4.5 Universal quantum gates

A small set of gates (AND, OR, NOT) is *universal* for classical computation. In fact, since the Toffoli gate is universal for classical computation, quantum circuits subsume classical circuits. A similar universality result is true for quantum computation, where a set of gates is said to be *universal for quantum computation* if any unnitary operation may be approximated to arbitrary accuracy bye a quantum circuit involving only those gates.

Any unitary operation can be approximated to arbitrary accuracy using Hadamard, phase, CNOT, and $\pi/8$ gates.

1. An arbitrary operator may be expressed *exactly* as a product of unitary operators that each acts non-trivially only on a subspace spanned by two computational basis states.
2. The second construction combines the first construction with the results of the previous section to show that an arbitrary unitary operator may be expressed *exactly* using single qubit and CNOT gates.
3. Single qubit operation may be approximated to arbitrary accuracy using the Hadamard, phase, and $\pi/8$ gates. This in turn implies that any unitary operation can be approximated to arbitrary accuracy using Hadamard, phase, CNOT, and $\pi/8$ gates.

4.5.1 Two-level unitary gates are universal

Consider a unitary matrix U which acts on a d-dimensional Hilbert space. We explain how U may be decompsed into a product of *two-level unitary matrices*; unitary matrices which act non-trivially only on two-or-fewer vector components.

Suppose that U has the form

$$U = \begin{bmatrix} a & d & g \\ b & e & h \\ c & f & j \end{bmatrix}.$$

We will find two-level unitary matrices $U_1, \ldots, U_3$ such that

$$U_3 U_2 U_1 U = I.$$

It follows that

$$U = U_1^\dagger U_2^\dagger U_3^\dagger.$$

U_1, U_2 and U_3 are all two-level unitary matrices, and it is easy to see that their inverse, $U_1^\dagger$, $U_2^\dagger$ and $U_3^\dagger$ are also two-level unitary matrices.

Use the following procedure to construct U_1: if $b = 0$ then set

$$U_1 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

if $b \neq 0$ then set

$$U_1 \equiv \begin{bmatrix} \frac{a^*}{\sqrt{|a|^2+|b|^2}} & \frac{b^*}{\sqrt{|a|^2+|b|^2}} & 0 \\ \frac{b}{\sqrt{|a|^2+|b|^2}} & \frac{-a}{\sqrt{|a|^2+|b|^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Thus

$$U_1 U = \begin{bmatrix} a' & d' & g' \\ 0 & e' & h' \\ c' & f' & j' \end{bmatrix}.$$

The key point to note is that the middle entry in the left had comlumn is zero.

Now apply a similar procedure to find a two-level matrix U_2 such that $U_2 U_1 U$ has no entry in the *bottom left* corner. That is, if $c' = 0$ we set

$$U_2 \equiv \begin{bmatrix} a'^* & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

while if $c' \neq 0$ then we set

$$U_2 \equiv \begin{bmatrix} \frac{a'^*}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{c'^*}{\sqrt{|a'|^2+|c'|^2}} \\ 0 & 1 & 0 \\ \frac{c'}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{-a'}{\sqrt{|a'|^2+|c'|^2}} \end{bmatrix}.$$

In either case, when we carry out the matrix multiplication we find that

$$U_2 U_1 U = \begin{bmatrix} 1 & d'' & g'' \\ 0 & e'' & h'' \\ 0 & f'' & j'' \end{bmatrix}.$$

Since U, U_1 and U_2 are unitary, it follows that $U_2 U_1 U$ is unitary, and thus $d'' = g'' = 0$ since the first row must have norm 1. Finally, set

$$U_3 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & e''^* & f''^* \\ 0 & h''^* & j''^* \end{bmatrix}.$$

It is now easy to verify that $U_3 U_2 U_1 U = I$, and thus $U = U_1^\dagger U_2^\dagger U_3^\dagger$, which is a decomposition of U into two-level unitaries.

More generally, suppose U acts on a d-dimensional space. Then, in a similar fashion to the 3×3 case, we can find two-level unitary matrices $U_1, \ldots, U_{d-1}$ such that the matrix $U_{d-1} U_{d-2} \ldots U_1 U$ has a one in the top left hand corner, and all zeroes elsewhere in the first row and column. We then repeat this procedure for the $d-1$ by $d-1$ unitary submatrix in the lower right hand corner of $U_{d-1} U_{d-2} \ldots U_1 U$, ans so on, with the end result that an arbitrary $d \times d$ unitary matrix may be written

$$U = V_1 \ldots V_k$$

where the matrices V_i are two-level unitary matrices, and $k \leq (d-1) + (d-2) + \ldots + 1 = d(d-1)/2$.

For specific unitry matrices, it may be possible to find much more efficient decompositions, but as you will now show there exist matrices which cannot be docomposed as a product of fewer than d-1 two-level unitary matrices!

4.5.2 Single qubit and CNOT gates are universal

Now we show that single qubit and CNOT gates together can be used to implement an arbitrary two-level unitary operation on the state space of n qubits.

Suppose U is a two-level unitary matrix on an n qubit quantum computer. Suppose in particular that U acts non-trivially on the space spanned by thecomputational basis state $|s\rangle$ and $|t\rangle$, where $s = s_1 \ldots s_n$ and $t = t_1 \ldots t_n$ are the binary expansions for s and t. Let $\tilde{U}$ be the non-trivial 2×2 unitary submatrix of U; $\tilde{U}$ can be thought of as a unitary operator on a single qubit.

Our immediate goal is to construct a circuit implementing U, built from single qubit and CNOT gates. To do this, we need to make use of *Gray codes*. Suppose we have distinct binary numbers, s and t. A *Gray code* connecting s and t is a sequence of binary numbers, starting with s and concluding with t, such that adjacent mumbers of the list differ in exactly one bit. For instance, with $s = 101001$ and $t = 110011$ we have the Gray code

$$\begin{array}{cccccc} 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{array}$$

Let g_1 through g_m be the elements of a Gray code connecting s and t, with $g_1 = s$ and $g_m = t$. Note that we can always find a Gray code such that $m \leq n + 1$ since s and t can differ in at most n locations.

The basic idea of the quantum circuit implementing U is to perform a sequence of gates effecting the state changes $|g_1\rangle \rightarrow |g_2\rangle \rightarrow \ldots \rightarrow |g_{m-1}\rangle$, then to perform a controlled-$\tilde{U}$ operation, with the target qubit located at the single bit where g_{m-1} and g_m differ, and then to undo the first stage, transforming $|g_{m-1}\rangle \rightarrow |g_{m-2}\rangle \rightarrow \ldots \rightarrow |g_1\rangle$.

A more precies description of the implementation is as follows. The first step is to swap the state $|g_1\rangle$ and $|g_2\rangle$. Suppose g_1 and g_2 differ at the ith qubit, donditional on the values of the other qubits being identical to those in both g_1 and g_2. Next we use a controlled operation to swap $|g_2\rangle$ and $|g_3\rangle$. We continue in this fashion until we swap $|g_{m-2}\rangle$ with $|g_{m-1}\rangle$.

$$\begin{array}{rcl} |g_1\rangle & \rightarrow & |g_{m-1}\rangle \\ |g_2\rangle & \rightarrow & |g_1\rangle \\ |g_{m-1}\rangle & \rightarrow & |g_{m-2}\rangle \end{array}$$

Suppose g_{m-1} and g_m differ in the jth bit. We apply a controlled-$\tilde{U}$ operation with the jth qubit as target, conditional on the other qubits having the same values as appear in both g_m and g_{m-1}. Finally, we complete the U operation by undoing the swap operations.

eg)

Suppose we wish to implement the two-level unitary transformation

$$U = \begin{bmatrix} a & 0 & 0 & 0 & 0 & 0 & 0 & c \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ b & 0 & 0 & 0 & 0 & 0 & 0 & d \end{bmatrix}$$

a, b, c and d are any complex numbers such that $\tilde{U} \equiv \begin{bmatrix} a & c \\ b & d \end{bmatrix}$ is a unitary matrix. We write a Gray code

connecting 000 and 111:

A	B	C
0	0	0
0	0	1
0	1	1
1	1	1

From this we read off the required circuit, shown in Figure 4.16. The first two gates shuffle the states so that $|000\rangle$ gets swapped with $|011\rangle$. Next the operation $\tilde{U}$ is applied to the first qubit of the states $|011\rangle$ and $|111\rangle$, conditional on the second and third qubits being in the state $|11\rangle$. Finally, we unshuffle the states, ensuring that $|011\rangle$ gets swapped back with the state $|000\rangle$.

Returning to the general case, we see that implementing the two-level unitary operation U requires at most $2(n-1)$ controlled operations to swap $|g_1\rangle$ with $|g_{m-1}\rangle$ and then back again. Each of thes controlled operations can be realized using $O(n)$ single qubit and CNOT gates; the controlled-$\tilde{U}$ operation also requires $O(n)$ gates. Thus, implementing U requires On^2 single qubit and CNOT gates.

2^n-dimensional state space of n qubits may be written as a product of $O(2^{2n}) = O(4^n)$ two-level unitary operations. Combining these results, we see that an arbitrary unitary operation on n qubits can be implemented using a circuit containing $O(n^2 4^n)$ single qubit and CNOT gates. Obviously, this construction does not provide terribly efficient quantum circuits. However, the construction is close to optimal in the sense that there are unitary operations that require an exponential number of gates to implement. Thus, to find fast quantum algorithms we will clearly need a different approach that is taken in the universality construction.

4.5.3 A discrete set of universal operations

No straightforward method is known to implement all these gates in a fashion thich is resistant to errors. Fortunately, in this section we'll find a discrete set of gates which can be used to perform universal quantum computation, and in Chapter 10 we'll show how to perform these gates in an error-resistant fashion, using quantum error-correcting codes.

Approximating unitary operators

A discrete set of gates can't be used to implement an arbitrary unitary operation *exactly*, since the set of unitary operations is continuous. Rather, it turns out that a discrete set can be used to *approximate* any unitary operation.

Suppose U and V are two unitary operators on the same state space. U is the target unitary operator that we wish to implement, and V is the unitary operator that is actually implemented in practice. We define the *error* when V is implemented instead of U by

$$E(U,V) \equiv \max_{|\psi\rangle} \|(U-V)|\psi\rangle\|,$$

where the maximum is oever all normalized quantum states $|\psi\rangle$ in the state space.

Box 4.1: Approximating quantum circuits

Suppose a quantum system starts in the state $|\psi\rangle$, and we perform either the unitary operation U, or V. Following this, we perform a measurement. Let M be a POVM element associated with the measurement, and let P_U (or P_V) be the probability of obtaining the corresponding measurement outcome if the operation U (or V) was performed. Then

$$|P_U - P_V| = |\langle\psi|U^{\dagger MU}|\psi\rangle - \langle\psi|V^\dagger MV|\psi\rangle|.$$

Let $|\Delta\rangle \equiv (U-V)|\psi\rangle$.

$$\begin{aligned} |P_U - P_V| &= |\langle\psi|U^\dagger M|\Delta\rangle - \langle\Delta|MV|\psi\rangle| \\ &\leq |\langle\psi|U^\dagger M|\Delta\rangle| + |\langle\Delta|MV|\psi\rangle| \\ &\leq \||\Delta\rangle\| + \||\Delta\rangle\| \\ &\leq 2E(U,V). \end{aligned}$$

The inequality $|P_U - P_V| \leq 2E(U,V)$ gives quantitative expression to the idea that when the error $E(U,V)$ is small, the difference in probabilities between measurement outcomes is also small. Suppose we perform a sequance $V_1, V_2, \ldots, V_m$ of gates intended to approximate some other sequence of gates, $U_1, U_2, \ldots, U_m$. Then it turns out that the error caused by the entire sequence of imperfect gate is at most the sum of the errors in the individual gates,

$$E(U_m U_{m-1} \ldots U_1, V_m V_{m-1} \ldots V_1) \leq \sum_{j=1}^{m} E(U_j, V_j).$$

To prove this we start with the case $m = 2$. Note that for some state $|\psi\rangle$ we have

$$\begin{aligned} E(U_2U_1, V_2V_1) &= \|(U_2U_1 - V_2V_1|\psi\rangle)\| \\ &= \|(U_2U_1 - V_2U_1)|\psi\rangle + (V_2U_1 - V_2V_1)|\psi\rangle\|. \end{aligned}$$

Using the triangle inequality $\||a\rangle + |b\rangle\| \leq \||a\rangle\| + \||b\rangle\|$, we obtain

$$\begin{aligned} E(U_2U_1, V_2V_1) &= \|(U_2 - V_2)U_1|\psi\rangle\| + \|V_2(U_1 - V_1)|\psi\rangle\| \\ &= E(U_2, V_2) + E(U_1, V_1), \end{aligned}$$

which was the desired result. The result for general m follows by induction.

This measure of error has the interpretation that if $E(U,V)$ is small, then any measurement performed on the state $V|\psi\rangle$ will give approximately the same measurement statistis as a measurement of $U|\psi\rangle$, for any initial state $|\psi\rangle$. More precisely, we show that if M is a POVM element in an arbitrary POVM, and P_U (or P_V) is the probability of obtaining this outcome if U (or V) were performed with a starting state $|\psi\rangle$, then

$$|P_U - P_V| \leq 2E(U,V).$$

Thus, if $E(U,V)$ is small, then measurement outcomes occur with similar probabilities, regardless of whether U or V were performed. If we perform a sequence of gates $V_1, \ldots, V_m$ intended to approximate some other sequence of gates $U_1, \ldots, U_m$, then the errors add at most linearly,

$$E(U_m U_{m-1} \ldots U_1, V_m V_{m-1} \ldots V_1) \leq \sum_{j=1}^{m} E(U_j, V_j).$$

Above approximation results are extremely useful. Suppose we wish to perform a quantum circuit containing m gates, U_1 through U_m. Unfortunately, we are only able to approximate the gate U_j by the gate V_j. In

order that the probabilities tolerance $\Delta > 0$ of the correct probabilities, it suffices that $E(U_j, V_j) \leq \Delta/(2m)$, by above results.

Universality of Hadamard + phase + CNOT + $\pi/8$ gates

We're going to consider two different discrete sets of gates, both of which are universal. The first set, the *standard set* of universal gates, consists of the Hardamard, phase, controlled-NOT and $\pi/8$ gates. The second set of gates we consider consists of the Hadamard gates, phase gate, the controlled-NOT gate, and the Toffoli gate.

We begin the universality proof by showing that the hadamard and $\pi/8$ gates can be used to approximate any singel qubit unitary operation to arbitrary accuracy. Consider the gates T and HTH. T is, up to an unimportant global phase, a rotation by $\pi/4$ radians around the $\hat{z}$ axis on the Bloch sphere, while HTH is a rotation by $\pi/4$ radians around the $\hat{x}$ axis on the Bloch sphere. Composing these two operations gives, up to a global phase,

$$\begin{aligned} \exp\left(-i\frac{\pi}{8}Z\right)\exp\left(-i\frac{\pi}{8}X\right) &= \left[\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}Z\right]\left[\cos\frac{\pi}{8} - i\sin\frac{\pi}{8}X\right] \\ &= \cos^2\frac{\pi}{8}I - i\left[\cos\frac{\pi}{8}(X+Z) + \sin\frac{\pi}{8}Y\right]\sin\frac{\pi}{8} \end{aligned}$$

This is a rotation of the Bloch sphere about an axis along $\vec{n} = (\cos\frac{\pi}{8}, \sin\frac{\pi}{8}, \cos\frac{\pi}{8})$ with corresponding unit vector $\hat{n}$, and through an angle θ defined by $\cos(\theta/2) \equiv \cos^2\frac{\pi}{8}$.That is, using only the Hadamard and $\pi/8$ gates we can construct $R_{\hat{n}}(\theta)$. Moreover, this θ can be shown to be an irrational multiple of 2π. Proving this latter fact is a little beyond our scope.

Next, we show that repeated iteration of $R_{\hat{n}}(\theta)$ can be used to approximate to arbitrary accuracy any rotation $R_{\hat{n}}(\alpha)$. To see this, let $\delta > 0$ be the desired accuracy, and let N be an integer larger thatn $2\pi/\delta$. Define θ_k so that $\theta_k \in [0, 2\pi)$ and $\theta_k = (k\theta) \bmod 2\pi$. Then the pigeonhole principle implies that there are distinct j and k in the range $1, \ldots, N$ such that $|\theta_k - \theta_j| \leq 2\pi/N < \delta$. Assume that $k > j$, so we have $\theta_{k-j} < \delta$. Since $j \neq k$ and θ is an irrational multiple of 2π we must have $\theta_{k-j} \neq 0$. It follows that the sequence $\theta_{l(k-j)}$ fills up the interval $[0, 2\pi)$ as l is varied, so that adjacent members of the sequence are no more than δ apart. It follows that for any $\epsilon > 0$ there exists an n such that

$$E(R_{\hat{n}}(\alpha), R_{\hat{n}}(\theta)^n) < \frac{\epsilon}{3}.$$

We are now in position to verify that any single qubit operation can be approximated to arbitrary accuracy using the Hadamard and $\pi/8$ gates. For any α

$$HR_{\hat{n}}(\alpha)H = R_{\hat{m}}(\alpha),$$

where $\hat{m}$ is a unit vector in the direction $(\cos\frac{\pi}{8}, -\sin\frac{\pi}{8}, \cos\frac{\pi}{8})$, from which it follows that

$$E(R_{\hat{m}}(\alpha), R_{\hat{m}}(\theta)^n) < \frac{\epsilon}{3}.$$

But an arbitrary unitary U on a single qubit may be written as

$$U = R_{\hat{n}}(\beta)R_{\hat{m}}(\gamma)R_{\hat{n}}(\delta),$$

upto an unimportant global phase shift. Thus

$$E(U, R_{\hat{n}}(\theta)^{n_1}HR_{\hat{n}}(\theta)^{n_2}HR_{\hat{n}}(\theta)^{n_3}) < \epsilon.$$

That is, given any single qubit unitary operator U and any $\epsilon > 0$ it is possible to approximate U to within ϵ using a circuit composed of Hadamard gates and $\pi/8$ gates alone.

Since the $\pi/8$ and Hadamard gates allow us to approximate any single qubit unitary operator, it follows from the arguments of Section 4.5.2 that we can approximate any m gate quantum circuit, as follows. Given a quantum circuit containing m gates, we may approximate it using Hadamard, controlled-NOT and $\pi/8$ gates. If we desire an accuracy of *epsilon* for the entire circuit, then this may be achieved by approximating each single qubit unitary using the above procedure to within ϵ/m and applying the chaining inequality to obtain an accuracy of ϵ for the entire circuit.

How efficient is this procedure for approximating quantum circuits using a discrete set of gates?

eg) Suppose that approximating an arbitrary single qubit unitary to within a distance ϵ were to require $\Omega(2^{1/\epsilon})$ gates from the discrete set. Then to approximate the m gate quantum circuit considered in the previous paragraph would require $\Omega(m2^{m/\epsilon})$ gates, an exponential increase over the original circuit size! Fortunately, the rate of convergence is much better than this. Intuitively, it is plausible that the sequence of angles θ_k 'fills in' the interval $[0, 2\pi)$ in a more or less uniform fashion, so that to approximate an arbitrary single qubit gate ought to take roughly $\Theta(1/\epsilon)$ gats from the discrete set.

# of gates to approximate an m gate circuit to accuracy *epsilon* $\rightarrow \Theta(m^2/\epsilon)$.

However, a much faster rate of convergence can be proved. The *Solovay-Kitaev theorem* implies that an arbitrary single qubit gate may be approximated to an accuracy ϵ using $O(\log^c(1/\epsilon))$ gates from our discrete set, where c is a constant approximately equal to 2. $\rightarrow O(m \log^c(m/\epsilon))$

To sum up, we have shown that the Hadamard, phase, controlled-NOT and $\pi/8$ gates are universal for quantum computation in the sense that given a circuit containing CNOTs and arbitrary single qubit unitaries it is possible to simulate this circuit to good accuracy using only this discrete set of gates.

4.5.4 Approximating arbitrary unitary gates is generically hard

We've seen that any unitary transformation on n qubits can be built up out of a small set of elementary gates. Is it always possible to do this efficiently? In fact, most unitary transformations can only be implemented very inefficiently.

How myny gates does it take to generate an arbitrary state of n qubits? A simple counting argument shows that this requires exponentially many operations, in general.

Suppose we have g different types of gates available, and each gate works on at most f input qubits. f and g may be considered to be constants. Suppose we have a quantum circuit containg m gates, starting from the computational basis state $|0\rangle^{\otimes n}$. For any particular gate in the circuit there are therefore at most $\left[\begin{array}{c} n \\ g \end{array} \right]^g = O(n^{fg})$ possible choices. It follows that at most $O(n^{fgm})$ different states may be computed using m gates.

Suppose we wish to approximate a particular state, $|\psi\rangle$, to within a distance *epsilon*. The space of state vectors of n qubits can be regarded as just the unit $(2^{n+1} - 1)$-sphere. Suppose the n qubit state has amplitudes $\psi_j = X_j + Y_j$ of the jth amplitude. The normalization condition for quantum states can be written $\sum_j (X_j^2 + Y_j^2) = 1$, which is just the condition for a point to be on the unit sphere in 2^{n+1} real dimensions, that is, the unit $(2^{n+1} - 1)$-sphere.

Similarly, the surface area of radius ϵ near $|\psi\rangle$ is approximately the same as the volume of a $(2^{n+1} - 2)$-sphere of radius ϵ. Using the formular $S_k(r) = 2\pi^{(k+1)/2} r^k / \Gamma((k+1)/2)$ for the surface area of a k-sphere of radius

r, and $V_k(r) = 2\pi^{(k+1)/2} r^{k+1}/[(k+1)\Gamma((k+1)/2)]$ for the volume of a k-sphere of radius r, we see that the number of pathes needed to cover the state space goes like

$$\frac{S_{2^{n+1}-1}(1)}{V_{2^{n+1}-2}(\epsilon)} = \frac{\sqrt{\pi}\Gamma(2^n - \frac{1}{2})(2^{n+1}-1)}{\Gamma(2^n)\epsilon^{2^{2n+1}-1}},$$

where Γ is the usual generalization of the factorial function. But $\Gamma(2^n - 1/2) \geq \Gamma(2^n)/2^n$, so the number of patches required to cover the space is at least

$$\Omega\left(\frac{1}{\epsilon^{2^{n+1}-1}}\right).$$

Recall that the number of pathces which can be reached in m gates is $O(n^{fgm})$, so in order to reach all the ϵ-patches we must have

$$O(n^{fgm}) \geq \Omega\left(\frac{1}{\epsilon^{2^{n+1}-1}}\right)$$

which gives us

$$m = \Omega\left(\frac{2^n \log(1/\epsilon)}{log(n)}\right).$$

That is, there are states of n qubits which take $\Omega(2^n \log(1/\epsilon)/\log(n))$ operations to approximate to within a distance ϵ. This is exponential in n, and thus is 'difficult', in the sense of computational complexity.

4.5.5 Quantum computational complexity