gate* for f is the defined as

$$U_f(|y\rangle_m |x\rangle_n) \stackrel{\text{def}}{=} |y \oplus f(x)\rangle_m |x\rangle_n .$$

Note that the $2^{n+m} \times 2^{n+m}$ matrix of U_f is a *permutation matrix*, i.e., the columns are permutations of the identity, since the $|y\rangle_m |x\rangle_n$ form a basis. Note that permutation matrices are always unitary. In this case, in fact, we have $U_f = U_f^\dagger$.

6. Quantum Circuits

Circuits apply gates to particular qubits. For instance:

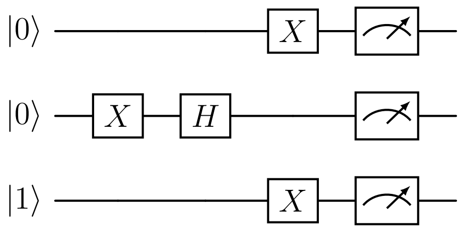


Breaking it down in steps:

$$\begin{aligned}
|001\rangle &\mapsto |011\rangle \\
&\mapsto |0\rangle \otimes \left(\frac{\sqrt{2}}{2}|0\rangle - \frac{\sqrt{2}}{2}|1\rangle\right) \otimes |1\rangle = \frac{\sqrt{2}}{2}|001\rangle + \frac{\sqrt{2}}{2}|011\rangle \\
&\mapsto \frac{\sqrt{2}}{2}|100\rangle + \frac{\sqrt{2}}{2}|110\rangle .
\end{aligned}$$

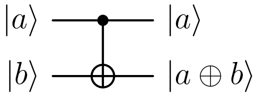
So, the probability that we get $|100\rangle$ or $|110\rangle$ is $1/2$ each, and 0 for every other state.

We can also add measurements to the circuit:



7. Multi-Qubit Gates

7.1. CNOT **Gate.** Here is the CNOT *Gate* (for *controlled not gate*) or *controlled NOT gate*: we have a control qubit and target qubit. If the control qubit is 1, then flip the value of the target bit. The graphical representation is



The dot is the control and the circle is the target. So, if the first qubit is the control and the second is the target, then this takes:

$$\begin{aligned}
|00\rangle &\mapsto |00\rangle\,, \\
|01\rangle &\mapsto |01\rangle\,, \\
|10\rangle &\mapsto |11\rangle\,, \\
|11\rangle &\mapsto |10\rangle\,.
\end{aligned}$$

Or, we can represent:

$$\mathrm{CNOT}\,|x\rangle\,|y\rangle \overset{\text{def}}{=} \begin{cases} |x\rangle\,|y\rangle\,, & \text{if } x = 0; \\ |x\rangle\,|y \oplus 1\rangle\,, & \text{if } x = 1. \end{cases}$$

As a matrix:

$$\mathrm{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Also note that

$$\begin{aligned} \mathrm{CNOT}\,|{+}{+}\rangle &= |{+}{+}\rangle\,, \\ \mathrm{CNOT}\,|{-}{+}\rangle &= |{-}{+}\rangle\,, \\ \mathrm{CNOT}\,|{+}{-}\rangle &= |{-}{-}\rangle\,, \\ \mathrm{CNOT}\,|{-}{-}\rangle &= |{+}{-}\rangle\,, \end{aligned}$$

and hence it is not "controlled" by the second $|\pm\rangle$ basis element: if it is $|+\rangle$, leaves unchanged, if it is $|-\rangle$, flip the sign of the first.

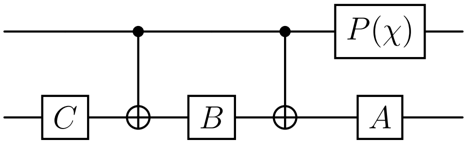
Theorem 7.1. *On n qubits, the set*

$$\mathcal{G} \stackrel{\text{def}}{=} \{1\text{-}\textit{qubit gates on any qubit}\} \cup \{\mathrm{CNOT}\ \textit{on any two qubits}\}$$

generates all gates, i.e., it generates $U_{2^n}(\mathbb{C})$.

Theorem 7.2. *We have that* $\dim_{\mathbb{R}} \mathrm{U}_n(\mathbb{C})$ *is* $2n^2$ *(as an Euclidean space), so* $\dim_{\mathbb{R}} \mathrm{U}_{n^2}(\mathbb{C}) = 2 \cdot (2^n)^2 = 2 \cdot 2^{2n} = 2 \cdot 4^n$.

Note that using Corollary 5.5, we can produce any controlled gate with rotations. More precisely, we have that if $U = \mathrm{e}^{\mathrm{i}\chi} AXBXC$, with $A, B, C \in \mathrm{U}_2(\mathbb{C})$ such that $ABC = 1$, then CU can be constructed as:



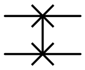
7.2. **Swap Gate.** The *swap gate* SWAP is defined by for $a, b \in \mathbb{F}_2$ as

$$\mathrm{SWAP}\,|ab\rangle \stackrel{\text{def}}{=} |ba\rangle\,.$$

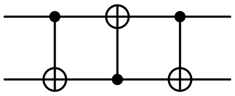
The corresponding matrix is

$$\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

It is represented by



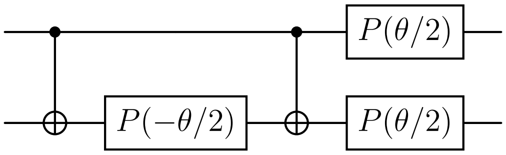
It can be constructed from CNOT gates:



7.3. **Controlled-Phase Gate.** The *controlled-phase gate* $CP(\theta)$, given by

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \mathrm{e}^{\mathrm{i}\theta} \end{pmatrix},$$

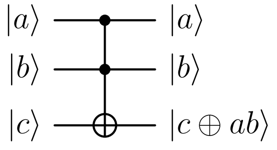
can be constructed as:



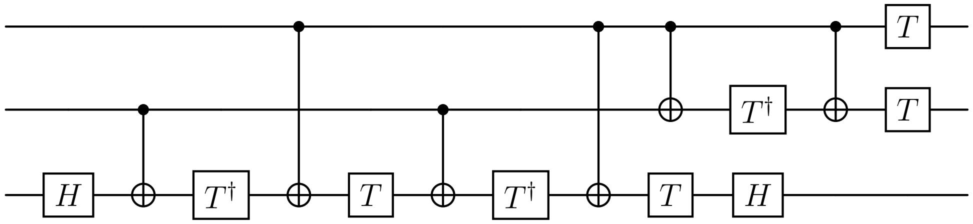
7.4. **Multiple Control Gates.** We can have multiple controls for a gate. In that case, all control qubits must be $|1\rangle$ in order for the gate to be applied to the target qubit. For a gate U on n qubits, the notation for it is $C^{n-1}U$.

Note: These may be hard to create using only one-qubit gates and CNOT! See this Stack exchange post.

7.5. **Toffoli Gate.** The *Toffoli Gate* is C^2X, so it has two controls and one target. We only switch the target, i.e., apply X, when *both* controls are 1.

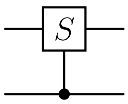


We can construct the Toffoli gate from H, T (and $T^\dagger = T^7 = P(-\pi/4)$), and CNOT gates:



7.6. **Other Controlled Gates.** One can use the CNOT gate to produce other controlled gates: CY, CZ, CS, CH.

Here is a graphic representation:



8. Measuring Singular Qubits

If we have:

$$|\psi_0\rangle = \frac{1}{2}|00\rangle + \frac{1}{4}|01\rangle + \frac{\sqrt{2}}{2}|10\rangle + \frac{\sqrt{3}}{4}|11\rangle$$

and we measure the first qubit to be 1, then the new state is

$$|\psi_1\rangle = c \cdot \left(\frac{\sqrt{2}}{2}|10\rangle + \frac{\sqrt{3}}{4}|11\rangle \right),$$

with

$$c^2\left(\frac{1}{2} + \frac{3}{16}\right) = 1$$

due to probabilities. Hence, $c = 4/\sqrt{11}$ and

$$|\psi_1\rangle = \frac{4}{\sqrt{22}}|10\rangle + \frac{\sqrt{3}}{\sqrt{11}}|11\rangle .$$

9. Elementary Operations

Definition 9.1. The *standard quantum gate set*, which is often used to measure number of "steps" in a quantum circuit are:

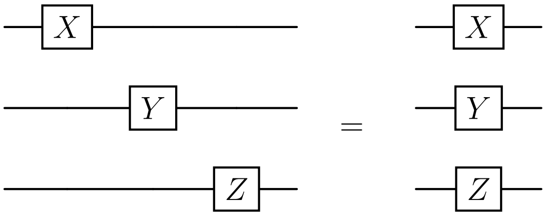
- The single-qubit gates X, Y, Z, H, S, $S^\dagger$, T, $T^\dagger$;
- the $CNOT$ gates;
- single-qubit standard basis measurements.

It can be shown that any unitary gate can be closely approximated by a circuit using these gates.

Definition 9.2. (1) The *size* of a circuit is the number of (elementary) gates it uses. (This is associated with *sequential running time*.)

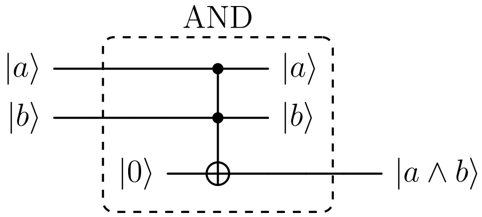
(2) The *depth* of a circuit is the number of *layers* of the circuit, i.e., the number of time steps need to execute all gates. (This is associated with *parallel running time.*)
(3) The circuit's *width* is the maximum number of gates to be executed (in parallel) at single time step. (Narrow usually means that we use few ancillas.)

Note that the depth is not the same as the size/number of gates, since these may run in parallel. For instance, the depth of

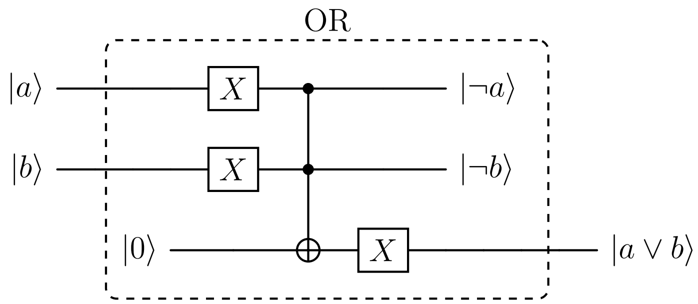


is one, and its width and size are three.

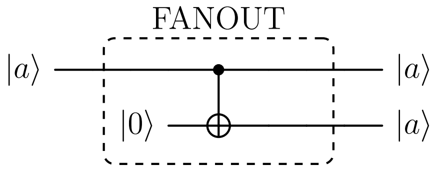
We can use elementary operations to simulate classical logical gates. We already have the NOT gate X. Here is the AND gate:



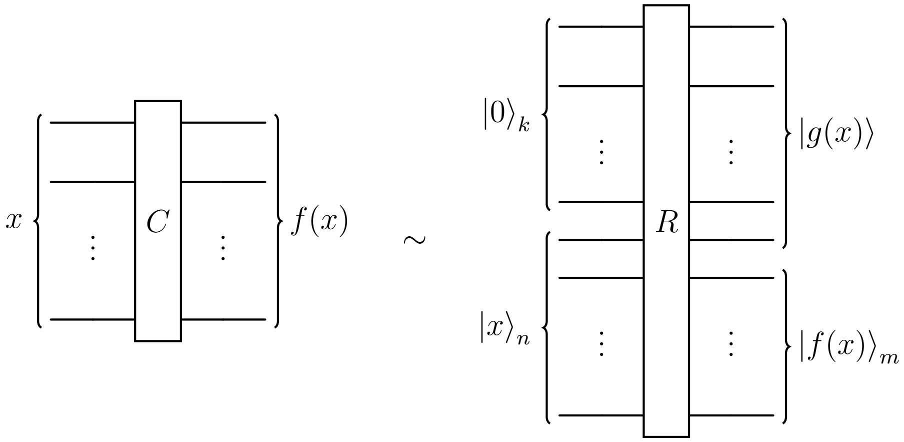
And here is the OR gate:



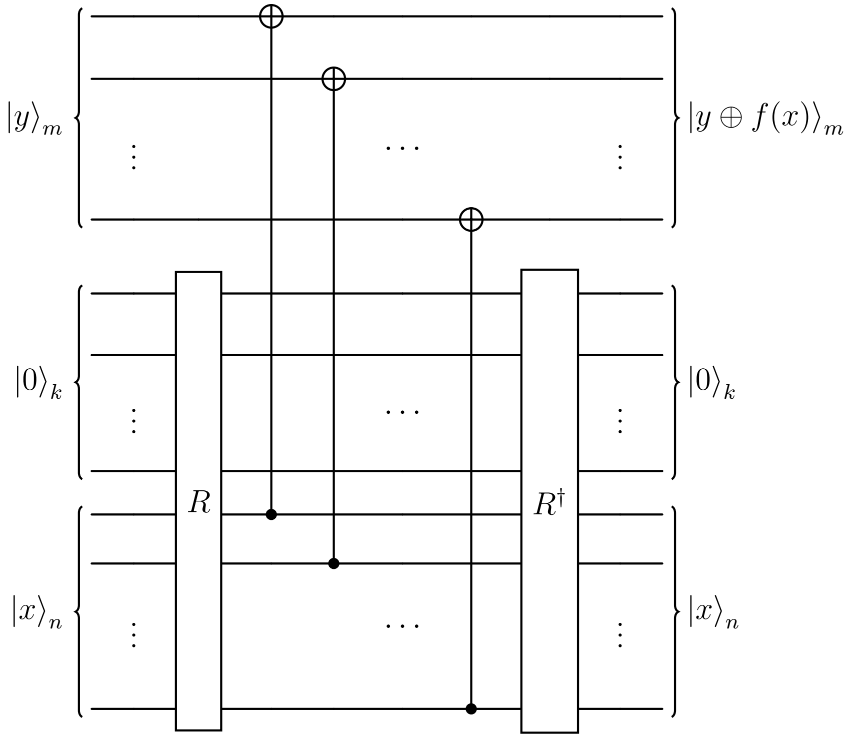
Finally, we have the *FANOUT* gate, which just replicates the input:



Hence, if we have logical gate C that computes $f : \mathbb{F}_2^n \to \mathbb{F}_2^m$ using t elementary logical gates, we can use the approximations above to implement it as a quantum circuit R, with $\mathcal{O}(t)$ elementary quantum gates:



Note that we have the *garbage* $|g(x)\rangle$, which can be problematic if we want to use it as a subroutine in a larger algorithm, as it might break interference patterns. To fix it, we can copy the result $|f(x)\rangle_m$ to a new set of m qubits:

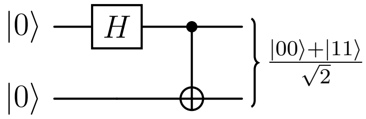


Remark. It's important here that R is *deterministic*, as it might now work when the qubits are in superposition states of the basis.

Note then that for any function f that can be implemented with classical boolean gates, we can implement a query gate U_f.

10. Entanglement

Consider the circuit:



$$\begin{aligned} |00\rangle &\mapsto \frac{\sqrt{2}}{2}\left(|0\rangle + |1\rangle\right) \otimes |0\rangle = \frac{\sqrt{2}}{2}\left(|00\rangle + |10\rangle\right) \\ &\mapsto \frac{\sqrt{2}}{2}\left(|00\rangle + |11\rangle\right). \end{aligned}$$

Hence, if the first qubit is measured as 0, then the second qubit must also be 0, and similarly if it is measured as 1, then the second must also be 1. So, these qubits are *entangled qubits*.

More precisely:

Definition 10.1. A state is *entangled state* if it cannot be factored as tensor products of individual qubits.

Example 10.2. We have that

$$\frac{\sqrt{3}}{2\sqrt{5}}|00\rangle + \frac{1}{2\sqrt{5}}|01\rangle + \frac{\sqrt{3}}{\sqrt{5}}|10\rangle + \frac{1}{\sqrt{5}}|11\rangle = \left(\frac{1}{\sqrt{5}}|0\rangle + \frac{2}{\sqrt{5}}|1\rangle\right) \otimes \left(\frac{\sqrt{3}}{2}|0\rangle + \frac{1}{2}|1\rangle\right)$$

so that state is *not* entangled. On the other hand

$$\frac{\sqrt{2}}{2}\left(|000\rangle + |011\rangle\right)$$

cannot be written as a tensor product, so it is. (Note that if we measure the second qubit, we know the state of the other two.)

Definition 10.3. (1) Qubits are *maximally entangled qubits* if measuring one of the qubits determine the other qubits.

(2) *Bell states* are some examples of maximally entangled qubits:
- $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$;
- $|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$;
- $|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$;
- $|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$.

(3) Qubits are *partially entangled* if measuring one of the qubits affect the probabilities for the other qubits. E.g., consider

$$|\psi\rangle = \frac{\sqrt{3}}{\sqrt{5}}\,|00\rangle + \frac{1}{\sqrt{5}}\,|01\rangle + \frac{1}{2\sqrt{5}}\,|10\rangle + \frac{\sqrt{3}}{2\sqrt{5}}\,|11\rangle\,.$$

If we measure the first qubit as 0, we get

$$|0\rangle \otimes \left(\frac{\sqrt{3}}{2}\,|0\rangle + \frac{1}{2}\,|1\rangle\right),$$

while if we measure the first qubit as 1, we get

$$|1\rangle \otimes \left(\frac{1}{2}\,|0\rangle + \frac{\sqrt{3}}{2}\,|1\rangle\right).$$

Remark. Note that if your algorithm does not entangle qubits, most likely it can be implemented in a classical computer, and a quantum computer would have no advantage.

11. Phase Kickback

Let $b, c \in \mathbb{F}_2$. Then, we have that

$$|b \oplus c\rangle = X^c\,|b\rangle = X^b\,|c\rangle\,.$$

So, if U_f is a query gate for the function $f : \mathbb{F}_2^n \to \mathbb{F}_2$, then $U_f(|b\rangle\,|a\rangle_n) = |b \oplus f(a)\rangle\,|a\rangle_n = (X^{f(a)}\,|b\rangle) \otimes |a\rangle_n$. More generally,

$$U_f(|\psi\rangle\,|a\rangle_n) = (X^{f(a)}\,|\psi\rangle\,|a\rangle_n).$$

In particular, since $X\,|-\rangle = -\,|-\rangle$ (so, $|-\rangle$ is an eigenstate of X), we have

$$\begin{aligned} U_f(|-\rangle\,|a\rangle_n) &= \left(X^{f(a)}\,|-\rangle\right) \oplus |a\rangle_n \\ &= (-1)^{f(a)}\,|-\rangle\,|a\rangle_n\,. \end{aligned}$$

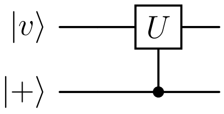
The formula

$$U_f(|-\rangle\,|a\rangle_n) = (-1)^{f(a)}\,|-\rangle\,|a\rangle_n\,, \tag{3}$$

is the phase kickback formula.

On the other hand, note that since $X\left|+\right\rangle = \left|+\right\rangle$, we have $U_f(\left|+\right\rangle \left|a\right\rangle_n) = \left|+\right\rangle \left|a\right\rangle_n$, and so it remains unchanged.

We can also consider:



and suppose the $\left|v\right\rangle$ is an *eigenstate* of U, meaning, $U\left|v\right\rangle = \mathrm{e}^{\mathrm{i}\theta}\left|v\right\rangle$. (Note that, due to normalization, all eigenvalues are of the form $\mathrm{e}^{\mathrm{i}\theta}$.)

We then have

$$\begin{aligned}\left|+\right\rangle \otimes \left|v\right\rangle &= \frac{\sqrt{2}}{2}\left(\left|0\right\rangle \otimes \left|v\right\rangle + \left|1\right\rangle \otimes \left|v\right\rangle\right)\\ &\mapsto \frac{\sqrt{2}}{2}\left(\left|0\right\rangle \otimes \left|v\right\rangle + \left|1\right\rangle \otimes U\left|v\right\rangle\right)\\ &= \frac{\sqrt{2}}{2}\left(\left|0\right\rangle \otimes \left|v\right\rangle + \mathrm{e}^{\mathrm{i}\theta}\left|1\right\rangle \otimes \left|v\right\rangle\right)\\ &= \frac{\sqrt{2}}{2}\left(\left|0\right\rangle + \mathrm{e}^{\mathrm{i}\theta}\left|1\right\rangle\right) \otimes \left|v\right\rangle .\end{aligned}$$

So, $\left|v\right\rangle$ is unchanged (even though it was the target qubit), and a relative phase was applied to the control qubit.

So, if we apply a controlled gate to a target that is an eigenvector of this gate, the phase of the control qubit is changed. This is called *phase kickback*.

12. Superdense Coding

Alice can send Bob two classical bits using only one qubit. Alice and Bob start with a maximally entangled pair of qubits

$$\left|\psi_0\right\rangle = \frac{1}{\sqrt{2}}(\left|00\right\rangle + \left|11\right\rangle).$$

Alice takes the first qubit and Bob the second. Then, when Alice applies the first operation (depending on which two bits she wants to send Bob) and Bob the last two $((H \otimes \mathbb{I}) \circ \text{CNOT})$:

$$|\psi_0\rangle \xmapsto{\mathbb{I}^{\otimes 2}} \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \xmapsto{\text{CNOT}} \frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle\right) = |+\rangle \otimes |0\rangle \xmapsto{H\otimes\mathbb{I}} |00\rangle$$

$$|\psi_0\rangle \xmapsto{X\otimes\mathbb{I}} \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right) \xmapsto{\text{CNOT}} \frac{1}{\sqrt{2}}\left(|11\rangle + |01\rangle\right) = |+\rangle \otimes |1\rangle \xmapsto{H\otimes\mathbb{I}} |01\rangle$$

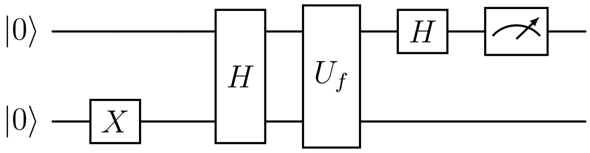
$$|\psi_0\rangle \xmapsto{\mathbb{I}\otimes Z} \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \xmapsto{\text{CNOT}} \frac{1}{\sqrt{2}}\left(|00\rangle - |10\rangle\right) = |-\rangle \otimes |0\rangle \xmapsto{H\otimes\mathbb{I}} |10\rangle$$

$$|\psi_0\rangle \xmapsto{\mathbb{I}\otimes XZ} \frac{1}{\sqrt{2}}\left(|01\rangle - |10\rangle\right) \xmapsto{\text{CNOT}} \frac{1}{\sqrt{2}}\left(|01\rangle - |11\rangle\right) = |-\rangle \otimes |1\rangle \xmapsto{H\otimes\mathbb{I}} |11\rangle$$

13. Deutsch's Algorithm

Problem: Given a function $f : \mathbb{F}^2 \to \mathbb{F}^2$ and an oracle $U_f(|y\rangle\,|x\rangle) = |y + f(x)\rangle\,|x\rangle$, find if $f(0) \oplus f(1)$ is either 0 or 1. Note that if $f(0) \oplus f(1) = 0$, then f is *constant.*

Here is the quantum circuit:



Let's check that it works. First, observe that

$$|0 \oplus a\rangle - |1 \oplus a\rangle = (-1)^a \left(|0\rangle - |1\rangle\right).$$

Then, due to phase kickback (Eq. (3)):

$$\begin{aligned}
\left|0\right\rangle\left|0\right\rangle &\mapsto \left|1\right\rangle\left|0\right\rangle \\
&\mapsto \frac{1}{2}\left(\left|0\right\rangle - \left|1\right\rangle)\right) \otimes \left(\left|0\right\rangle + \left|1\right\rangle)\right) \\
&= \frac{1}{\sqrt{2}}\left(\left|-\right\rangle\left|0\right\rangle + \left|-\right\rangle\left|1\right\rangle\right) \\
&\mapsto \frac{1}{\sqrt{2}}\Big((-1)^{f(0)}\left|-\right\rangle\left|0\right\rangle + (-1)^{f(1)}\left|-\right\rangle\left|1\right\rangle\Big) \\
&= (-1)^{f(0)}\left|-\right\rangle \otimes \frac{1}{\sqrt{2}}\Big(\left|0\right\rangle + (-1)^{f(0)\oplus f(1)}\left|1\right\rangle\Big) \\
&\mapsto (-1)^{f(0)}\left|-\right\rangle \otimes \left|f(0)\oplus f(1)\right\rangle,
\end{aligned}$$

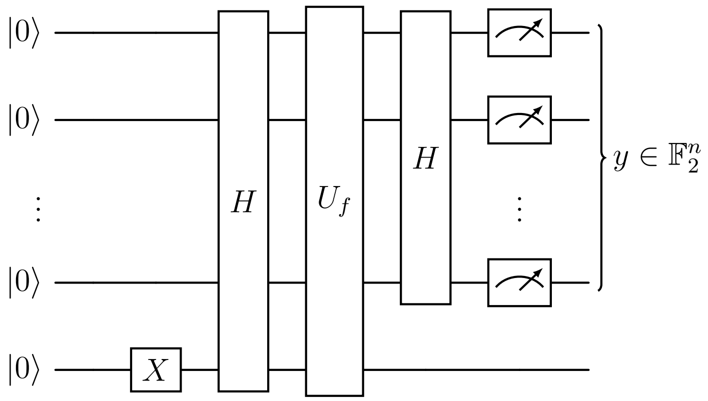
so, we measure $f(0) \oplus f(1)$.

Note that we call U_f only once, but due to "interference": we compute both $f(0)$ and $f(1)$, at the same time, due to the Hadamard gate at the top quibit. We make constructive interference for the correct answer, and destructive interference for the wrong one.

14. Deutsch-Jozsa Algorithm

Now, we will consider $f : \mathbb{F}_2^n \to \mathbb{F}_2$, for $n \geq 1$.

Here is the circuit for the Deutsch-Jozsa algorithm, which just generalizes the previous one:



This circuit can be used to solve a multiple problems.

14.1. **The Deutsch-Jozsa Problem. Problem:** Given $f : \mathbb{F}_2^n \to \mathbb{F}^2$, determine if the function is constant or balanced (i.e., the number of inputs that give 0 and the number of inputs that give 1 are equal). (Note that f might be neither, but we just "don't care" about such functions, i.e., we *assume* that the f is either constant or balanced.)

Using Eq. (2) and phase kickback, we have

$$\begin{aligned}
|0\rangle_n &\mapsto |1\rangle\,|0\rangle_{n-1} \\
&\mapsto |-\rangle \otimes \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} |x\rangle_n \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} |-\rangle\,|x\rangle_n \\
&\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)}\,|-\rangle\,|x\rangle_n \\
&= |-\rangle \otimes \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)}\,|x\rangle_n \\
&\mapsto |-\rangle \otimes \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} \sum_{y \in \mathbb{F}_2^n} (-1)^{f(x)+x\cdot y}\,|y\rangle_n \\
&= |-\rangle \otimes \frac{1}{2^n} \sum_{y \in \mathbb{F}_2^n} \left[\sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+x\cdot y} \right] |y\rangle_n .
\end{aligned}$$

Now,

$$\mathbb{P}(|0\rangle_n) = \left| \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)} \right|^2 = \begin{cases} 1, & \text{if } f \text{ is constant;} \\ 0, & \text{if } f \text{ is balanced.} \end{cases}$$

Therefore, if we measure 0, we have that the function is constant and if we don't, then the function is constant. Hence, with a single query we can answer the questions.

Classically, in the worst case scenario we'd need $2^{n-1} + 1$ queries, but in terms of probability, k *random* queries being equal for a balanced function would have a probability of $1/2^{k-1}$, which gets small quickly. Since quantum computers are not perfect in reality, the advantage might not be that great.

14.2. **The Bernstein-Vazirani Problem. Problem:** Given $f : \mathbb{F}_2^n \to \mathbb{F}_2$ for which we know there is some $s \in \mathbb{F}_2^n$ such that $f(x) = x \cdot z$ for all $x \in \mathbb{F}_2^n$, find z.

The same circuit as the one for Deutsch-Jozsa immediately solves the problem, since

$$\begin{aligned}\frac{1}{2^n}\sum_{y\in\mathbb{F}_2^n}\left[\sum_{x\in\mathbb{F}_2^n}(-1)^{f(x)+x\cdot y}\right]|y\rangle_n &= \frac{1}{2^n}\sum_{y\in\mathbb{F}_2^n}\left[\sum_{x\in\mathbb{F}_2^n}(-1)^{x\cdot z+x\cdot y}\right]|y\rangle_n \\ &= \frac{1}{2^n}\sum_{y\in\mathbb{F}_2^n}\left[\sum_{x\in\mathbb{F}_2^n}(-1)^{x\cdot(z\oplus y)}\right]|y\rangle_n .\end{aligned}$$

Then,

$$\mathbb{P}(|z\rangle_n) = \left|\frac{1}{2^n}\sum_{x\in\mathbb{F}_2^n}1\right|^2 = 1.$$

One can also see that if $y\in\mathbb{F}_2^n\setminus\{(0,0,\dots,0)\}$, then

$$\sum_{x\in\mathbb{F}_2^n}(-1)^{x\cdot y}=0.$$

Indeed, if $y_i\neq 0$, i.e., $y_i=1$, then if

$$(-1)^{x_0y_0+\cdots+0\cdot y_i+\cdots+x_{n-1}y_{n-1}}+(-1)^{x_0y_0+\cdots+1\cdot y_i+\cdots+x_{n-1}y_{n-1}}=0,$$

and the summands above can be paired this way.

Again, we can solve the problem with a single query, while in classic computing, we'd need n.

15. Simon's Algorithm

Problem: Let $f:\mathbb{F}_2^n\to\mathbb{F}_2^m$ such that there is $z\in\mathbb{F}_2^n$ such that

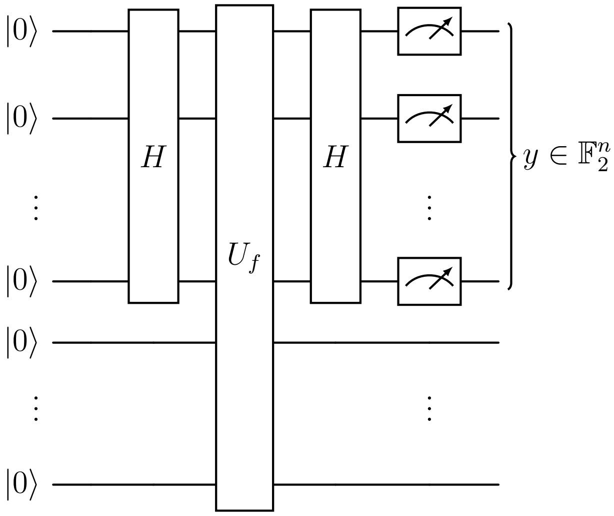
$$f(x)=f(y)\quad\text{if and only if}\quad x=y\text{ or }x\oplus z=y$$

for all $x,y\in\mathbb{F}_2^n$. Find z.

We break it in cases:

(1) **Case 1:** $z = (0, \ldots, 0)$. In this case the condition is that f is one-to-one.
(2) **Case 2:** $z \neq (0, \ldots, 0)$. In this case the condition is that f is two-to-one, as $f(x) = f(x \oplus z)$.

Simon's algorithm consists of running the following circuit several times, followed by some *classical* post processing steps described below.



We then have:

$$\begin{aligned}
|0\rangle_m |0\rangle_n &\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} |0\rangle_m |x\rangle_n \\
&\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{F}_2^n} |f(x)\rangle_m |x\rangle_n \\
&\mapsto \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} |f(x)\rangle_m \otimes \sum_{y \in \mathbb{F}_2^n} (-1)^{x \cdot y} |y\rangle_n \\
&= \frac{1}{2^n} \sum_{y \in \mathbb{F}_2^n} \left[\sum_{x \in \mathbb{F}_2^n} (-1)^{x \cdot y} |f(x)\rangle_m \right] \otimes |y\rangle_n .
\end{aligned}$$

Therefore, in measuring the last n-qubits, we have

$$\begin{aligned}
\mathbb{P}(|y\rangle_n) &= \left\| \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} (-1)^{x\cdot y} \, |f(x)\rangle_m \right\|^2 \\
&= \left\| \frac{1}{2^n} \sum_{t \in \text{range}(f)} \left[\sum_{x \in f^{-1}(t)} (-1)^{x\cdot y} \right] |t\rangle_m \right\|^2 \\
&= \frac{1}{2^{2n}} \sum_{t \in \text{range}(f)} \left| \sum_{x \in f^{-1}(t)} (-1)^{x\cdot y} \right|^2
\end{aligned}$$

Hence, in Case 1, when f is one-to-one we have

$$\left| \sum_{x \in f^{-1}(t)} (-1)^{x\cdot y} \right|^2 = 1,$$

since the sum has a single term. So,

$$\mathbb{P}(|y\rangle_n) = \frac{1}{2^{2n}} \cdot 2^n = \frac{1}{2^n}$$

for every $y \in \mathbb{F}_2^n$, meaning the every single y has the exact same probability.

In Case 2, when $z \neq (0, \ldots, 0)$, we have that $f^{-1}(t) = \{w, w \oplus z\}$ (for some w such that $f(w) = t$). Then,

$$\begin{aligned}
\left| \sum_{x \in f^{-1}(t)} (-1)^{x\cdot y} \right|^2 &= \left| (-1)^{w\cdot y} + (-1)^{(w \oplus z)\cdot y} \right|^2 \\
&= |1 + (-1)^{z\cdot y}| \\
&= \begin{cases} 4, & \text{if } z \cdot y = 0; \\ 0, & \text{if } z \cdot y = 1. \end{cases}
\end{aligned}$$

Hence, in this case, since range(f) has 2^{n-1} elements, we have

$$\mathbb{P}(|y\rangle_n) = \begin{cases} \frac{1}{2^{n-1}}, & \text{if } z \cdot y = 0; \\ 0, & \text{if } z \cdot y = 1. \end{cases}$$

Now, suppose that we run this circuit $k \overset{\text{def}}{=} n + r$ times (for some r), and measure $y^{(1)}, y^{(2)}, \ldots y^{(k)}$, and let

$$y^{(i)} = y^{(i)}_{n-1} y^{(i)}_{n-2} \cdots y^{(i)}_1 y^{(i)}_0 ,$$

and let

$$M \overset{\text{def}}{=} \begin{bmatrix} y^{(1)}_{n-1} & y^{(1)}_{n-2} & \cdots & y^{(1)}_0 \\ y^{(2)}_{n-1} & y^{(2)}_{n-2} & \cdots & y^{(2)}_0 \\ \vdots & \vdots & \ddots & \vdots \\ y^{(k)}_{n-1} & y^{(k)}_{n-2} & \cdots & y^{(k)}_0 \end{bmatrix} \in M_{k,n}(\mathbb{F}_2)$$

Then,

$$M \cdot \begin{bmatrix} z_{n-1} \\ z_{n-2} \\ \vdots \\ z_0 \end{bmatrix} = \begin{bmatrix} y^{(1)} \cdot z \\ y^{(2)} \cdot z \\ \vdots \\ y^{(k)} \cdot z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} .$$

This is clearly true in Case 1, where $z = \vec{0}$, but also true in Case 2, due to the probabilities. So, the desired z is in the kernel of M, which we can find. We will have that this kernel will be $\{\vec{0}, z\}$ with probability greater than $1 - 2^{-r}$.

One can show that every classical algorithm to solve Simon's problem requires an *exponential* number of queries, while Simon's algorithm requires a *linear* number.

16. Grover's Algorithm

Grover's algorithm is an *unstructured search*, meaning, no assumption on the data that might help with searching.

Problem statement: given a list $[x_0, x_1, \ldots, x_{N-1}]$ that we can query (i.e., given j we can read x_j from the list) and some y, find if there is j_0 such that $x_{j_0} = y$ and output j_0 if so.

Traditionally, the search is $\mathcal{O}(N)$, with expected number of queries and comparisons $(N+1)/2$.

Assumptions: Assume $N = 2^n$ (or pad the list) and that y is *guaranteed* to be in the list.

Recast: We have the binary representation:

$$\{0, 1, 2, \ldots, 2^n - 1\} \to \mathbb{F}_2^n$$

and so we can produce a function $f : \mathbb{F}_2^n \to \mathbb{F}_2$, where the domain corresponds to the binary representation of the index in the list, and the output is a boolean, such that:

$$f(j) = \begin{cases} 1, & \text{if } x_j = y; \\ 0, & \text{otherwise.} \end{cases}$$

So, we need to find j such that $f(j) = 1$.

Last assumption: We have access to an *oracle* (a quantum circuit) U_f on n qubits such that:

$$U_f \left| j \right\rangle = (-1)^{f(j)} \left| j \right\rangle = \begin{cases} -\left| j \right\rangle, & \text{if } f(j) = 1; \\ \left| j \right\rangle, & \text{otherwise.} \end{cases}$$

Hence, U_f allows us to check if j is the index for y in the list by checking for a phase change. This is equivalent to saying that we can query the list. (*Question:* Can we create such circuit in practice?) Depending on the nature of the objects in the list, this oracle can be quite difficult to implement.

Define

$$U_S \overset{\text{def}}{=} H^{\otimes n} \circ X^{\otimes n} \circ C^{n-1}Z \circ X^{\otimes n} \circ H^{\otimes n}.$$

Note that, for $x \in \mathbb{F}_2^n$, we have

$$C^{n-1}Z \left| x \right\rangle = \begin{cases} -\left| x \right\rangle, & \text{if } x = (1, 1, \ldots, 1); \\ \left| x \right\rangle, & \text{otherwise.} \end{cases}$$

Remark. Synthesizing the $C^{n-1}Z$ gate can be quite difficult in practice. We need to find a way to do it with only 1-qubit gates and CNOT.

Then, we have:

$$\begin{aligned} U_S \left|s\right\rangle &= H^{\otimes n} \circ X^{\otimes n} \circ C^{n-1}Z \circ X^{\otimes n} \circ H^{\otimes n} \circ H^{\otimes n} \left|0\right\rangle^{\otimes n} \\ &= H^{\otimes n} \circ X^{\otimes n} \circ C^{n-1}Z \circ X^{\otimes n} \left|0\right\rangle^{\otimes n} \\ &= H^{\otimes n} \circ X^{\otimes n} \circ C^{n-1}Z \circ \left|1\right\rangle^{\otimes n} \\ &= -H^{\otimes n} \circ X^{\otimes n} \left|1\right\rangle^{\otimes n} \\ &= -H^{\otimes n} \left|0\right\rangle^{\otimes n} \\ &= -\left|s\right\rangle . \end{aligned}$$

Note: If $\left|\psi\right\rangle$ is such that $\langle \psi \mid s\rangle = 0$, then $U_S \left|\psi\right\rangle = \left|\psi\right\rangle$.

Proof. Since $H^{\otimes n}$ is unitary, we have that

$$0 = \langle \psi \mid s\rangle = \langle H^{\otimes n} \left|\psi\right\rangle \mid H^{\otimes n} \left|s\right\rangle\rangle = \langle H^{\otimes n} \left|\psi\right\rangle \mid H^{\otimes n} H^{\otimes n} \left|0\right\rangle^{\otimes n}\rangle = \langle H^{\otimes n} \left|\psi\right\rangle \mid \left|0\right\rangle^{\otimes n}\rangle,$$

and hence the component of $H^{\otimes n} \left|\psi\right\rangle$ in $\left|0\right\rangle^{\otimes n}$ is 0.

So, the component of $X^{\otimes n} \circ H^{\otimes n} \left|\psi\right\rangle$ in $\left|1\right\rangle^{\otimes n}$ is 0, which implies that

$$C^{n-1}Z \circ X^{\otimes n} \circ H^{\otimes n} \left|\psi\right\rangle = X^{\otimes n} \circ H^{\otimes n} \left|\psi\right\rangle .$$

Therefore,

$$U_S \left|\psi\right\rangle = H^{\otimes n} \circ X^{\otimes n} \circ X^{\otimes n} \circ H^{\otimes n} \left|\psi\right\rangle = \left|\psi\right\rangle .$$

□

Summary:

$$\text{target state: } |j_0\rangle \qquad U_f\,|j\rangle = \begin{cases} -\,|j\rangle\,, & \text{if } j = j_0; \\ \,|j\rangle\,, & \text{otherwise (or if } \langle j \mid j_0\rangle = 0). \end{cases}$$

$$\text{initial state: } |s\rangle \qquad U_S\,|\psi\rangle = \begin{cases} -\,|\psi\rangle\,, & \text{if } |\psi\rangle = |s\rangle; \\ \,|\psi\rangle\,, & \text{if } \langle \psi \mid s\rangle = 0. \end{cases}$$

Remark. Since the operators U_f and U_S negate one non-zero vector, and fix all vectors perpendicular to this one, they are *reflections* on the orthogonal space to the span of the non-zero vector. (Just think in terms of orthonormal bases.)

We define *Grover's oracle* as $G = -U_S U_f$.

Theorem 16.1 (Grover, 1996)**.** *Let k be a positive integer and let $|\psi_k\rangle \overset{\text{def}}{=} G^k\,|s\rangle$. Then, when measuring, we have*

$$\mathbb{P}\left(j_0 \mid \psi_k\right) = \left|\langle j_0 \mid \psi_k\rangle\right|^2 = \sin^2\left((2k+1)\arcsin\left(2^{-n/2}\right)\right).$$

Corollary 16.2. *If we let*

$$k \overset{\text{def}}{=} \left\lfloor \frac{\pi}{4\arcsin(2^{-n/2})} - \frac{1}{2} \right\rceil \approx \frac{\pi}{4}2^{n/2} = \frac{\pi}{4}\sqrt{N},$$

(where $\lfloor x \rceil$ is the closest integer to x) then

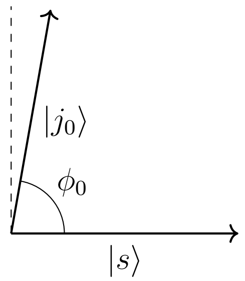
$$\mathbb{P}\left(j_0 \mid \psi_k\right) = 1 - \mathcal{O}\left(\frac{1}{N}\right).$$

Takeaway: After about $\sqrt{N}$ queries, there is a very high probability we will get j_0 when measuring.

Proof of Grover's Theorem. Note that

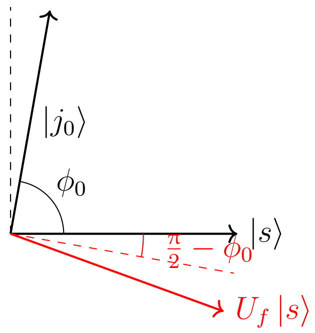
$$\langle j_0 \mid s\rangle = \frac{1}{2^{n/2}} \sum_{j \in \mathbb{F}_2^n} \langle j_0 \mid j\rangle = \frac{1}{2^{n/2}}.$$

Since this inner product is small, the vectors are almost perpendicular. Let ϕ_0 be the angle between them.

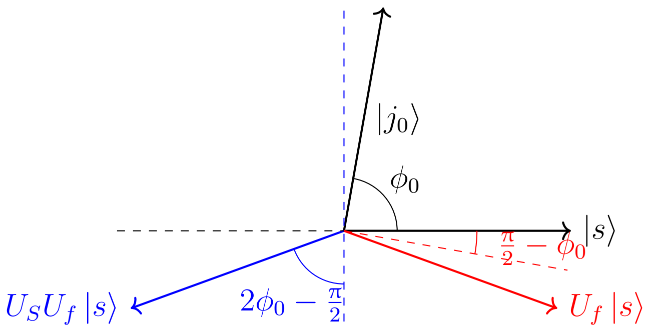


Then, by the inner product above, we have that $\phi_0 = \arccos(2^{-n/2})$.

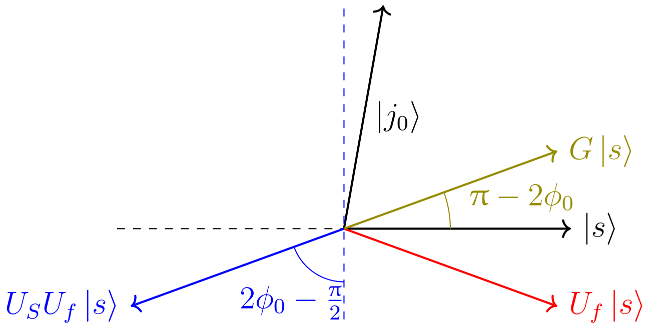
Note that U_S and U_f are *reflections*, so G is a rotation by some angle α. In the plane containing $|s\rangle$ and $|j_0\rangle$, U_f reflects on the line perpendicular to $|j_0\rangle$:



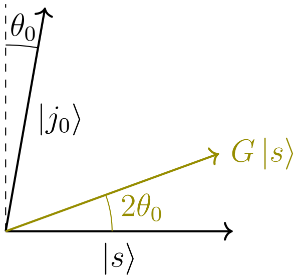
Similarly, U_S reflects on the line perpendicular to $|s\rangle$.



Hence, we have:



Hence, G is a rotation of $\alpha = \pi - 2\phi_0$. Since ϕ_0 is close to $\pi/2$, we have that α is close to 0. So, we have:



Therefore, the angle for $|\psi_k\rangle \overset{\text{def}}{=} G^k |s\rangle$ is $2k\theta_0$. Hence, the angle between $|\psi_k\rangle$ and $|j_0\rangle$ is $\phi_0 - 2k\theta_0 = (2k+1)\phi_0 - k\pi$. Noting that

- $\phi_0 = \arccos(2^{-n/2})$,
- $\cos(x + \pi/2) = -\sin(x)$, and
- $\pi/2 - \arccos(\pi/x) = \arcsin(x)$,

we have

$$\begin{aligned} |\langle j_0 \mid \psi_k \rangle|^2 &= \cos^2((2k+1)\phi_0 - k\pi) \\ &= \cos^2(-(2k+1)(\pi/2 - \phi_0) + \pi/2) \\ &= \sin^2((2k+1)\arcsin(2^{-n/2})). \end{aligned}$$

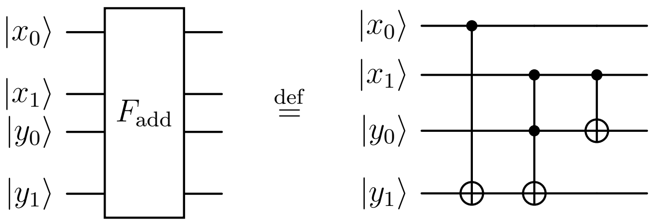
□

17. Application of Grover's Algorithm: Solving Pseudo-Boolean Equations

We illustrate Grover's algorithm with a simple example. Let $F : \{0,1\}^2 \to \mathbb{Z}$ given by $F(x_0, x_1) = 2x_0 + x_1$. Our goal is to solve $F(x_0, x_1) = 2$. Note that the image is $\{0,1,2,3\}$, so we can actually see F as $F : \mathbb{Z}/4\mathbb{Z} \to \mathbb{Z}/4\mathbb{Z}$, where we use the binary representation of the input to get (x_0, x_1), i.e., $2x_1 + x_0 \mapsto (x_0, x_1)$.

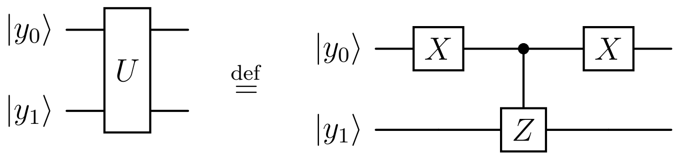
17.1. The Maker Circuit.

The first idea is to create a circuit such that $|x\rangle_2 |0\rangle_2 \mapsto |x\rangle_2 |F(x)\rangle$, or, more generally, $|x\rangle_2 |y\rangle_2 \mapsto |x\rangle_2 |x + y \mod 4\rangle$. The following circuit accomplishes that:



The first CNOT gate adds $2x_0$ to y_1, the second takes care of "carrying-over" when $x_1 = y_1 = 1$, while the last add x_1 to y_0. Let's refer to this circuit as the F_{adder} gate.

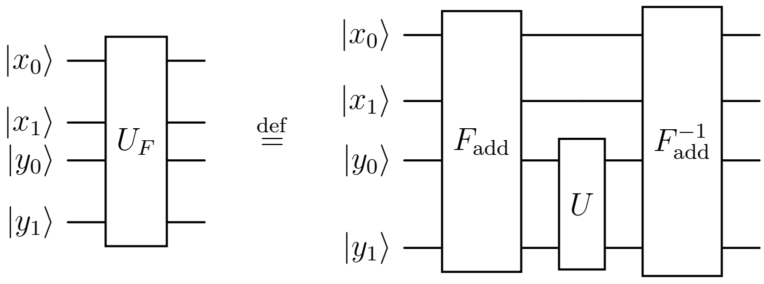
Now, we need an Oracle U that can determine if $F(x_0, x_1) = 2$. So, the "target state" is 2, associated to $|10\rangle$, and we want U so that $U|10\rangle = -|10\rangle$ and $U|y_1y_0\rangle = |y_1y_0\rangle$ if $(y_0, y_1) \neq (0, 1)$. The following circuit accomplishes that step:



Indeed:

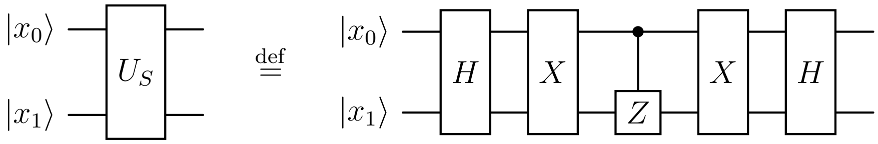
$$\begin{array}{rrrr}
|00\rangle \mapsto |10\rangle \mapsto & |10\rangle \mapsto & |00\rangle \\
|01\rangle \mapsto |00\rangle \mapsto & |00\rangle \mapsto & |01\rangle \\
|10\rangle \mapsto |11\rangle \mapsto & -|11\rangle \mapsto & -|10\rangle \\
|11\rangle \mapsto |01\rangle \mapsto & |01\rangle \mapsto & |11\rangle
\end{array}$$

So, we implement is as:



So, $U_F|x_1x_0\rangle|00\rangle = -|x_1x_0\rangle|00\rangle$ if $F(x_0, x_1) = 2$, and 0 otherwise, as needed for Grover's algorithm.

17.2. **Diffuser Circuit.** We now simply create the gate U_S as in the description the algorithm:

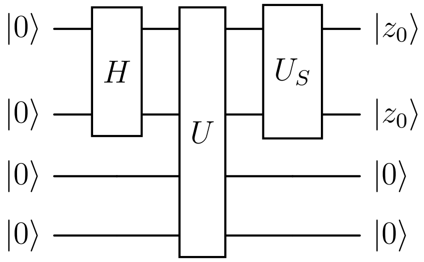


17.3. **Full Grover Circuit.** The number of iterations of $G = -U_S U$ we need is

$$k = \left\lfloor \frac{\pi}{4\arcsin(1/2)} - 1/2 \right\rceil = \lfloor 1 \rceil = 1,$$

so, only one round.

Therefore, we have that the full Grover circuit is given by:



After running, $|z_1 z_0\rangle$, associated to the input (z_0, z_1), should have a high probability of being $|01\rangle$.

18. Quantum Fourier Transform

(More of a quantum “subroutine”.)

Lemma 18.1. *Let $\zeta_N \overset{\text{def}}{=} \mathrm{e}^{2\pi\mathrm{i}/N}$, with $N \geq 2$, and $a \in \mathbb{R}$. Then we have*

$$\sum_{b=0}^{N-1} \zeta_N^{ab} = \begin{cases} N, & \text{if } a \in \mathbb{Z} \text{ and } a \equiv 0 \pmod N; \\ 0, & \text{if } a \in \mathbb{Z} \text{ and } a \not\equiv 0 \pmod N; \\ \frac{\left(\zeta_N^a\right)^N - 1}{\zeta_N - 1}, & \text{if } a \notin \mathbb{Z}. \end{cases}$$

Proof. We have that $\zeta_N^a = 1$ if and only if $a \in \mathbb{Z}$ and $a \equiv 0 \pmod N$, in which case the result is trivial. So, suppose that $\zeta_N^a \neq 1$. Then:

$$\sum_{b=0}^{N-1} \left(\zeta_N^a\right)^b = \frac{\zeta_N^{aN} - 1}{\zeta_N^a - 1}.$$

If $a \in \mathbb{Z}$, then $\left(\zeta_N^a\right)^N = \left(\zeta_N^N\right)^a = 1$. □

18.1. Discrete Fourier Transform.

Definition 18.2. Let $f : \mathbb{Z}/N\mathbb{Z} \to \mathbb{C}$ and $\zeta \overset{\text{def}}{=} \mathrm{e}^{2\pi\mathrm{i}/N}$. Then the *Discrete Fourier Transform (DFT)* is defined as

$$\mathcal{F}(f)(z) \overset{\text{def}}{=} \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \zeta_N^{xz} f(x).$$

It decomposes f into periodic functions.

Proposition 18.3. *We have:*

(1) *$\mathcal{F}$ is linear.*

(2) Parseval identity*: if*

$$\langle f_1 \mid f_2 \rangle \overset{\text{def}}{=} \sum_{x=0}^{N-1} \overline{f}_1(x) f_2(x),$$

then

$$\langle \mathcal{F}(f_1) \mid \mathcal{F}(f_2) \rangle = \langle f_1 \mid f_2 \rangle,$$

i.e., $\mathcal{F}$ is unitary.

(3) Let $k \in \mathbb{Z}/N\mathbb{Z}$ and define the translation

$$T_k(f)(x) \overset{\text{def}}{=} f(x-k).$$

Then,

$$T_k(\mathcal{F}(f))(z) = \zeta_N^{kz} \mathcal{F}(T_k(f))(z).$$

Proof. Let $\zeta_k \overset{\text{def}}{=} \mathrm{e}^{2\pi\mathrm{i}/k}$. For Parseval's identity, using Theorem 18.1 on the previous page we have:

$$\begin{aligned}
\langle \mathcal{F}(f_1) \mid \mathcal{F}(f_2) \rangle &= \sum_{x=0}^{N-1} \left[\frac{1}{\sqrt{N}} \sum_{x_1=0}^{N-1} \zeta_N^{-x_1 x} \overline{f}_1(x_1) \right] \cdot \left[\frac{1}{\sqrt{N}} \sum_{x_2=0}^{N-1} \zeta_N^{x_2 x} \overline{f}_2(x_2) \right] \\
&= \frac{1}{N} \sum_{x=0}^{N-1} \sum_{x_1,x_2=0}^{N-1} \zeta_N^{x(x_2-x_1)} \overline{f}_1(x_1) f_2(x_2) \\
&= \frac{1}{N} \sum_{x=0}^{N-1} \sum_{x_1=0}^{N-1} \overline{f}_1(x_1) f_2(x_1) \\
&= \frac{1}{N} N \sum_{x_1=0}^{N-1} \overline{f}_1(x_1) f_2(x_1) \\
&= \langle f_1 \mid f_2 \rangle.
\end{aligned}$$

□

18.2. **Quantum Fourier Transform.**

Definition 18.4. Let $N \overset{\text{def}}{=} 2^n$, $\zeta_N \overset{\text{def}}{=} \mathrm{e}^{2\pi\mathrm{i}/N}$ and

$$|\psi\rangle = \sum_{x=0}^{N-1} \psi(x) \left| x \right\rangle_n, \quad \psi : \mathbb{Z}/N\mathbb{Z} \to \mathbb{C}, \ \|\psi\| = 1.$$

Then,

$$\begin{aligned}
\mathrm{QFT} \left| \psi \right\rangle &\overset{\text{def}}{=} \sum_{z=0}^{2^n-1} \mathcal{F}(\psi)(z) \left| z \right\rangle_n \\
&= \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \left[\sum_{x=0}^{2^n-1} \zeta_{2^n}^{xz} \psi(x) \right] \left| z \right\rangle_n \\
&= \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} \psi(x) \left[\sum_{z=0}^{2^n-1} \zeta_{2^n}^{xz} \left| z \right\rangle_n \right].
\end{aligned}$$

In particular,

$$\mathrm{QFT}\,|x\rangle_n = \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \zeta_{2^n}^{xz}\,|z\rangle_n$$

Also note that

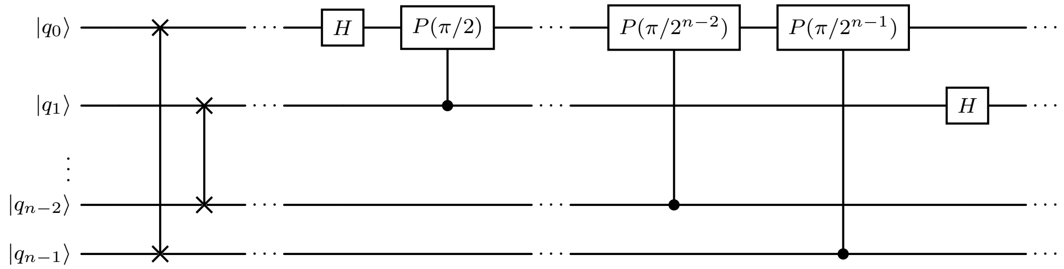
$$\mathrm{QFT}^{-1}\,|x\rangle_n = \mathrm{QFT}^{\dagger}\,|x\rangle_n = \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \zeta_{2^n}^{-xz}\,|z\rangle_n$$

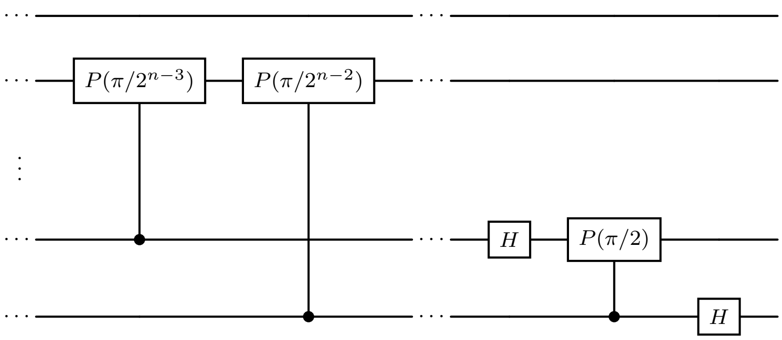
Indeed, using Lemma 18.1 again:

$$\begin{aligned}\mathrm{QFT}^{-1}\,\mathrm{QFT}\,|x\rangle_n &= \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \zeta_{2^n}^{xz} \left[\frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} \zeta_{2^n}^{-yz}\,|y\rangle_n\right] \\ &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \left[\sum_{z=0}^{2^n-1} \zeta_{2^n}^{z(x-y)}\right] |y\rangle_n \\ &= |x\rangle_n\,.\end{aligned}$$

Remark. One can compute $\mathcal{F}(f)$ from f using the fast Fourier transform in $\mathcal{O}(\log_2(N)N)$ time.

Fact: There are quantum circuits for the quantum Fourier transform using $\mathcal{O}(n^2) = \mathcal{O}((\log_2(N))^2)$ gates, using only 1-qubit gates and CNOT gates. Here it is, letting $R_k \overset{\text{def}}{=} P(2\pi/2^k)$:



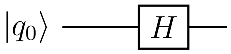


Note that the initial swaps are done to invert all qubits.

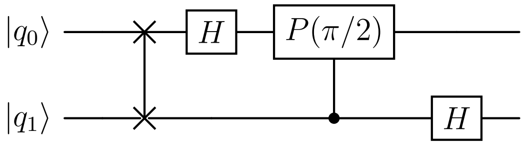
This implementation has no ancillas (auxiliary qubits), so it takes “small space”, but there are more modern implementations that are “shallower”, meaning lower depth, and therefore are faster.

Definition 18.5. *Ancillas* (or ancilla qubits) are extra qubits used in the quantum circuit.

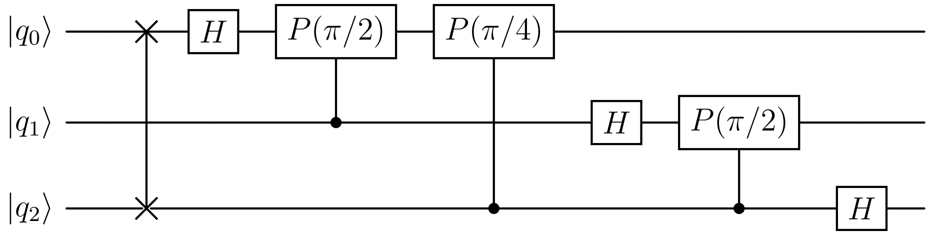
Example 18.6. Here are some examples. For one qubit, the QFT is



For two qubits:



For three:



18.3. **Application: Adder Circuit.** We will need the following (*non-standard*) definition:

Definition 18.7. We define the $\hat{P}$ *gate*:

$$\hat{P}(k)\left|x\right\rangle_n \overset{\text{def}}{=} \mathrm{e}^{2\pi \mathrm{i} xk/2^n}\left|x\right\rangle_n.$$

Thus, ordering the basis as $0 = |00\cdots00\rangle, 1 = |00\cdots01\rangle, 2 = |00\cdots10\rangle, \ldots$ we have the following $2^n \times 2^n$ matrix representation:

$$\hat{P}(k) = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \mathrm{e}^{2\pi \mathrm{i}k/2^n} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \mathrm{e}^{4\pi \mathrm{i}k/2^n} & 0 & \cdots & 0 \\ 0 & 0 & 0 & \mathrm{e}^{6\pi \mathrm{i}k/2^n} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathrm{e}^{\pi \mathrm{i}k} \end{pmatrix}$$

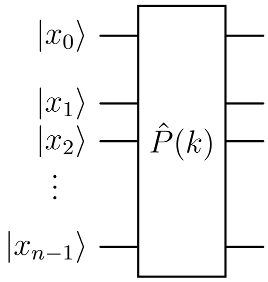
Remembering that the phase gate

$$P(\theta) = \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\theta} \end{pmatrix}.$$

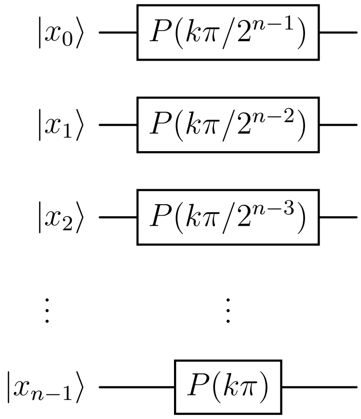
we have

$$\hat{P}(k) = \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\pi \mathrm{i}k} \end{pmatrix} = P(\pi k) = \mathrm{e}^{-\pi \mathrm{i}/2} R_Z(\pi k/2).$$

and hence, we can easily implement the $\hat{P}$ gates. The circuit



which, as a multi-qubit gate, is the same as



Proof. Let

$$x = x_0 + x_1 \cdot 2 + x_2 \cdot 2^2 + \cdots + x_{n-1} \cdot 2^{n-1} = \sum_{r=0}^{n-1} x_r 2^r.$$

Then,

$$\exp\left(\frac{2\pi \mathrm{i}}{2^n} xk\right) = \exp\left(k\pi\mathrm{i} \sum_{r=0}^{n-1} \frac{1}{2^{n-r-1}} x_r\right) \tag{4}$$

Also, note that for a qubit x_r, we have

$$P(\theta)\,|x_r\rangle = \mathrm{e}^{\mathrm{i}\theta x_r}\,|x_r\rangle\,.$$

So,

$$P\left(\frac{k\pi}{2^{n-1-r}}\right)|x_r\rangle = \exp\left(\frac{k\pi\mathrm{i}}{2^{n-1-r}} x_r\right)|x_r\rangle\,.$$

Therefore:

$$\begin{aligned}
\bigotimes_{r=0}^{n-1} P\left(\frac{k\pi}{2^{n-1-r}}\right)|x_r\rangle &= \bigotimes_{r=0}^{n-1} \exp\left(\frac{k\pi\mathrm{i}}{2^{n-1-r}}x_r\right)|x_r\rangle \\
&= \left[\prod_{r=0}^{n-1} \exp\left(\frac{k\pi\mathrm{i}}{2^{n-1-r}}x_r\right)\right] \cdot \bigotimes_{r=0}^{n-1} |x_r\rangle_n \\
&= \left[\exp\left(k\pi\mathrm{i}\sum_{r=0}^{n-1} \frac{1}{2^{n-1-r}}x_r\right)\right]|x\rangle_n \\
&= \exp\left(\frac{2\pi\mathrm{i}}{2^n}xk\right)|x\rangle_n \\
&= P(k)\,|x\rangle_n\,.
\end{aligned}$$

So, indeed the last circuit gives the $\hat{P}(k)$ gate. □

Definition 18.8. For $k \in \mathbb{Z}/2^n\mathbb{Z}$, define the *plus k adder*

$$A_k\,|x\rangle_n \overset{\text{def}}{=} |x+k\rangle_n\,.$$

(Note that the sum $x+k$ is modulo 2^n.)

We have:

$$A_k\,\mathrm{QFT}^\dagger\,|x\rangle_n = \frac{1}{2^{n/2}}\sum_{z=0}^{2^n-1} \zeta_{2^n}^{-xz}\,|z+k\rangle_n \tag{5}$$

$$= \frac{1}{2^{n/2}}\sum_{z=0}^{2^n-1} \zeta_{2^n}^{-x(z-k)}\,|z\rangle_n \tag{6}$$

$$= \frac{1}{2^{n/2}}\zeta_{2^n}^{xk}\sum_{z=0}^{2^n-1} \zeta_{2^n}^{-xz}\,|z\rangle_n \tag{7}$$

$$= \zeta_{2^n}^{xk}\,\mathrm{QFT}^\dagger\,|x\rangle_n\,, \tag{8}$$

i.e.,

$$(\mathrm{QFT}\circ A_k \circ \mathrm{QFT}^\dagger)\,|x\rangle_n = \zeta_{2^n}^{xk}\,|x\rangle_n = \hat{P}(k)\,|x\rangle_n\,, \tag{9}$$

or

$$A_k = \mathrm{QFT}^\dagger \circ \hat{P}(k) \circ \mathrm{QFT}\,. \tag{10}$$

(So, *shift* is turned in to *phase*.)

So, using Eq. (10), and assuming that QFT can be implemented, we can also implement A_k.

19. Quantum Phase Estimation

19.1. **Fejér States.** What happens if we replace $k \in \mathbb{Z}$ in $A(k)$ by some $k \in \mathbb{R}$?

Definition 19.1. For n bits and $k \in \mathbb{R}$, define $|k\rangle_F$, the *k-th Fejér state* as

$$\begin{aligned} |k\rangle_F &\overset{\text{def}}{=} \mathrm{QFT}^\dagger \circ \hat{P}(k) \circ \mathrm{QFT}\,|0\rangle_n \\ &= \sum_{z=0}^{2^n-1} \exp\left(\pi \mathrm{i}(1-1/2^n)(k-z)\right) \frac{\sin(\pi(k-z))}{2^n \sin(\pi(k-z)/2^n)} \,|z\rangle_n \,. \end{aligned}$$

Then, for $x \in \{0, 1, \ldots, 2^n - 1\}$, we have

$$\mathbb{P}\left(x \mid |k\rangle_F\right) = \left|\langle x \mid k\rangle_F\right|^2 = \frac{\sin^2(\pi(k-x))}{4^n \sin^2(\pi(k-x)/2^n)},$$

the *Fejér kernel.*

Figure 2 on the facing page shows the graph of the Fejér kernel

$$\frac{\sin^2(\pi(k-x))}{4^n \sin^2(\pi(k-x)/2^n)}$$

for $n = 5$ and $k = 17.3$.

If we let $k = x_0 + r$, where $x_0 = \lfloor k \rceil$, so $r \in [-1/2, 1/2]$, then

$$\mathbb{P}\left(x_0 \mid |k\rangle_F\right) = \frac{\sin^2(\pi r)}{4^n \sin^2(\pi r/2^n)} \geq \frac{4}{\pi^2} \approx 0.41,$$

and in fact

$$\mathbb{P}\left(\lfloor x \rfloor \mid |k\rangle_F\right) + \mathbb{P}\left(\lceil x \rceil \mid |k\rangle_F\right) \geq \frac{8}{\pi^2} \approx 0.82.$$

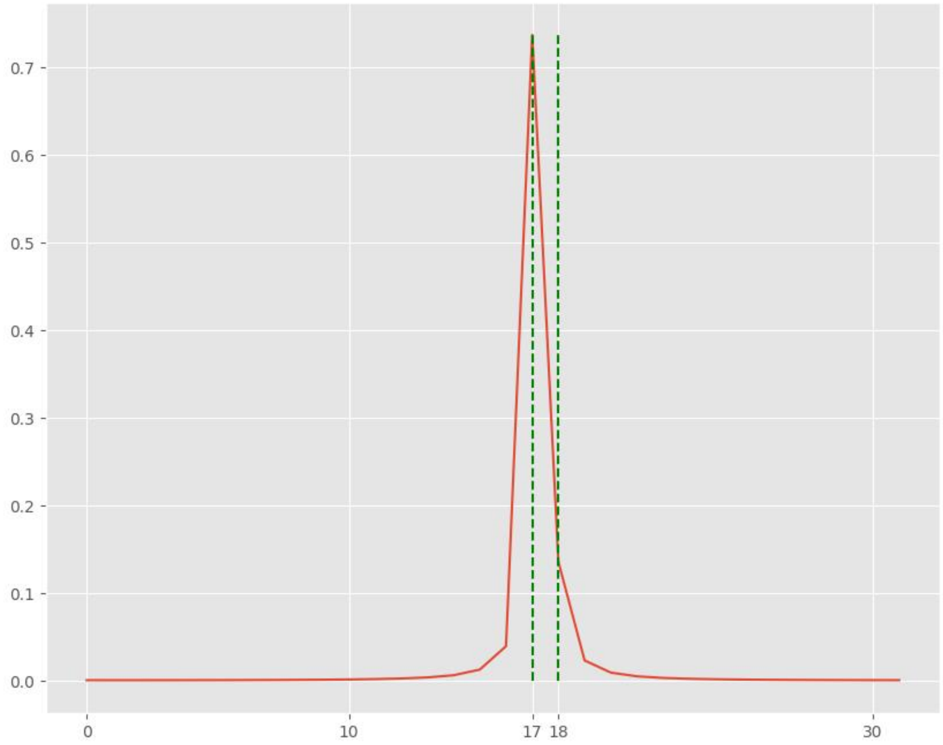


FIGURE 2. Ferjér kernel for $k = 17.3$ and $n = 5$.

19.2. **Quantum Phase Estimator/Digitizer.** Let $\mathcal{H}$ be the Hilbert space of m-qubit states. Given

- an implementation of an unitary operator U;
- an *eigenstate* $|\psi\rangle \in \mathcal{H}$ of U, with $U|\psi\rangle = \mathrm{e}^{2\pi\mathrm{i}\theta}|\psi\rangle$, with $0 \le \theta < 1$;

we want to find or estimate θ.

Remark. Note that $|\psi\rangle \equiv \mathrm{e}^{2\pi\mathrm{i}\theta}|\psi\rangle$, so physically there is no difference (and hence asking for θ, as is, is not a proper question). On the other hand, when we have a controlled gate CU given by

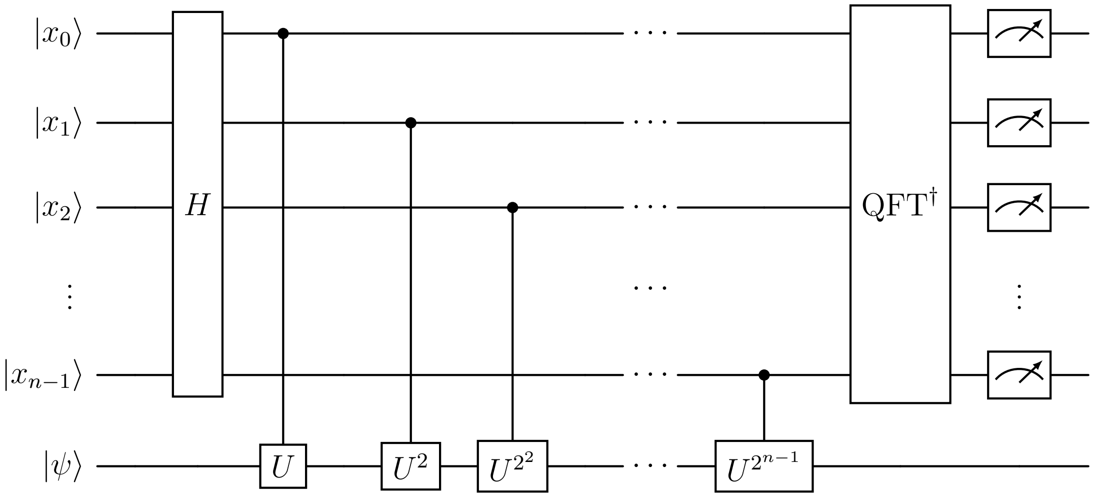
$$CU(|a\rangle|\psi\rangle) = |a\rangle U^a|\psi\rangle \quad (\text{for } a \in \mathbb{F}_2),$$

i.e.,

$$CU(\alpha \left|0\right\rangle \left|\psi\right\rangle + \beta \left|1\right\rangle \left|\psi\right\rangle) = \alpha \left|0\right\rangle \left|\psi\right\rangle + \beta \mathrm{e}^{2\pi \mathrm{i} \theta} \left|1\right\rangle \left|\psi\right\rangle ,$$

then θ is relevant.

Definition 19.2. We define the n-qubit *quantum phase estimator* $\mathrm{QPE} \left|x\right\rangle_n \left|\psi\right\rangle$ as:



We then have that

$$\mathrm{QPE} \left|x\right\rangle_n \left|\psi\right\rangle = \left|x + 2^n\theta\right\rangle_F \left|\psi\right\rangle . \qquad \text{(Is this really true?)}$$

In particular,

$$\mathrm{QPE} \left|0\right\rangle_n \left|\psi\right\rangle = \left|2^n\theta\right\rangle_F \left|\psi\right\rangle .$$

Hence, measuring the first n qubits, we get either $\lfloor 2^n\theta \rceil$ or $\lceil 2^n\theta \rceil$ with probability at least 82%. Pick the most likely $x_\theta \in \mathbb{Z}$ (i.e., repeat it a few times and choose the one that appears the most) and then $\theta \approx x_\theta/2^n$.

Proof. We prove only the fact that $\mathrm{QPE} \left|0\right\rangle_n \left|\psi\right\rangle = \left|2^n\theta\right\rangle_F \left|\psi\right\rangle$. First, note that

$$\begin{aligned} \hat{P}(k) \, \mathrm{QFT} \left|0\right\rangle_n &= \hat{P}(k) \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \zeta_{2^n}^{0 \cdot z} \left|z\right\rangle_n \\ &= \frac{1}{2^{n/2}} \sum_{z=0}^{2^n-1} \zeta_{2^n}^{kz} \left|z\right\rangle_n . \end{aligned}$$

Also,

$$\begin{aligned} H^{\otimes n} \left|0\right\rangle_n &= \bigotimes_{i=0}^{n-1} \frac{1}{\sqrt{2}} \left(\left|0\right\rangle + \left|1\right\rangle\right) \\ &= \frac{1}{2^{n/2}} \sum_{z=0}^{2^{n-1}} \left|z\right\rangle_n . \end{aligned}$$

where $\sigma(z)$ is either 1 or -1 depending on the binary digits of z.

Then, also denoting $z = z_0 + z_1 \cdot 2 + z_2 \cdot 2^2 + \cdots + z_n \cdot 2^{n-1}$, with $z_i \in \{0,1\}$ and $k \overset{\text{def}}{=} 2^n\theta$, we have:

$$\begin{aligned} \left|0\right\rangle_n \left|\psi\right\rangle &\mapsto \frac{1}{2^{n/2}} \sum_{z=0}^{2^{n-1}} \left|z\right\rangle_n \left|\psi\right\rangle \\ &\mapsto \frac{1}{2^{n/2}} \sum_{z=0}^{2^{n-1}} \mathrm{e}^{2\pi \mathrm{i}\theta z_0} \left|z\right\rangle_n \left|\psi\right\rangle \\ &\mapsto \frac{1}{2^{n/2}} \sum_{x=0}^{2^{n-1}} \mathrm{e}^{2\pi \mathrm{i}\theta (2z_1+z_0)} \left|z\right\rangle_n \left|\psi\right\rangle \\ &\vdots \\ &\mapsto \frac{1}{2^{n/2}} \sum_{x=0}^{2^{n-1}} \mathrm{e}^{2\pi \mathrm{i}\theta z} \left|z\right\rangle_n \otimes \left|\psi\right\rangle \\ &= \frac{1}{2^{n/2}} \sum_{x=0}^{2^{n-1}} \zeta_{2n}^{kz} \left|z\right\rangle_n \otimes \left|\psi\right\rangle \\ &= \hat{P}(k)\, \mathrm{QFT} \left|0\right\rangle_n \otimes \left|\psi\right\rangle \\ &\mapsto \mathrm{QFT}^\dagger\, \hat{P}(k)\, \mathrm{QFT} \left|0\right\rangle_n \otimes \left|\psi\right\rangle \\ &= \left|k\right\rangle_F \left|\psi\right\rangle = \left|2^n\theta\right\rangle_F \left|\psi\right\rangle . \end{aligned}$$

□

20. Shor's Algorithm

Given $a \in \mathbb{Z}/N\mathbb{Z}$, with $\gcd(a, N) = 1$, we want to find the order of $a \in \mathbb{Z}/N\mathbb{Z}^{\times}$. Classical methods are superpolynomial in $\log_2(N)$, with $\mathcal{O}\left((1+\epsilon)^{\log_2(N)}\right)$ for any $\epsilon > 0$.

Let $n \overset{\text{def}}{=} \lceil \log_2(N) \rceil$. Shor showed that one can implement

$$U_a \left| x \right\rangle_n = \begin{cases} \left| ax \bmod N \right\rangle_n, & \text{if } 0 \leq x < N; \\ \left| x \right\rangle_n, & \text{if } N \leq x < 2^n; \end{cases}$$

with $\mathcal{O}(n^2)$ gates and ancillas. Then, one can use QPE to compute the order.

20.1. **The Spectrum of U_a.** Suppose $\gcd(a, N) = 1$. Then for $x \geq N$, we have that $\left| x \right\rangle_n$ is an eigenstate of U_a, with eigenvalue 1.

Now, let $r \overset{\text{def}}{=} |a|$. Then, clearly we have that $x^r - 1$ is the minimal polynomial of U_a, so the eigenvalues of U_a are of the form $\mathrm{e}^{2\pi \mathrm{i} k/r}$, for $k \in \{0, 1, \ldots, r-1\}$.

Let $\zeta_r \overset{\text{def}}{=} \mathrm{e}^{2\pi\mathrm{i}/r}$ and

$$\left| \phi_k \right\rangle \overset{\text{def}}{=} \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \zeta_r^{-ks} \left| a^s \bmod N \right\rangle$$

for $k \in \{0, 1, \ldots, r-1\}$. Then, it is easy to check that

$$\begin{aligned} U_a \left| \phi_k \right\rangle &= \zeta_r^{-k} \left| \phi_k \right\rangle; \\ \left\langle \phi_{k_1} \mid \phi_{k_2} \right\rangle &= \delta_{k_1, k_2}; \\ \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \left| \phi_k \right\rangle &= \left| 1 \right\rangle_n = \left| 00\ldots01 \right\rangle. \end{aligned}$$

Proof. We have, remembering that $r = |a|$:

$$\begin{aligned}
U_a \left|\phi_k\right\rangle &= \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \zeta_r^{-ks} \left|a^{s+1} \bmod N\right\rangle \\
&= \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \zeta_r^{-k(s-1)} \left|a^{s} \bmod N\right\rangle \\
&= \zeta_r^k \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \zeta_r^{-ks} \left|a^{s} \bmod N\right\rangle \\
&= \zeta_r^k \left|\phi_k\right\rangle .
\end{aligned}$$

Also,

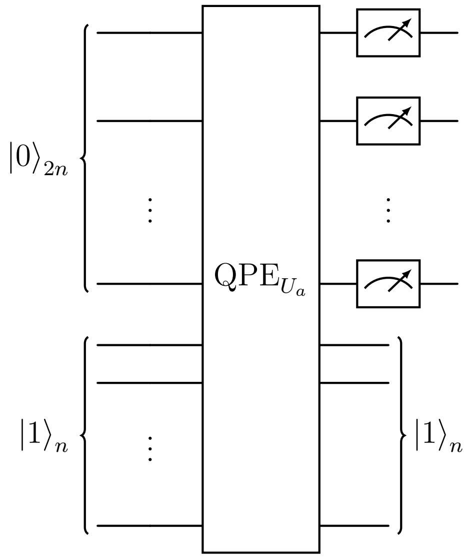
$$\begin{aligned}
\left\langle \phi_{k_1} \mid \phi_{k_2} \right\rangle &= \frac{1}{r} \sum_{s=0}^{r-1} \zeta_r^{k_1 s} \zeta_r^{-k_2 s} \\
&= \frac{1}{r} \sum_{s=0}^{r-1} \zeta_r^{(k_1-k_2)s} \\
&= \delta_{k_1,k_2}.
\end{aligned}$$

Finally,

$$\begin{aligned}
\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \left|\phi_k\right\rangle &= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \zeta_r - ks \left|a^{s} \bmod N\right\rangle \\
&= \frac{1}{r} \sum_{s=0}^{r-1} \left[\sum_{k=0}^{r-1} \zeta_r - ks \right] \left|a^{s} \bmod N\right\rangle \\
&= \left|1\right\rangle_n .
\end{aligned}$$

□

Consider the circuit:



Since $U_{a^{2^k}}$ needs $\mathcal{O}(n^2)$ gates, we have that QPE_{U_a} needs $\mathcal{O}(n^3)$ gates.

Now, since $U_a |\phi_k\rangle = \mathrm{e}^{2\pi \mathrm{i} k/r}$:

$$\begin{aligned}\mathrm{QPE}_{U_a}(|0\rangle_{2n} |1\rangle_n) &= \mathrm{QPE}_{U_a}\left(|0\rangle_{2n} \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\phi_k\rangle \right) \\ &= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \mathrm{QPE}_{U_a} \left(|0\rangle_{2n} |\phi_k\rangle \right) \\ &= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |2^{2n} k/r\rangle_F |\phi_k\rangle .\end{aligned}$$

Let then

$$|2^{2n} k/r\rangle_F = \sum_{y=0}^{2^{2n}-1} c_{k,y} |y\rangle ,$$

and hence $|c_{k,y}|$ is quite small, except when y is the integer closest to $2^{2n}k/r$.

Then, we can write

$$\mathrm{QPE}_{U_a}(|0\rangle_{2n} |1\rangle_n) = \frac{1}{r} \sum_{k=0}^{r-1} \sum_{s=0}^{r-1} \sum_{y=0}^{2^{2n}-1} c_{k,y} \zeta_r^{-ks} |y\rangle |a^s \bmod N\rangle ,$$

and therefore, the probability of reading y in the first $2n$-qubits is

$$\sum_{s=0}^{r-1} \frac{1}{r^2} \left| \sum_{k=0}^{r-1} c_{k,y} \zeta_r^{-ks} \right|^2 \leq \frac{1}{r^2} \sum_{s=0}^{r-1} \sum_{k=0}^{r-1} |c_{k,y}|^2 ,$$

which is quite small if y is not the closest integer to $2^{2n}k/r$ for some k. Therefore, we most likely will get $y = \lfloor 2^{2n}k/r \rceil$ for some k, in which case we must have

$$\left| y - 2^{2n}\frac{k}{r} \right| \leq \frac{1}{2} \quad \iff \quad \left| \frac{y}{2^{2n}} - \frac{k}{r} \right| \leq \frac{1}{2^{2n+1}}.$$

We now need the following result:

Lemma 20.1. *Let* $n \overset{\text{def}}{=} \lceil \log_2(N) \rceil \geq 2$. *Then, for a fraction* k/r *with* $0 \leq k < r < N$, *there is a unique* $y \in \{0, 1, 2, \ldots, 2^{2n} - 1\}$ *such that*

$$\left| \frac{y}{2^{2n}} - \frac{k}{r} \right| < \frac{1}{2^{2n+1}}.$$

Proof. We have

$$0 = \frac{0}{2^{2n}} \leq \frac{k}{r} \leq \frac{N-1}{N} = 1 - \frac{1}{N} \leq 1 - \frac{1}{2^n} < 1 - \frac{2}{2^{2n}} = \frac{2^{2n} - 2}{2^{2n}},$$

i.e.,

$$\frac{0}{2^{2n}} \leq \frac{k}{r} < \frac{2^{2n} - 2}{2^{2n}},$$

So, there is $y_0 \in \{0, 1, \ldots, 2^{2n} - 2\}$ such that

$$\frac{y_0}{2^{2n}} \leq \frac{k}{r} < \frac{y_0 + 1}{2^{2n}}.$$

Thus, taking y as either y_0 or $y_0 + 1$, and hence $y \in \{0, 1, 2, \ldots, 2^{2n} - 1\}$, we can obtain

$$\left| \frac{y}{2^{2n}} - \frac{k}{r} \right| \leq \frac{1}{2} \frac{1}{2^{2n}} = \frac{1}{2^{2n+1}}.$$

On the other hand, the only way we get the equality is if

$$\frac{k}{r} - \frac{y_0}{2^{2n}} = \frac{1}{2^{2n+1}} \quad \Longrightarrow \quad \frac{k}{r} = \frac{2y_0 + 1}{2^{2n+1}},$$

which would imply that $2^{2n+1} \mid r$ (since the fraction on the right is reduced), which is a contradiction, as $r \leq N \leq 2^n$. Thus, we have that there exists y with

$$\left| \frac{y}{2^{2n}} - \frac{k}{r} \right| < \frac{1}{2^{2n+1}}.$$

Now, suppose that we also have $y' \neq y$, with $y' \in \{0, 1, 2, \ldots, 2^{2n} - 1\}$, such that

$$\left| \frac{y'}{2^{2n}} - \frac{k}{r} \right| < \frac{1}{2^{2n+1}}.$$

Then,

$$\frac{1}{2^{2n}} \leq \left| \frac{y'}{2^{2n}} - \frac{y}{2^{2n}} \right| \leq \left| \frac{y'}{2^{2n}} - \frac{k}{r} \right| + \left| \frac{y}{2^{2n}} - \frac{k}{r} \right| < \frac{1}{2^{2n}},$$

which is a contradiction.

□

Hence, by this lemma, if measuring y in the first $2n$ qubits, then there is a unique fraction k/r (where $r = |a|$) such that

$$\left| \frac{y}{2^{2n}} - \frac{k}{r} \right| < \frac{1}{2^{2n+1}}.$$

We then can use *continued fractions* to find r: remember that there are $a_1, a_2, a_3, \ldots \in \mathbb{Z}$ such that

$$\frac{y}{2^{2n}} = [a_1, a_2, a_3, \ldots] \stackrel{\text{def}}{=} \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}}.$$

Then, if $[a_1, a_2, \ldots, a_n] = p_n/q_n$ in reduced form, by the theory of continued fractions, we must have

$$\left| \frac{y}{2^{2n}} - \frac{p_n}{q_n} \right| < \frac{1}{2q_n^2}.$$

Therefore, we use continued fractions until we get $\frac{1}{2q_n^2} \leq \frac{1}{2^{2n+1}}$ (with $q_n < N$ still). Then, we must have

$$\frac{k}{r} = \frac{p_n}{q_n},$$

with $q_n \mid r$, since p_n/q_n is reduced. We can then compute a^{q_n}: if we get 1, we found the order r. If not, we can try some small multiples that are still less than N. If that also fails, we can repeat the method to find $|a^{q_n}|$. Alternatively, we could do another reading, which would likely give us another y' such that $y'/2^{2n}$ is close to k'/r, for some likely k', with likely $k' \neq k$, and find a new fraction p'_t/q'_t using continued fractions. If $q'_t s$ also does not work, but it is different from q_n, we know that $\operatorname{lcm}(q_n, q'_t) \mid r$.

Remark. Currently, the depth of for Shor's Algorithm is $\mathcal{O}(n \log(n))$

21. Quantum Data Access Oracles

In classical data we often need functions $f : \mathbb{F}_2^n \to \mathbb{F}_2^d$, e.g., for lists, dictionaries, etc. How do we implement these on a quantum computer?

One possibility is to construct a *data access oracle*:

$$U_f \left|x\right\rangle_n \left|0\right\rangle_d \overset{\text{def}}{=} \left|x\right\rangle_n \left|f(x)\right\rangle_d .$$

(Note we do not specify $U_f \left|x\right\rangle_n \left|y\right\rangle_d$ for $y \neq 0$.) *Problem:* How do we construct U_f?

Remark. Quantum data access oracles are sometimes called QRAM, QROM, or quantum dictionaries.

Alternatively, we could use *diagonal unitaries*:

$$\mathcal{D}_f \left|x\right\rangle_n \overset{\text{def}}{=} \mathrm{e}^{2\pi \mathrm{i}/2^d f(x)} \left|x\right\rangle_n$$

(seeing $f(x)$ as the integer corresponding to the one given by its binary representation). *Problem:* How do we construct $\mathcal{D}_f$?

In most cases, having one constructions leads to the other.

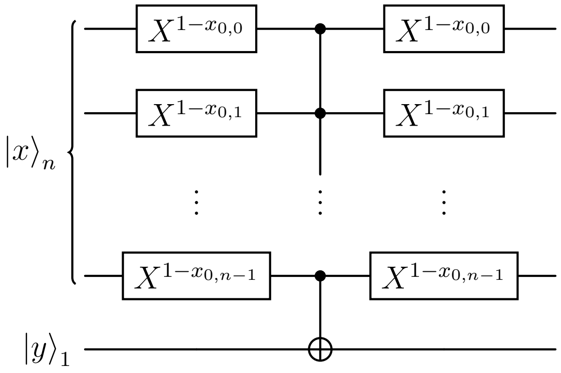
21.1. **Implementing U_f.** There are two main implementations: select QROM and swap QRAM, and a combination of the two called select-swap. (There are many others, though.)

21.1.1. *Select QROM.* QROM means that f is embedded in the circuit, and not coming from data in the quantum computer.

Definition 21.1. Given $x_0 \in \mathbb{F}_2^n$, we define the gate $C_{x_0}^n X$ by:

$$C_{x_0}^n X \left|x\right\rangle_n \left|y\right\rangle_1 = \begin{cases} \left|x\right\rangle_n \left|y\right\rangle_1, & \text{if } x \neq x_0, \\ \left|x\right\rangle_n \left|y+1\right\rangle_1, & \text{if } x = x_0. \end{cases}$$

If $x_0 = x_{0,0} + x_{0,1} \cdot 2 + c_{0,2} \cdot 2^2 + \cdots + c_{0,n-1} \cdot 2^{n-1}$, then we have the $C_{x_0}^n X$ is given by



The circuit above works by simply making $\left|x_0\right\rangle_n \mapsto \left|11\cdots 1\right\rangle$ in the first step (and $\left|x\right\rangle$ does not give $\left|111\cdots 1\right\rangle$ when $x \neq x_0$), then the control gates flips $\left|0\right\rangle$ to $\left|1\right\rangle$, and then undoing the first step.

Remark. Note that $C_{x_0}^n X$ is the same U_f for

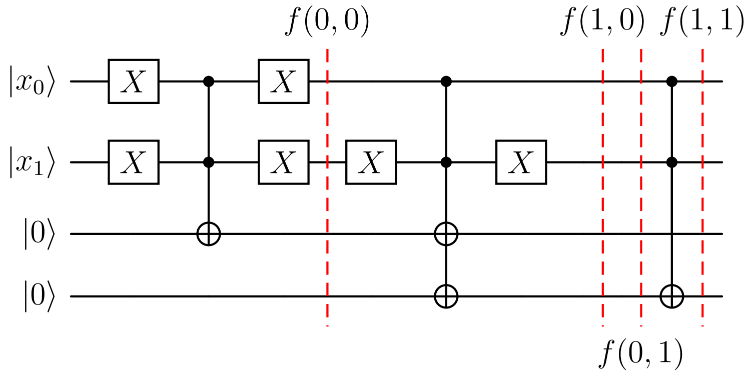
$$f(x) = \begin{cases} 0, & \text{if } x \neq x_0, \\ 1, & \text{if } x = x_0. \end{cases}$$

Then, to implement C_f, we just use various $C_{x_0}^n X$ flipping the corresponding qubits in the output. (See example below.)

Example 21.2. Consider

$$f(0,0) = (1,0), \qquad f(1,0) = (1,1),$$
$$f(0,1) = (0,0), \qquad f(1,1) = (0,1).$$

Here is C_f:



Remarks. (1) This naive implementation often have many cancellations of consecutive X gates.

(2) The state of the art implementation can be done with $\mathcal{O}(d2^n)$ H, S, and CX gates, and $\mathcal{O}(2^n)$ T gates. (Note that while H, S, and CX gates are relatively "cheap", the T gate is relatively "expensive".) Here, $d2^n$ can, more or less, be replaced by the number of non-zero bits in all outputs (so at most $d2^n$), and 2^n (for T gates) be replaced by the number of outputs different from $(0,\dots,0)$ (at most $2n$).

(3) This select QROM is efficient when f has either few non-zero entries or if its *unstructured*.

(4) The size of the ancilla in this case is just d, so it is narrow.

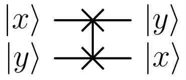
(5) On the other hand, it is deep, meaning of high depth.

21.1.2. *Swap QRAM.* QRAM means that the quantum computer contains the data for f, while the circuit itself does not.

Definition 21.3. The *swap gate* SWAP is defined as

$$\text{SWAP}\,|x\rangle_1\,|y\rangle_1 \overset{\text{def}}{=} |y\rangle_1\,|x\rangle_1\,.$$

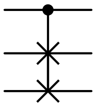
It is represented by:



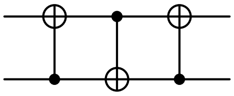
The *controlled swap gate* CSWAP is defined as

$$\text{CSWAP}\,|xyz\rangle = \begin{cases} |xyz\rangle\,, & \text{if } x = 0, \\ |xzy\rangle\,, & \text{if } x = 1. \end{cases}$$

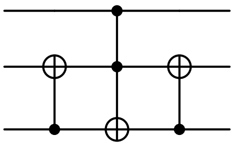
It is represented as



Note that SWAP can be implemented as



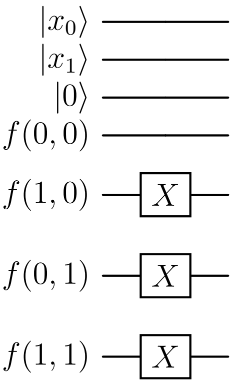
and CSWAP as



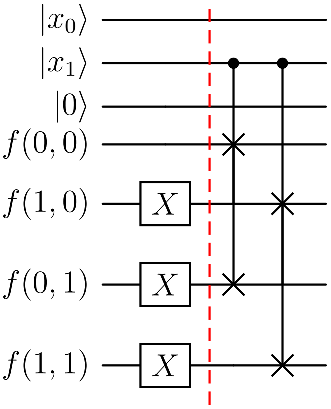
Example 21.4. Let's give an example of a QRAM swap with $n = 2$ and $d = 1$:

$$f(0,0) = 0, \qquad f(1,0) = 1$$
$$f(0,1) = 1, \qquad f(1,1) = 1$$

We first create the possible outputs in the last 2^n qubits, corresponding to $f(0,0)$, $f(1,0)$, $f(0,1)$, and $f(1,1)$ respectively:

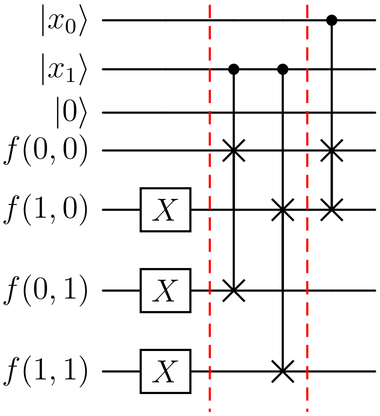


If the last qubit of the input is 1, we swap the correct output to the places where the last qubits were 0:

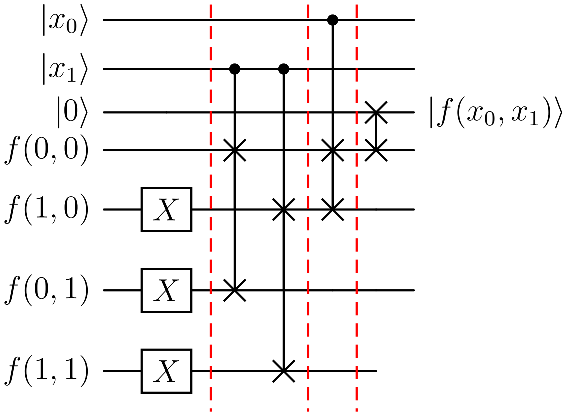


This makes the correct output $f(x_0, x_1)$ to be in the spots of $f(x_0, 0)$.

Then, if the first qubit is 1, we swap the correct output to where the first qubits were 0:



This makes then the correct output $f(x_0, x_1)$ to be in the spot of $f(0, 0)$. At this point we could just measure the fourth qubit (corresponding to $f(0, 0)$), but we could do a final swap to put it in the third spot:



Example 21.5. Here is a more complex example (without the final swap), the same as above:

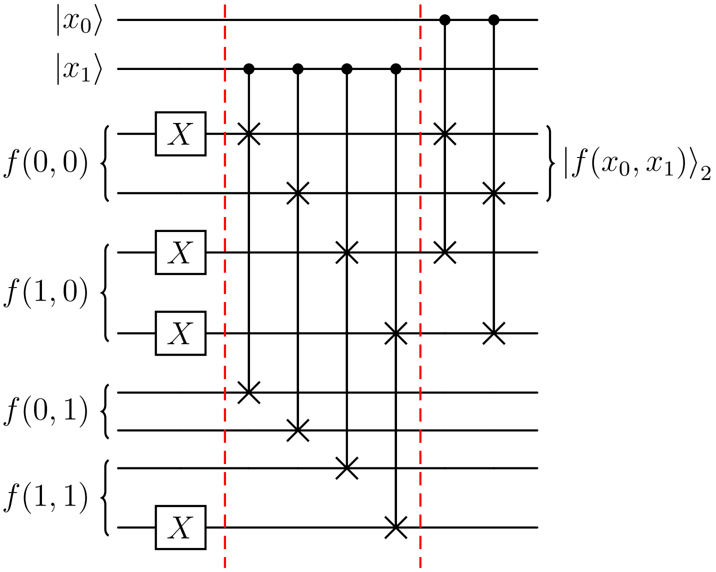
$$f(0,0) = (1,0), \qquad f(1,0) = (1,1),$$
$$f(0,1) = (0,0), \qquad f(1,1) = (0,1).$$

The steps are the same:

(1) set the correct outputs for $f(x_0, y_0)$ in their respective spots;

(2) place the result of $f(x_0, y_0)$ in the spots for $f(x_0, 0)$;
(3) place the result of $f(x_0, y_0)$ in the spots for $f(0, 0)$.

We have:



Remark. The ancilla needed in this method needs $2^n d$ qubits, so it is very *wide*, but is shallow (low depth). Even the state of the art has $\mathcal{O}(d2^n)$ gates. (The, $d2^n$ can be replaced by the number of non-zero bits in the outputs of f, which is much better if the data is sparse, with many zeros.)

21.1.3. *Select-Swap.* One can combine the two methods. We can select k qubits at the end of the input of f to obtain a function $g : \mathbb{F}_2^{n-k} \to \mathbb{F}_2^{d2^k}$:

$$g(x_0, \ldots, x_{n-k-1}) \stackrel{\text{def}}{=} (g_1(x_0, \ldots, x_{n-k-1}), \ldots, g_{2^k-1}(x_0, \ldots, x_{n-k-1})),$$

where $g_i : \mathbb{F}_2^{n-k} \to \mathbb{F}_2^d$ is defined as

$$g_i(x_0, \ldots, x_{n-k-1}) \stackrel{\text{def}}{=} f(x_0, \ldots, x_{n-k-1}, \text{“}k \text{ binary representation of } i\text{”}).$$

We can then use the select method for each g_i and then swap to get the correct answer in the $f(0, 0, \ldots, 0)$ spot. This makes it so we need $d2^k$ ancillas, compared to the $d2^n$ of the swap method.

Note that if $k = n$, then we are using the swap method, and if $k = 0$, then we are using the select method.

Example 21.6. Consider

$$\begin{array}{llll}
f(0,0,0,0) = (0,1,0) & f(1,0,0,0) = (1,0,0), & f(0,1,0,0) = (0,0,0) & f(1,1,0,0) = (1,1,0), \\
f(0,0,1,0) = (0,0,1) & f(1,0,1,0) = (0,0,1), & f(0,1,1,0) = (0,1,0) & f(1,1,1,0) = (0,1,0), \\
f(0,0,0,1) = (0,0,0) & f(1,0,0,1) = (1,1,0), & f(0,1,0,1) = (1,1,1) & f(1,1,0,1) = (0,0,0), \\
f(0,0,1,1) = (1,1,1) & f(1,0,1,1) = (0,1,0), & f(0,1,1,1) = (0,0,0) & f(1,1,1,1) = (0,1,0).
\end{array}$$

For $k = 2$, we have:

$$\begin{aligned}
g_0(0,0) &= (0,1,0), & g_0(1,0) &= (1,0,0), \\
g_0(0,1) &= (0,0,0), & g_0(1,1) &= (1,1,0),
\end{aligned}$$

and

$$\begin{aligned}
g_1(0,0) &= (0,0,1), & g_1(1,0) &= (0,0,1), \\
g_1(0,1) &= (0,1,0), & g_1(1,1) &= (0,1,0),
\end{aligned}$$

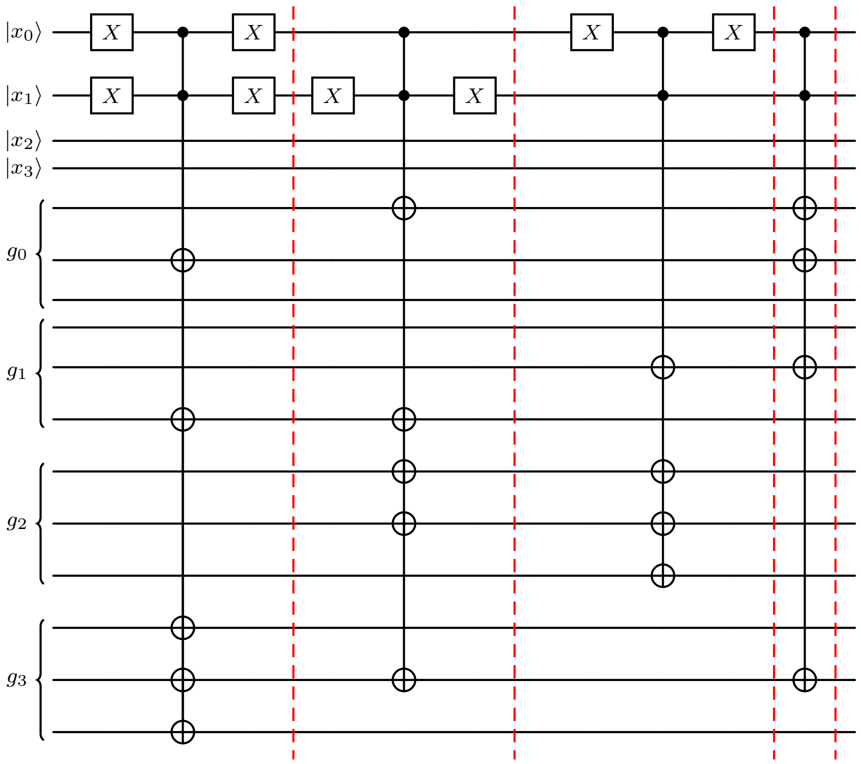
and

$$\begin{aligned}
g_2(0,0) &= (0,0,0), & g_2(1,0) &= (1,1,0), \\
g_2(0,1) &= (1,1,1), & g_2(1,1) &= (0,0,0),
\end{aligned}$$

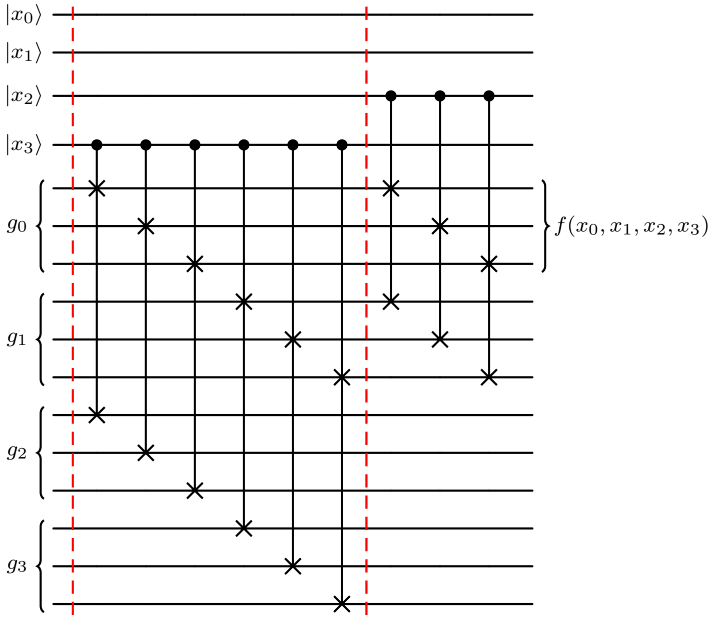
and

$$\begin{aligned}
g_3(0,0) &= (1,1,1), & g_3(1,0) &= (0,1,0), \\
g_3(0,1) &= (0,0,0), & g_3(1,1) &= (0,1,0).
\end{aligned}$$

We then get:



followed by the swaps:



So,

$$U_f^{\mathrm{Sel-Swap}} \left|x\right\rangle_n \left|0\right\rangle_{d2^k} = \left|x\right\rangle_n \left|f(x)\right\rangle_d \left|\mathrm{garbage}\right\rangle_{d(2^k-1).}$$

The total cost is the cost of the 2^k select, plus the swap of k. So, the total gate cost is still $\mathcal{O}(d2^n)$ and width $\mathcal{O}(2^{n-k} + d2^k)$. Choosing k such that $2^k \approx \sqrt{2^n/d}$ (or, more roughly, $k \approx n/2$), we get width of $\mathcal{O}\left(\sqrt{d2^n}\right)$.

Remark. If we restrict to H, T, S, and CX, then the T cost is only $\mathcal{O}\left(\sqrt{d2^n}\right)$. (A bit higher on the others.)

21.2. **Diagonal Unitary.** Given some $\theta : \mathbb{F}_2^n \to [0,1)$, we might want to construct the operator

$$U_\theta \left|x\right\rangle_n = \mathrm{e}^{2\pi\mathrm{i}\theta(x)} \left|x\right\rangle_n .$$

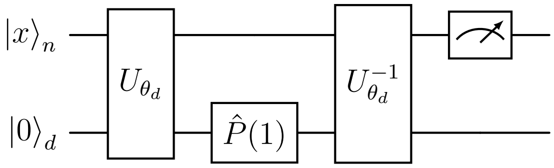
We can approximate it using data access oracles.

If $\epsilon > 0$ is the error, and in binary

$$\theta(x) = 0.\underbrace{y_0 y_1 \cdots y_d}_{\overset{\mathrm{def}}{=} \theta_d(x)} y_{d+1} y_{d+2} \cdots$$

then $\theta_d : \mathbb{F}_2^n \to \mathbb{F}_2^d$ and $\left|\mathrm{e}^{2\pi\mathrm{i}\theta(x)} - \mathrm{e}^{2\pi\mathrm{i}\theta_d(x)}\right| \leq c/2^d < \epsilon$, for some constant c. So, we need $d \approx \log_2(1/\epsilon) + \mathcal{O}(1)$.

So, with the data access oracle U_{θ_d}, we can make



where the $\hat{P}$ gate is given as in Definition 18.7:

$$\begin{aligned}\hat{P}(1)\left|x\right\rangle_n &\overset{\text{def}}{=} \hat{P}(1/2^{n-1})\left|x_0\right\rangle \otimes \hat{P}(1/2^{n-2})\left|x_1\right\rangle \otimes \cdots \otimes \hat{P}(1)\left|x_{n-1}\right\rangle \\ &= P(\pi/2^{n-1})\left|x_0\right\rangle \otimes P(\pi/2^{n-2})\left|x_1\right\rangle \otimes \cdots \otimes P(\pi)\left|x_{n-1}\right\rangle .\end{aligned}$$

This works since we get

$$\begin{aligned}\left|x\right\rangle_n \left|0\right\rangle_d &\mapsto \left|x\right\rangle_n \left|\theta_d(x)\right\rangle_d \\ &\mapsto \left|x\right\rangle_n \mathrm{e}^{2\pi \mathrm{i}\theta_d(x)/2^d}\left|\theta_d(x)\right\rangle_d \\ &= \mathrm{e}^{2\pi \mathrm{i}\theta_d(x)/2^d}\left|x\right\rangle_n \left|\theta_d(x)\right\rangle_d \\ &\mapsto \left(\mathrm{e}^{2\pi \mathrm{i}\theta_d(x)/2^d}\left|x\right\rangle_n\right)\left|0\right\rangle_d \\ &\approx \left(\mathrm{e}^{2\pi \mathrm{i}\theta(x)}\left|x\right\rangle_n\right)\left|0\right\rangle_d .\end{aligned}$$

One more observation. Let:

Definition 21.7. Given $\theta : \mathbb{F}_2^n \to [0,1)$ as above, a *multiplexer* of type Y (or X, or Z):

$$R_Y(\theta)\left|x\right\rangle_n \left|\psi\right\rangle_1 = \left|x\right\rangle_n R_Y(\theta(x))\left|\psi\right\rangle_1 ,$$

with

$$R_Y(\theta) = \begin{pmatrix}\cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2)\end{pmatrix}$$

as in Section 5.2.

These can also be done with data access oracles!

22. Quantum State Preparation

Given $\{\psi_x : x \in \mathbb{F}_2^n\} \subseteq \mathbb{C}^{2n}$, can we construct U_ψ such that $U_\psi \left|0\right\rangle_n = \left|\psi\right\rangle_n$?

Here is a construction that is not always the best, but always works and often is the best. We construct it inductively.

(1) Let $\psi_x = \mathrm{e}^{2\pi \mathrm{i}\theta(x)}\rho_x$, where $\rho_x \overset{\text{def}}{=} |\psi_x|$. Then, if

$$\rho \overset{\text{def}}{=} \sum_{x=0}^{2^n-1} \rho_x \left|x\right\rangle ,$$

we have that $\left|\psi\right\rangle = U_\theta \left|\rho\right\rangle$. (We can choose, for instance, $\theta(x) = 0$ if $\rho_x = 0$.) Since we can get U_θ, it suffices to prepare ρ, and hence we may assume without loss of generality that $\psi_x \geq 0$ for all x (and therefore, real).

(2) Let

$$\begin{aligned} \left|\psi\right\rangle_n &= \sum_{x\in\mathbb{F}_2^n} \psi_x \left|x\right\rangle_n \\ &= \sum_{y\in\mathbb{F}_2^{n-1}} \psi_{y,0} \left|y\right\rangle_{n-1} \left|0\right\rangle + \psi_{y,1} \left|y\right\rangle_{n-1} \left|1\right\rangle \\ &= \sum_{y\in\mathbb{F}_2^{n-1}} \sqrt{\psi_{y,0}^2 + \psi_{y,1}^2} \left|y\right\rangle_{n-1} \otimes \left(C(y)\left|0\right\rangle + S(y)\left|1\right\rangle\right), \end{aligned}$$

where

$$\begin{aligned} C(y) &\overset{\text{def}}{=} \frac{\psi_{y,0}}{\sqrt{\psi_{y,0}^2 + \psi_{y,1}^2}}, \\ S(y) &\overset{\text{def}}{=} \frac{\psi_{y,1}}{\sqrt{\psi_{y,0}^2 + \psi_{y,1}^2}}. \end{aligned}$$

(Note that if $\psi_{y,0} = \psi_{y,1} = 0$ above, we can just skip the term in the summation.)

Note that since $C(y), S(y) \geq 0$ and $C(y)^2 + S(y)^2 = 1$, there exists a unique $\theta(y) \in [0, \pi]$ such that $C(y) = \cos(\theta(y)/2)$ and $S(y) = \sin(\theta(y)/2)$.

Also, if we let

$$\left|\chi\right\rangle_{n-1} \overset{\text{def}}{=} \sum_{y\in\mathbb{F}_2^{n-1}} \sqrt{\psi_{y,0}^2 + \psi_{y,1}^2} \left|y\right\rangle_{n-1},$$

we have that

$$\left\langle \chi \mid \chi \right\rangle = \sum_{y\in\mathbb{F}_2^{n-1}} \psi_{y,0}^2 + \psi_{y,1}^2 = \sum_{x\in\mathbb{F}_2^n} \psi_x^2 = \left\langle \psi \mid \psi \right\rangle = 1.$$

(3) Hence, if we can prepare $|\chi\rangle_{n-1}$, we can prepare $|\psi\rangle_n$ using a multiplexer (as defined above, but not explicitly shown), as

$$|\psi\rangle_n = R_Y(\theta)\,|\chi\rangle_{n-1}\,|0\rangle_1\,,$$

where θ is the function on y above.

(4) Iterating this process, we see that it suffices to prepare any initial 1-qubit state, which we can do exactly as above, as given $|\psi\rangle_1 = a\,|0\rangle + b\,|1\rangle$, with $a, b \geq 0$ there is some θ such that $R_Y(\theta)\,|0\rangle = |\psi\rangle_1$.

Example 22.1. Let's look at the $n = 2$ case, with $\psi_x \geq 0$:

$$\begin{aligned}
|\psi\rangle &= \psi_{0,0}\,|00\rangle + \psi_{1,0}\,|10\rangle + \psi_{0,1}\,|01\rangle + \psi_{1,1}\,|11\rangle \\
&= |0\rangle\left(\psi_{0,0}\,|0\rangle + \psi_{0,1}\,|1\rangle\right) + |1\rangle\left(\psi_{1,0}\,|0\rangle + \psi_{1,1}\,|1\rangle\right) \\
&= \sqrt{\psi_{0,0}^2 + \psi_{0,1}^2}\,|0\rangle\left(\frac{\psi_{0,0}}{\sqrt{\psi_{0,0}^2 + \psi_{0,1}^2}}\,|0\rangle + \frac{\psi_{0,1}}{\sqrt{\psi_{0,0}^2 + \psi_{0,1}^2}}\,|1\rangle\right) \\
&\quad + \sqrt{\psi_{1,0}^2 + \psi_{1,1}^2}\,|1\rangle\left(\frac{\psi_{1,0}}{\sqrt{\psi_{1,0}^2 + \psi_{1,1}^2}}\,|0\rangle + \frac{\psi_{1,1}}{\sqrt{\psi_{1,0}^2 + \psi_{1,1}^2}}\,|1\rangle\right).
\end{aligned}$$

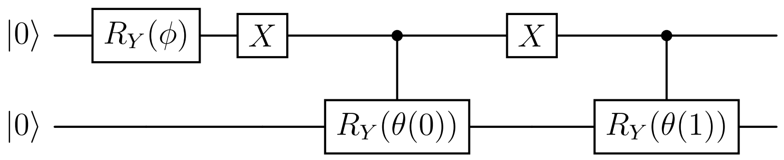
Then,

$$\begin{aligned}
\cos\left(\frac{\theta(0)}{2}\right) &\overset{\text{def}}{=} \frac{\psi_{0,0}}{\sqrt{\psi_{0,0}^2 + \psi_{0,1}^2}}, \\
\cos\left(\frac{\theta(1)}{2}\right) &\overset{\text{def}}{=} \frac{\psi_{1,0}}{\sqrt{\psi_{1,0}^2 + \psi_{1,1}^2}}, \\
\cos\left(\frac{\phi}{2}\right) &\overset{\text{def}}{=} \sqrt{\psi_{0,0}^2 + \psi_{0,1}^2}.
\end{aligned}$$

Hence, we have:

$$\begin{aligned}
|\psi\rangle = \cos\left(\frac{\phi}{2}\right)|0\rangle\left(\cos\left(\frac{\theta(0)}{2}\right)|0\rangle + \sin\left(\frac{\theta(0)}{2}\right)\right)|1\rangle \\
+ \sin\left(\frac{\phi}{2}\right)|1\rangle\left(\cos\left(\frac{\theta(1)}{2}\right)|0\rangle + \sin\left(\frac{\theta(1)}{2}\right)|1\rangle\right).
\end{aligned}$$

So, we need:



Indeed, we have:

$$\begin{aligned}
|00\rangle &\mapsto (\cos(\phi/2)\,|0\rangle + \sin(\phi/2)\,|1\rangle) \otimes |0\rangle \\
&= \cos(\phi/2)\,|0\rangle\,|0\rangle + \sin(\phi/2)\,|1\rangle\,|0\rangle \\
&\mapsto \cos(\phi/2)\,|1\rangle\,|0\rangle + \sin(\phi/2)\,|0\rangle\,|0\rangle \\
&\mapsto \cos(\phi/2)\,|1\rangle\,(\cos(\theta(0)/2)\,|0\rangle + \sin(\theta(0)/2)\,|1\rangle) + \sin(\phi/2)\,|0\rangle\,|0\rangle \\
&\mapsto \cos(\phi/2)\,|0\rangle\,(\cos(\theta(0)/2)\,|0\rangle + \sin(\theta(0)/2)\,|1\rangle) + \sin(\phi/2)\,|1\rangle\,|0\rangle \\
&\mapsto \cos(\phi/2)\,|0\rangle\,(\cos(\theta(0)/2)\,|0\rangle + \sin(\theta(0)/2)\,|1\rangle) \\
&\qquad + \sin(\phi/2)\,|1\rangle\,(\cos(\theta(1)/2)\,|0\rangle + \sin(\theta(1)/2)\,|1\rangle)\,.
\end{aligned}$$

23. Quantum (Hamiltonian) Simulations

Quantum systems modeled by Hilbert spaces $\mathcal{H}$.

Important operator: H, the Hamiltonian. H is a self-adjoint operator, so it has real spectrum, and it can be diagonalized with an eigenbasis that is orthonormal.

Physical meanings:

(1) The spectrum of H are the possible energies of the system. For every energy value E, there is a quantum state $|\psi_E\rangle$ such that $H\,|\psi_E\rangle = E\,|\psi_E\rangle$.

(2) The *time dependent Schrödinger equation*: the *initial value problem*

$$\begin{aligned} |\psi(t_0)\rangle &= |\psi_0\rangle \,, \\ \frac{\mathrm{d}}{\mathrm{d}t} |\psi(t)\rangle &= -\frac{i}{\hbar} H \, |\psi(t)\rangle \,, \end{aligned}$$

where $\hbar$ is an universal constant. (H is assumed here to not depend on time.) The solution is

$$|\psi(t)\rangle = \exp\left(-\frac{i}{\hbar} tH\right) |\psi_0\rangle \,.$$

We want to simulate the time evolution!

Input: $|\psi_0\rangle$, H, and $t > 0$.
Output: $|\psi(t)\rangle$, or at least an approximation.

We have the formula, but it is hard to compute.

Remark. This, along with Shor's algorithm, is one of the most important applications of quantum computing, resulting in an exponential speed-up over conventional algorithms.

Definition 23.1. For an operator U, a *singular value* of U is the square root of an eigenvalue of the self-adjoint operator $U^\dagger U$. (The eigenvalues are then necessarily real and non-negative). Then, the *operator norm* of U, denoted by $\|U\|$, is the maximal singular value of U.

Remark. If U is diagonalizable, with real eigenvalues (like self-adjoint matrices), then $\|U\|$ is the maximal absolute value of the eigenvalues of U.

Definition 23.2. A Hamiltonian on n qubits can be *efficiently simulated* if there exists a quantum circuit $U_{t,\epsilon}$, for any small $t, \epsilon > 0$, such that

$$\|\exp(-\mathrm{i}tH/\hbar) - U_{t,\epsilon}\| < \epsilon,$$

and U is made out of $\mathrm{Poly}(n, t, 1/\epsilon)$ gates.

Example 23.3. Consider:

$$H = \sum_{r=1}^{m} H_r,$$

where each H_r is acting on at most $k \ll n$ qubits. (These are called *k-local Hamiltonians.*)

23.1. **Trotter's Formula.**

Theorem 23.4 (Trotter)**.** *We have*

$$\exp\left(-\frac{\mathrm{i}}{\hbar}(H_1 + H_2 + \cdots + H_m)t\right) = \lim_{\ell\to\infty}\left(\exp\left(-\frac{\mathrm{i}H_1}{\hbar\frac{t}{\ell}}\right)\exp\left(-\frac{\mathrm{i}H_2}{\hbar\frac{t}{\ell}}\right)\cdots\exp\left(-\frac{\mathrm{i}H_m}{\hbar\frac{t}{\ell}}\right)\right)^{\ell}$$

So, fix large $\ell > 0$: if you can construct $\exp(-\mathrm{i}H_r/\hbar \cdot t/\ell)$ for each r, then we have:

$$\exp\left(-\frac{\mathrm{i}}{\hbar}Ht\right) \approx \left(\exp\left(-\frac{\mathrm{i}H_1}{\hbar\frac{t}{\ell}}\right)\exp\left(-\frac{\mathrm{i}H_2}{\hbar\frac{t}{\ell}}\right)\cdots\exp\left(-\frac{\mathrm{i}H_m}{\hbar\frac{t}{\ell}}\right)\right)^{\ell}.$$

(See more details at the Montanaro's Notes.)

Example 23.5. Suppose that $H_r = \alpha_r\left(\sigma_{r,0}\otimes\sigma_{r,1}\otimes\cdots\otimes\sigma_{r,n-1}\right)$, with $\alpha_r \in \mathbb{R}$ and $\sigma_{r,s} \in \{\mathbb{I}, X, Y, Z\}$. $H = \sum_{r=1}^{m} H_r$ is k-local if and only if the number of non-identities is less than or equal to k in each H_r.

Claim: $\exp\left(-\mathrm{i}H_r t/\hbar\right)$ can be constructed (exactly, not approximated) using a single R_Z rotation, $\mathcal{O}(1)$ 1-qubit gates, and $\mathcal{O}(n)$ CX gates.

The idea of the proof is that

$$Y = SHZ(SH)^{\dagger} = SHZHS^{\dagger}, \qquad X = HZH^{\dagger} = HZH,$$

(so X, Y, and Z are conjugates of each other). So,

$$\begin{aligned}\exp\left(-\frac{\mathrm{i}H_r}{\hbar}t\right) &= \exp\left(-\frac{\mathrm{i}}{\hbar}\alpha_r(\sigma_{r,0}\otimes\sigma_{r,1}\otimes\cdots\otimes\sigma_{r,n-1})t\right)\\ &= \exp\left(-\frac{\mathrm{i}\alpha_r t}{\hbar}R\sigma R^\dagger\right)\\ &= R\exp\left(-\frac{\mathrm{i}\alpha_r t}{\hbar}\sigma\right)R^\dagger,\end{aligned}$$

where R is an unitary operator made of factors of H, S, and σ is a tensor products of $\mathbb{I}$'s and Z's.

Then, we have

$$\exp\left(-\mathrm{i}tZ^{\otimes c}\right) =$$

24. Quantum Error Correction

Definition 24.1. A $[n, k, d]$-*code* is an error correcting code with n physical bits, i.e., the number of bits in the code words (or the dimension of the space of code words), k is the number of logical bits, i.e., the number of bits of a decoded code word, and minimum distance k, i.e., the number of bit flips needed to get from a code word to another.

Remarks. (1) We must have $k < n$.

(2) Also, $d \leq n$ and we can correct $\lfloor (d-1)/2 \rfloor$ errors.

Errors in quantum computing are more complicated, even if all errors are one-qubit errors, as besides bit-flips, we also might have phase-flips.

Moreover, they can occur at any stage, e.g., quantum state preparation, applying Pauli gates, measuring, or even while idling! Note that we need to recover the *superposition*, not just the 0/1 state.

A 1-qubit error U_ϵ, with $\|U_\epsilon - \mathbb{I}\| < \mathcal{O}(\epsilon)$. For example,

$$U_\epsilon = \sqrt{1-\epsilon^2}\mathbb{I} + \epsilon\,(aX + bY + cZ)\,,$$

for some a, b, and c fixed. (So, X gives a bit-flip, Z gives a phase-flip, and Y gives both.)

So, we might start with $|0\rangle$ and end with $U_\epsilon\,|0\rangle$.

24.1. **Quantum 3-Qubit Code.**

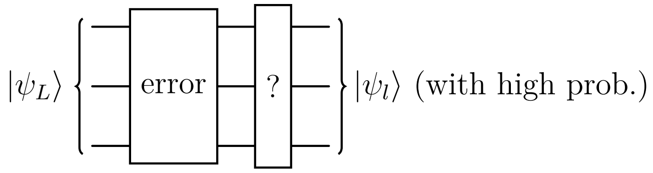
We have one logical qubit with three physical one:

$$\begin{aligned} |0_L\rangle &\overset{\text{def}}{=} |000\rangle\,, \\ |1_L\rangle &\overset{\text{def}}{=} |111\rangle\,. \end{aligned}$$

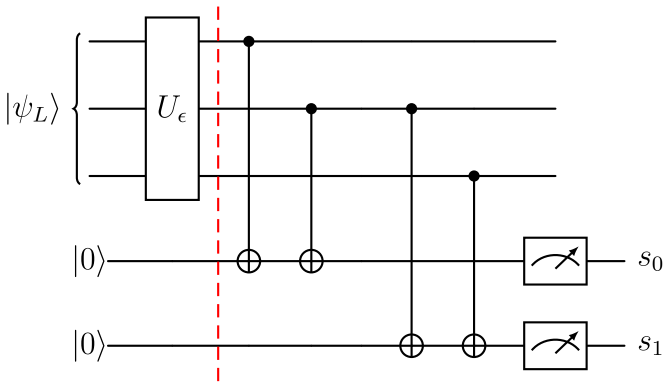
The code states have the form:

$$|\psi_L\rangle = \alpha\,|000\rangle + \beta\,|111\rangle\,, \qquad |\alpha|^2 + |\beta|^2 = 1.$$

We need something such that:



Here is the idea:



(These s_0 and s_1 are syndromes.)

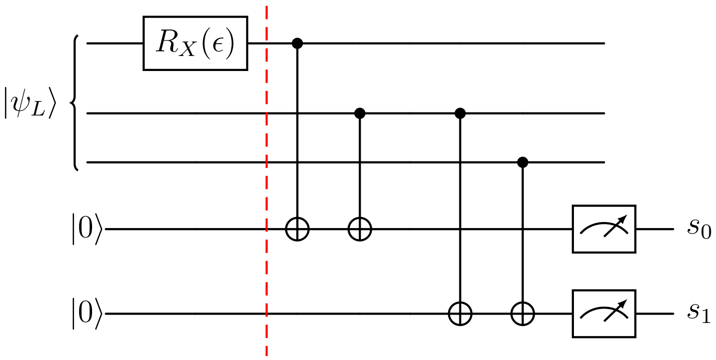
Here U_ϵ is the error. (We write it as gate, but it is unintentional, unlike the gates we apply to a circuit.) Suppose we get $|abc\rangle$ after the error (so, after the barrier/slice in the figure), then $s_0 = a + b$ and $s_1 = b + c$. *Assuming* that no errors occurred in the ancilla, we have

$$\begin{aligned} s_0 = 1 &\quad\iff\quad a \neq b, \\ s_1 = 1 &\quad\iff\quad b \neq c. \end{aligned}$$

Since we want $a = b = c$, if either s_0 or s_1 is 1, we have an error. More precisely,

(s_0, s_1)	$(0,0)$	$(1,0)$	$(0,1)$	$(1,1)$
Equalities	$a = b = c$	$a \neq b = c$	$a = b \neq c$	$b \neq a = c$
Error	None	a is off	c is off	b is off
Fix	None	X on $\|x_0\rangle$	X on $\|x_2\rangle$	X on $\|x_1\rangle$

To make it concrete, let's assume that the error is that one of the of the qubits are chosen, e.g., the first one, and $R_X(\epsilon)$ is acted on it, remembering that $R_X(\epsilon) = \mathrm{e}^{-\mathrm{i}\epsilon X/2} = \cos(\epsilon/2)\,\mathbb{I} - \mathrm{i}\sin(\epsilon/2) X$. So, $U_\epsilon = R_X(\epsilon) \otimes \mathbb{I} \otimes \mathbb{I}$:



Now,

$$\begin{aligned}(R_X(\epsilon) \otimes \mathbb{I} \otimes \mathbb{I}) \left|\psi\right\rangle &= (R_X(\epsilon) \otimes \mathbb{I} \otimes \mathbb{I})(\alpha \left|000\right\rangle + \beta \left|111\right\rangle) \\ &= \alpha(R_X(\epsilon) \left|0\right\rangle) \left|00\right\rangle + \beta(R_X(\epsilon) \left|1\right\rangle) \left|11\right\rangle \\ &= \alpha \cos(\epsilon/2) \left|000\right\rangle - \alpha \mathrm{i} \sin(\epsilon/2) \left|100\right\rangle \\ &\qquad + \beta \cos(\epsilon/2) \left|111\right\rangle - \beta \mathrm{i} \sin(\epsilon/2) \left|011\right\rangle .\end{aligned}$$

So, in the whole process, we get:

$$\begin{aligned}\left|\psi\right\rangle_3 \left|00\right\rangle &\mapsto \alpha \cos(\epsilon/2) \left|000\right\rangle \left|00\right\rangle - \alpha \mathrm{i} \sin(\epsilon/2) \left|100\right\rangle \left|00\right\rangle \\ &\qquad + \beta \cos(\epsilon/2) \left|111\right\rangle \left|00\right\rangle - \beta \mathrm{i} \sin(\epsilon/2) \left|011\right\rangle \left|00\right\rangle \\ &\mapsto \alpha \cos(\epsilon/2) \left|000\right\rangle \left|00\right\rangle - \alpha \mathrm{i} \sin(\epsilon/2) \left|100\right\rangle \left|10\right\rangle \\ &\qquad + \beta \cos(\epsilon/2) \left|111\right\rangle \left|00\right\rangle - \beta \mathrm{i} \sin(\epsilon/2) \left|011\right\rangle \left|10\right\rangle \\ &= \cos(\epsilon/2) \left(\alpha \left|000\right\rangle + \beta \left|111\right\rangle\right) \left|00\right\rangle \\ &\qquad - \mathrm{i} \sin(\epsilon/2) \left(\alpha \left|100\right\rangle + \beta \left|011\right\rangle\right) \left|10\right\rangle .\end{aligned}$$

We then measure the last two qubits. For small ϵ, it is likely we measure $s_0 = 0$ (with probability $\cos^2(\epsilon/2)$). And, in the case of our example, we are sure that s_1 will measure zero.

If we get $s_0 = 0$, then we get the correct $\alpha \left|000\right\rangle + \beta \left|111\right\rangle$ back. If we get $s_0 = 1$, we get $\alpha \left|100\right\rangle + \beta \left|011\right\rangle$ back, and we need to apply an X-gate to the first qubit.

Remark. Note that if there are two qubit-flips, the process above will "correct" incorrectly. More importantly, it does not correct Z-errors.

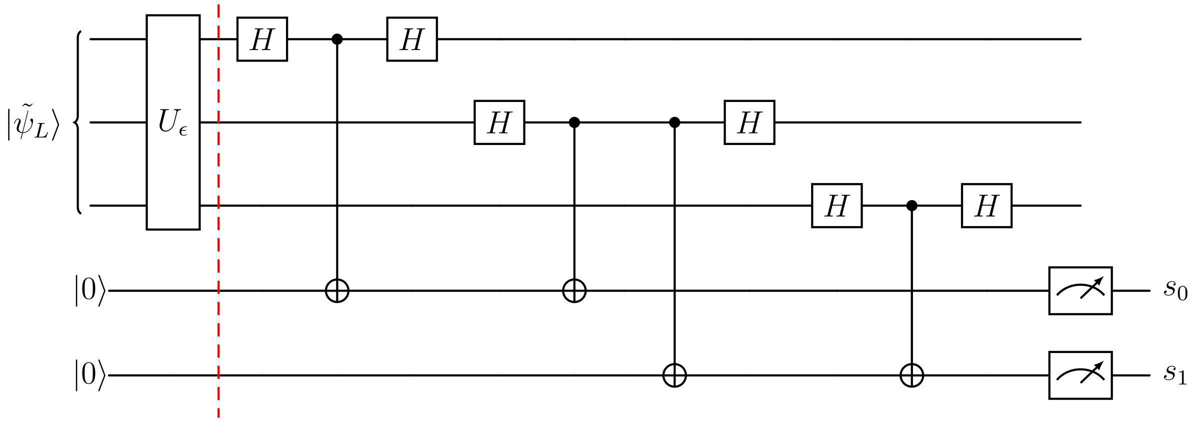
24.2. **Correcting Z-Errors.** Note that $Z = HXH$, and so

$$Z \left|+\right\rangle = \left|-\right\rangle , \qquad Z \left|-\right\rangle = \left|+\right\rangle .$$

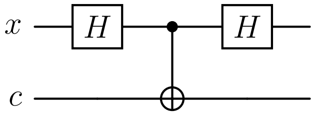
Thus, one can use a similar idea as above, where we apply $H = H^{-1}$ to make it into an X-error, and then another H to restore. More precisely, let

$$|\tilde{0}_L\rangle \overset{\text{def}}{=} |+++\rangle = H^{\otimes 3}|000\rangle,$$
$$|\tilde{1}_L\rangle \overset{\text{def}}{=} |---\rangle = H^{\otimes 3}|111\rangle,$$

and given $|\tilde{\psi}_L\rangle$, we can detect Z-errors with:



The basic circuit is:

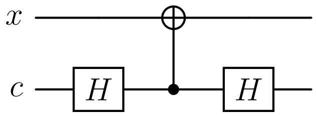


Then, the idea is that $Z|\pm\rangle = Z|\mp\rangle$, so

$$|+\rangle|0\rangle \xmapsto{H\otimes\mathbb{I}} |0\rangle|0\rangle \xmapsto{\text{CNOT}} |0\rangle|0\rangle \xmapsto{H\otimes\mathbb{I}} |+\rangle|0\rangle$$

$$|-\rangle|0\rangle \xmapsto{H\otimes\mathbb{I}} |1\rangle|0\rangle \xmapsto{\text{CNOT}} |1\rangle|1\rangle \xmapsto{H\otimes\mathbb{I}} |-\rangle|0\rangle$$

On the other hand, note that the circuit above is equivalent to



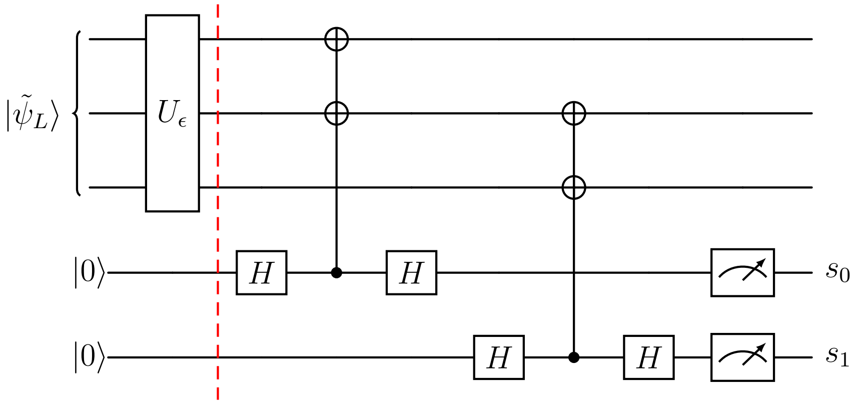
Indeed, for $x \in \{0, 1\}$, we have:

$$\begin{aligned}
|+\rangle\,|x\rangle &\mapsto |+\rangle\,|(-1)^x\rangle \\
&= \frac{1}{2}\left[(|0\rangle + |1\rangle) \otimes (|0\rangle + (-1)^x\,|1\rangle)\right] \\
&= \frac{1}{2}\left[|00\rangle + |10\rangle + (-1)^x\,|01\rangle + (-1)^x\,|11\rangle\right] \\
&\mapsto \frac{1}{2}\left[|00\rangle + |10\rangle + (-1)^x\,|11\rangle + (-1)^x\,|01\rangle\right] \\
&= |+\rangle\,|(-1)^x\rangle \\
&\mapsto |+\rangle\,|x\rangle\,,
\end{aligned}$$

and

$$\begin{aligned}
|-\rangle\,|x\rangle &\mapsto |-\rangle\,|(-1)^x\rangle \\
&= \frac{1}{2}\left[(|0\rangle - |1\rangle) \otimes (|0\rangle + (-1)^x\,|1\rangle)\right] \\
&= \frac{1}{2}\left[|00\rangle - |10\rangle + (-1)^x\,|01\rangle - (-1)^x\,|11\rangle\right] \\
&\mapsto \frac{1}{2}\left[|00\rangle - |10\rangle + (-1)^x\,|11\rangle - (-1)^x\,|01\rangle\right] \\
&= |-\rangle\,|(-1)^{x\oplus 1}\rangle \\
&\mapsto |-\rangle\,|x \oplus 1\rangle\,.
\end{aligned}$$

This second representation can save some gates, since $H^2 = \mathbb{I}$, in the implementation of the error detection. For instance, the full circuit above becomes:



24.3. **Shor's 9-Qubit Code.** (See also:A Methods Focused Guide to Quantum Error Correction and Fault-Tolerant Quantum Computation by Abdullah Khalid.)

Shor's 9-qubit code is a combination of a 3-qubit repetition code to correct X errors and Z errors. We have:

$$|0_L^{\text{Shor}}\rangle \overset{\text{def}}{=} \left[\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)\right]^{\otimes 3},$$
$$|1_L^{\text{Shor}}\rangle \overset{\text{def}}{=} \left[\frac{1}{\sqrt{2}}(|000\rangle - |111\rangle)\right]^{\otimes 3}.$$

Each bundle of three can be used to correct Z-errors, and the subbundles of three can be used to correct X-errors. It can correct arbitrary 1-qubit errors. It can also correct one X and one Z error on any different groups of 3 (bundles).

So, Shor's code is a $[[9, 1, 3]]$ code.

An alternative point of view: the physical Hilbert space is $\mathcal{H}_{\text{phys}} = (\mathbb{C}^2)^{\otimes 9}$ while the logical Hilbert space is $\mathcal{H}_{\text{logical}} = \text{span}\left(|0_L^{\text{Shor}}\rangle, |1_L^{\text{Shor}}\rangle\right)$. We also can describe $\mathcal{H}_{\text{logical}}$ as the simultaneous $+1$-eigenspace of the group

$$S \overset{\text{def}}{=} \langle Z_{0,1}, Z_{1,2}, Z_{3,4}, Z_{4,5}, Z_{6,7}, Z_{7,8}, X_{0,1,2,3,4,5}, X_{3,4,5,6,7,8}\rangle$$

where $Z_{i,j}$ is the tensor product of nine $\mathbb{I}$ and Z, with Z appearing in the i-th and j-th components only (counting from 0), and similarly for the $X_{i,j,\dots}$ above. In other words,

$$\mathcal{H}_{\text{logical}} = \operatorname{stab}(S) \overset{\text{def}}{=} \{|\psi\rangle \in \mathcal{H}_{\text{phys}} : A\,|\psi\rangle = |\psi\rangle \text{ for all } A \in S\}.$$

Note that S is commutative and for all $A \in S$, we have that $A^2 = \mathbb{I}$. In fact, one can prove that $S \cong \mathbb{F}_2^8$.

Definition 24.2. A group $S = \langle P_1, P_2, \dots, P_k\rangle$ is an n-qubit *stabilizer group* if

(1) P_i is a tensor product of $\mathbb{I}$'s, X's, and Z's, i.e., it is a *Pauli string*;
(2) S is commutative (i.e., $P_iP_j = P_jP_i$ for all i and j). (This is needed for the sake of diagonalization.)

Note that for all $P \in S$, then, we have that $P^2 = \mathbb{I}$.

We then define

$$\mathcal{H}^S_{\text{logical}} \overset{\text{def}}{=} \operatorname{stab}(S) \overset{\text{def}}{=} \{|\psi\rangle \in \mathcal{H}_{\text{phys}} : P\,|\psi\rangle = |\psi\rangle \text{ for all } P \in S\}.$$

Remark. If $S = \langle P_1, P_2, \dots, P_k\rangle$, with the P_i's *independent* (i.e., removing any P_i would yield a different group, or, alternatively, if S as an $\mathbb{F}_2$-vector space has dimension k, with the P_i's forming a basis), then $\mathcal{H}^S_{\text{logical}}$ is a Hilbert space of dimension 2^{n-k}.

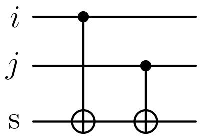
Remembering that P_i is a tensor product of $\mathbb{I}$, X, and X, then if, say, an X-error U occurs on a qubit for which P_i has a Z factor, then $P_iU = -UP_i$, since $XZ = -ZX$. Then, for $|\psi_L\rangle \in \mathcal{H}^S_{\text{logical}}$, we have

$$P_i(U\,|\psi_L\rangle) = -U(P_i\,|\psi_L\rangle) = -(U\,|\psi_L\rangle).$$

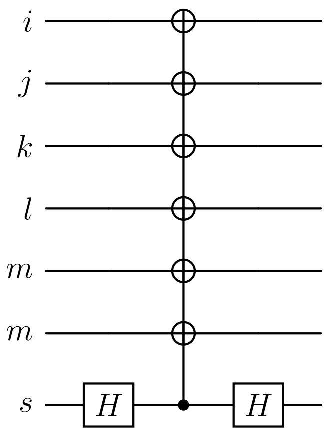
So, the erroneous state is not a $+1$ eigenvector of P_i, but a -1 eigenvector!

Thus, if we can measure eigenvalues of P_i, we can get syndromes for errors.

Then, back to Shor's Code, with $Z_{i,j}$ we can detect X errors on i or j.



Similarly, with $X_{i,j,k,l,m,n}$ we can detect Z errors on the corresponding indices.



Remark. Shor's 9-qubit code is "low quality", since we need 9 qubits for a single logical qubit and still can fix only one error, but it illustrates important ideas.

There are quantum *low density parity check* (LDPC) that are $[[n, k, d]]$ with $k \approx n(1 - \epsilon_1)$ and $d \approx n(1 - \epsilon_2)$, which scale well.

24.4. 5 **Qubit Code.** Here is an example of a more efficient code, using only 5 qubits per logical qubit:

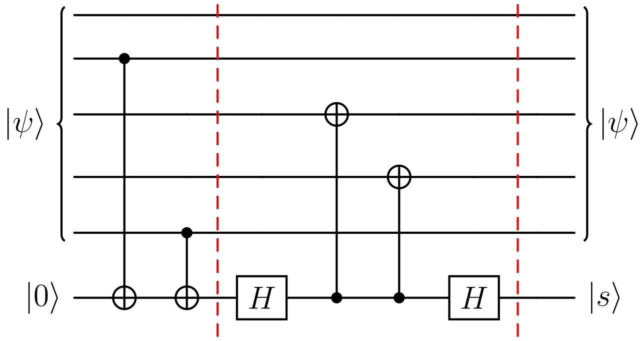
Example 24.3. Consider $S \stackrel{\text{def}}{=} \langle P_1, P_2, P_3, P_4 \rangle$ with:

$$P_1 \stackrel{\text{def}}{=} \mathbb{I} \otimes X \otimes Z \otimes Z \otimes X,$$
$$P_2 \stackrel{\text{def}}{=} X \otimes \mathbb{I} \otimes X \otimes Z \otimes Z,$$
$$P_3 \stackrel{\text{def}}{=} Z \otimes X \otimes \mathbb{I} \otimes X \otimes Z,$$
$$P_4 \stackrel{\text{def}}{=} X \otimes Z \otimes Z \otimes X \otimes \mathbb{I}.$$

(Note how the P_i's are "rotations" of P_1 as strings.) Then S is a 5-qubit stabilizer group. It yields (via the simultaneous $+1$-eigenspaces for the P_i's) a $[[5, 1, 3]]$ code.

24.5. **Extracting Syndromes.** A syndrome s associated to some P in a stabilizer group, is either 0 or 1, by $P\,|\psi\rangle = (-1)^s\,|\psi\rangle$. But how do we measure it.

For example, consider $P_1 = \mathbb{I} \otimes X \otimes Z \otimes Z \otimes X$, as in the 5-qubit code. We add CNOT gates targeting the syndrome where we have X's and CNOT gates, surrounded by H gates, controlled by the syndrome, where we have Z's:



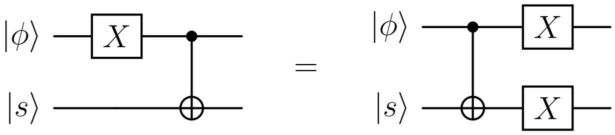
This is just like the Shor's code, as described above.

If $|\psi\rangle_5 = |\phi\rangle_1\,(H\,|s_1\rangle_1)\,|s_2\rangle_1\,|s_3\rangle_1\,(H\,|s_4\rangle_1)$, with $|\phi\rangle$ an arbitrary 1-qubit state and $s_i \in \mathbb{F}_2$, then $|\psi\rangle$ is an eigenvector of P_1 (remembering that the only possible eigenvalues are ±1). Indeed, since $XH = HZ$ and $Z\left|s\right\rangle = (-1)^s \left|s\right\rangle$, we have

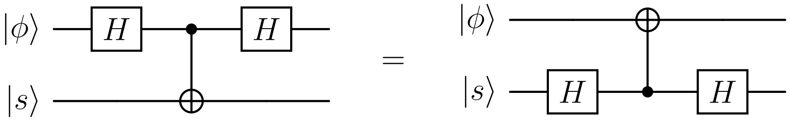
$$\begin{aligned}
P_1 \left|\psi\right\rangle &= \left|\phi\right\rangle (XH \left|s_1\right\rangle) Z \left|s_2\right\rangle Z \left|s_3\right\rangle (XH \left|s_4\right\rangle) \\
&= \left|\phi\right\rangle (HZ \left|s_1\right\rangle) (-1)^{s_2} \left|s_2\right\rangle (-1)^{s_3} \left|s_3\right\rangle (HZ \left|s_4\right\rangle) \\
&= \left|\phi\right\rangle (H(-1)^{s_1} \left|s_1\right\rangle) (-1)^{s_2} \left|s_2\right\rangle (-1)^{s_3} \left|s_3\right\rangle (H(-1)^{s_4} \left|s_4\right\rangle) \\
&= (-1)^{s_1+s_2+s_3+s_4} \left|\phi\right\rangle (H \left|s_1\right\rangle) \left|s_2\right\rangle \left|s_3\right\rangle (H \left|s_4\right\rangle) \\
&= (-1)^{s_1+s_2+s_3+s_4} \left|\psi\right\rangle .
\end{aligned}$$

Note that these $\left|\psi\right\rangle$'s, as above, with $\left|\phi\right\rangle$ as $\left|0\right\rangle$ and $\left|1\right\rangle$, give a basis for the whole space, so this is a basis of eigenvectors.

Now, noticing that if $s \in \mathbb{F}_2$ we have



and, as observed above



Hence, breaking the process with the slices/barriers marked in the circuit above, we have

$$\begin{aligned}
\left|\psi\right\rangle_5 \left|0\right\rangle_1 &= \left|\phi\right\rangle (H \left|s_1\right\rangle) \left|s_2\right\rangle \left|s_3\right\rangle (H \left|s_4\right\rangle) \left|0\right\rangle \\
&\mapsto \left|\phi\right\rangle (H \left|s_1\right\rangle) \left|s_2\right\rangle \left|s_3\right\rangle (H \left|s_4\right\rangle) \left|s_1 \oplus s_4\right\rangle \\
&\mapsto \left|\phi\right\rangle (H \left|s_1\right\rangle) \left|s_2\right\rangle \left|s_3\right\rangle (H \left|s_4\right\rangle) \left|s_1 \oplus s_4 \oplus s_2 \oplus s_3\right\rangle .
\end{aligned}$$

(Here it seems we've used only the second equality of circuits above.)

Hence, indeed, the circuit above gives $|s\rangle = |s_1 \oplus s_2 \oplus s_3 \oplus s_4\rangle$.

With P_2, P_3, and P_4, we narrow down the $+1$-eigenvectors (as so far we only have that $s_4 = s_1 + s_2 + s_3$), giving our logical qubits, and the corresponding syndromes would help detect errors.

But how do we now correct an error using the syndromes?

Remark. After measuring syndromes an error, that is not necessarily a Pauli error, but after measuring the syndromes they become a Pauli error. For instance, if we start with $|\psi\rangle$ and have an error U_ϵ that is a tensor product of 1-qubit gates (so, no entangling), say, $U_\epsilon = \cos(\epsilon)\mathbb{I} + \sin(\epsilon)X_i$, then after measuring the syndromes, $U_\epsilon |\psi\rangle$ becomes $E|\psi\rangle$, where E is either $\mathbb{I}$ or X_i. Even though U_ϵ is very close to the identity, it could collapse to X_i instead, but we still can fix it easily, as it is a "simple" error.

The process to go from the syndromes to the recovery is the *decoding*: if we have a stabilizer group $S = \langle P_1, \dots, P_k\rangle$, then the syndromes are $|s\rangle_k$ with $s \in \mathbb{F}_2^k$. Then, given an Pauli string error E, if $\sigma(E) = (s_1, \dots, s_k) \in \mathbb{F}_2^k$ is the syndrome generated by the error, then

$$EP_i = (-1)^{s_i} P_i E.$$

The *decoder* is a map:

$$\mathcal{D} : \mathbb{F}_2^k \to \mathcal{P}_n$$

where $\mathcal{P}_n$ is the *Pauli group* (the group of Pauli strings of length n) such that for any error $E \in \mathcal{P}_n$, $\mathcal{D}(\sigma(E)) = E' \equiv E \pmod{S}$, i.e., $E = E'P$ for some $P \in S$. Noting that S fixes the logical qubits (and therefore code states), by definition, we have that the actions of E and E' on logical qubits are identical. So, the recovery is applying E' to the erroneous state, since $E'E|\psi\rangle = PEE|\psi\rangle = P|\psi\rangle = |\psi\rangle$, where $|\psi\rangle$ is a code state.

24.6. **Steane's Code.** Steane's code is a $[[7,1,3]]$ code, defined with

$$S = \langle X_{0356}, X_{1346}, X_{2456}, Z_{0356}, Z_{1346}, Z_{2456} \rangle.$$

Stabilizer codes that have only X and Z-type stabilizers are called *CSS codes*.

Another example of CSS codes are the Kitaev's toric codes, which are $[[2k^2, 2, k]]$, for $k \in \mathbb{Z}_{>0}$.

Notation

Index