

## Quantum Circuits

Here we are describing quantum algorithms, the language of **quantum circuits** – assemblies of discrete sets of components which describe computational procedures. This construction will enable us to quantify the cost of an algorithm in terms of things like the total number of gates required, or the circuit depth. The circuit language also comes with a toolbox of tricks that simplifies algorithm design and provides ready conceptual understanding.

### Quantum Algorithms

The spectacular promise of quantum computers is to enable new algorithms which render feasible problems requiring exorbitant resources for their solution on a classical computer.

There are two well-known types of algorithms that solve computational problems efficiently:

1. **Shor's quantum Fourier transform** - includes remarkable algorithms for solving the factoring and discrete logarithm problems, providing a striking exponential speedup over the best-known classical algorithms.
2. **Quantum searching** - based upon Grover's algorithm for performing quantum searching.

These provide a less striking but still remarkable quadratic speedup over the best possible classical algorithms. The quantum searching algorithm derives its importance from the widespread use of search-based techniques in classical algorithms, which in many instances allows a straightforward adaptation of the classical algorithm to give a faster quantum algorithm. The quantum searching algorithm has many potential applications. It can be used to extract statistics, such as the minimal element, from an unordered data set, more quickly than is possible on a classical computer. It can be used to speed up algorithms for some problems in NP – specifically, those problems for which a straightforward search for a solution is the best algorithm known. Finally, it can be used to speed up the search for keys to cryptosystems such as the widely used Data Encryption Standard (DES).

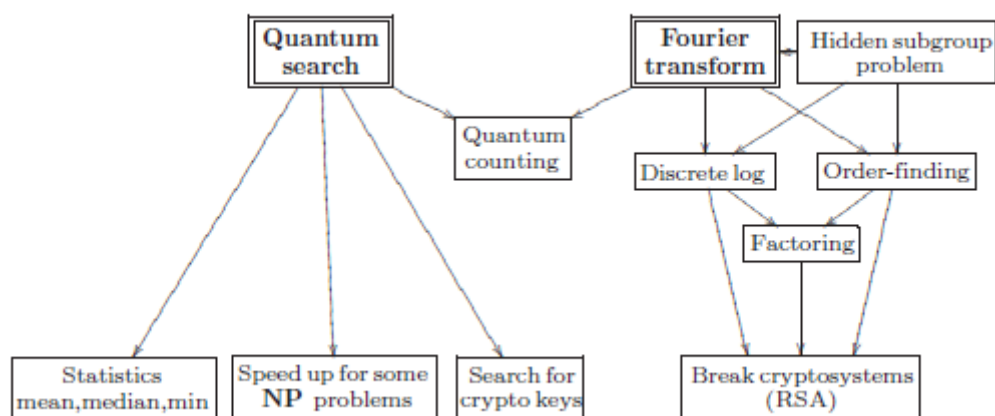


Figure 1. The main quantum algorithms and their relationships, including some notable applications.

Naturally, at the core of the diagram are the quantum Fourier transform and the quantum searching algorithm. Of particular interest in the figure is the quantum counting algorithm. This algorithm is a clever combination of the quantum searching and Fourier transform

algorithms, which can be used to estimate the number of solutions to a search problem more quickly than is possible on a classical computer.

The quantum Fourier transform has many interesting applications. It can be used to solve the discrete logarithm and factoring problems. These results, in turn, enable a quantum computer to break many of the most popular cryptosystems now in use, including the RSA cryptosystem. The Fourier transform also turns out to be closely related to an important problem in mathematics, finding a hidden subgroup. The quantum Fourier transform and several of its applications, including fast quantum algorithms for factoring and discrete logarithm.

Coming up with good quantum algorithms seems to be a difficult problem. There are at least two reasons for this:

- a) Algorithm design, be it classical or quantum, is not an easy business! The history of algorithms shows us that considerable ingenuity is often required to come up with near optimal algorithms, even for apparently very simple problems, like the multiplication of two numbers. Finding good quantum algorithms is made doubly difficult because of the additional constraint that we want our quantum algorithms to be better than the best-known classical algorithms.
- b) Our intuitions are much better adapted to the classical world than they are to the quantum world. If we think about problems using our native intuition, then the algorithms which we come up with are going to be classical algorithms. It takes special insights and special tricks to come up with good quantum algorithms.

### Single Qubit Operations

A single qubit is a vector  $|\psi\rangle = a|0\rangle + b|1\rangle$  parameterized by two complex numbers satisfying  $|a|^2 + |b|^2 = 1$ . Operations on a qubit must preserve this norm, and thus are described by  $2 \times 2$  unitary matrices. Of these, some of the most important are the Pauli matrices;

$$X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \quad Y \equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}; \quad Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Three other quantum gates will play a large part in what follows, the Hadamard gate, phase gate, and  $\pi/8$  gate (denoted T):

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}; \quad S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}; \quad T = \begin{bmatrix} 1 & 0 \\ 0 & \exp(i\pi/4) \end{bmatrix}$$

A couple of useful algebraic facts to keep in mind are that  $H = (X+Z)/\sqrt{2}$  and  $S = T^2$ . You might wonder why the T gate is called the  $\pi/8$  gate when it is  $\pi/4$  that appears in the definition. The reason is that the gate has historically often been referred to as the  $\pi/8$  gate, simply because up to an unimportant global phase T is equal to a gate which has  $\exp(\pm i\pi/8)$  appearing on its diagonals.

$$T = \exp(i\pi/8) \begin{bmatrix} \exp(-i\pi/8) & 0 \\ 0 & \exp(i\pi/8) \end{bmatrix}$$

A single qubit in the state  $a|0\rangle + b|1\rangle$  can be visualized as a point  $(\theta, \phi)$  on the unit sphere, where  $a = \cos(\theta/2)$ ,  $b = e^{i\phi} \sin(\theta/2)$ , and  $a$  can be taken to be real because the overall phase of the state is unobservable. This is called the Bloch sphere representation, and the vector  $(\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta)$  is called the Bloch vector. We shall return to this often as an aid to intuition. An arbitrary unitary operator on a single qubit can be written in many ways as a combination of rotations, together with global phase shifts on the qubit. The following theorem provides a means of expressing an arbitrary single qubit rotation that will be particularly useful in later applications to controlled operations.

(Z-Y decomposition for a single qubit) Suppose  $U$  is a unitary operation on a single qubit. Then there exist real numbers  $\alpha, \beta, \gamma$  and  $\delta$  such that:

$$U = e^{i\alpha} R_z(\beta) R_y(\gamma) R_z(\delta).$$

Since  $U$  is unitary, the rows and columns of  $U$  are orthonormal, from which it follows that there exist real numbers  $\alpha, \beta, \gamma$  and  $\delta$  such that:

$$U = \begin{bmatrix} e^{i(\alpha-\beta/2-\delta/2)} \cos \frac{\gamma}{2} & -e^{i(\alpha-\beta/2+\delta/2)} \sin \frac{\gamma}{2} \\ e^{i(\alpha+\beta/2-\delta/2)} \sin \frac{\gamma}{2} & e^{i(\alpha+\beta/2+\delta/2)} \cos \frac{\gamma}{2} \end{bmatrix}$$

The utility of lies in the mysterious looking of corollary, which is the key to the construction of controlled multi-qubit unitary operations.

Suppose  $U$  is a unitary gate on a single qubit. Then there exist unitary operators  $A, B, C$  on a single qubit such that  $ABC = I$  and  $U = e^{i\alpha} AXBXC$ , where  $\alpha$  is some overall phase factor.

In the notation of Theorem 4.1, set  $A \equiv R_z(\beta) R_y(\gamma/2)$ ,  $B \equiv R_y(-\gamma/2) R_z(-(\delta + \beta)/2)$  and  $C \equiv R_z((\delta - \beta)/2)$ . Note that

$$ABC = R_z(\beta) R_y\left(\frac{\gamma}{2}\right) R_y\left(-\frac{\gamma}{2}\right) R_z\left(-\frac{\delta + \beta}{2}\right) R_z\left(\frac{\delta - \beta}{2}\right) = I$$

Since  $X^2 = I$ , we see that

$$XBX = XR_y\left(-\frac{\gamma}{2}\right) XX R_z\left(-\frac{\delta + \beta}{2}\right) X = R_y\left(\frac{\gamma}{2}\right) R_z\left(\frac{\delta + \beta}{2}\right)$$

Thus

$$\begin{aligned} AXBXC &= R_z(\beta) R_y\left(\frac{\gamma}{2}\right) R_y\left(\frac{\gamma}{2}\right) R_z\left(\frac{\delta + \beta}{2}\right) R_z\left(\frac{\delta - \beta}{2}\right) \\ &= R_z(\beta) R_y(\gamma) R_z(\delta). \end{aligned}$$

Hence  $U = e^{i\alpha} AXBXC$  and  $ABC = I$ , as required.

Symbols for the common single qubit gates are shown below. Time proceeds from left to right; wires represent qubits, and a '/' may be used to indicate a bundle of qubits.







Hadamard		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli-X		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Phase		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

Figure 2. Names, symbols, and unitary matrices for the common single qubit gates.

### Controlled Operations

If A is true, then do B. This type of **controlled operation** is one of the most useful in computing, both classical and quantum. Here, we explain how complex controlled operations may be implemented using quantum circuits built from elementary operations.

CNOT is a quantum gate with two input qubits, known as the **control qubit** and **target qubit**.

In terms of the computational basis, the action of the is given by  $|c\rangle|t\rangle \rightarrow |c\rangle|t \oplus c\rangle$ ; that is, if the control qubit is set to  $|1\rangle$  then the target qubit is flipped, otherwise the target qubit is left alone. Thus, in the computational basis  $|control, target\rangle$  the matrix representation of CNOT is

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

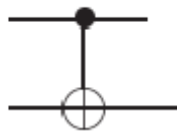


Figure 3. Circuit representation for the controlled- gate. The top line represents the control qubit, the bottom line the target qubit.

More generally, suppose  $U$  is an arbitrary single qubit unitary operation. A controlled- $U$  operation is a two-qubit operation, again with a control and a target qubit. If the control qubit is set then  $U$  is applied to the target qubit, otherwise the target qubit is left alone; that is,  $|c\rangle|t\rangle \rightarrow |c\rangle U^c|t\rangle$ . The controlled- $U$  operation is represented by the circuit shown below

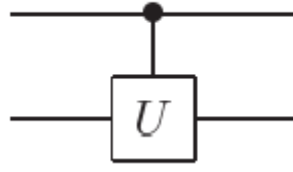


Figure 4. Controlled-U operation. The top line is the control qubit, and the bottom line is the target qubit. If the control qubit is set then  $U$  is applied to the target, otherwise it is left alone.

The goal is to understand how to implement the controlled- $U$  operation for arbitrary single qubit  $U$ , using only single qubit operations and the gate. Our strategy is a two-part procedure based upon the decomposition  $U = e^{i\alpha}AXBXC$  given in Corollary.

The first step will be to apply the phase shift  $\exp(i\alpha)$  on the target qubit, controlled by the control qubit. That is, if the control qubit is  $|0\rangle$ , then the target qubit is left alone, while if the control qubit is  $|1\rangle$ , a phase shift  $\exp(i\alpha)$  is applied to the target. A circuit implementing this operation using just a single qubit unitary gate is depicted on the right hand side. To verify that this circuit works correctly, note that the effect of the circuit on the right hand side is

$$|00\rangle \rightarrow |00\rangle, |01\rangle \rightarrow |01\rangle, |10\rangle \rightarrow e^{i\alpha}|10\rangle, |11\rangle \rightarrow e^{i\alpha}|11\rangle,$$

which is exactly what is required for the controlled operation on the left hand side

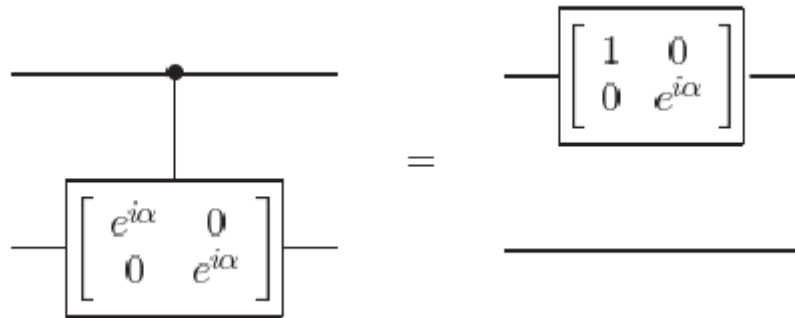


Figure 5. Controlled phase shift gate and an equivalent circuit for two qubits.

To understand why this circuit works, recall from Corollary that  $U$  may be written in the form  $U = e^{i\alpha}AXBXC$ , where  $A$ ,  $B$  and  $C$  are single qubit operations such that  $ABC = I$ . Suppose that the control qubit is set. Then the operation  $e^{i\alpha}AXBXC = U$  is applied to the second qubit. If, on the other hand, the control qubit is not set, then the operation  $ABC = I$  is applied to the second qubit; that is, no change is made. That is, this circuit implements the controlled- $U$  operation. Now that we know how to condition on a single qubit being set, what about conditioning on multiple qubits? We've already met one example of multiple qubit conditioning, the Toffoli gate, which flips the third qubit, the target qubit, conditioned on the first two qubits, the control qubits, being set to one. More generally, suppose we have  $n + k$  qubits, and  $U$  is a  $k$  qubit unitary operator. Then we define the controlled operation  $C^n(U)$  by the equation

$$C^n(U)|x_1x_2 \dots x_n\rangle|\psi\rangle = |x_1x_2 \dots x_n\rangle U^{x_1x_2 \dots x_n}|\psi\rangle$$

where  $x_1 x_2 \dots x_n$  in the exponent of  $U$  means the product of the bits  $x_1, x_2, \dots, x_n$ . That is, the operator  $U$  is applied to the last  $k$  qubits if the first  $n$  qubits are all equal to one, and otherwise, nothing is done. Such conditional operations are so useful that we

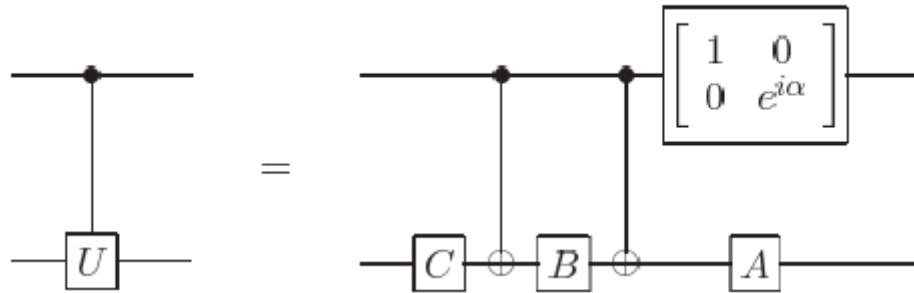


Figure 6. Circuit implementing the controlled- $U$  operation for single qubit  $U$ .  $\alpha, A, B$  and  $C$  satisfy  $U = \exp(i\alpha)AXBXC$ ,  $ABC = I$ .

introduce a special circuit notation below, we assume that  $k = 1$ , for simplicity. Larger  $k$  can be dealt with using essentially the same methods, however for  $k \geq 2$  there is the added complication that we don't (yet) know how to perform arbitrary operations on  $k$  qubits.

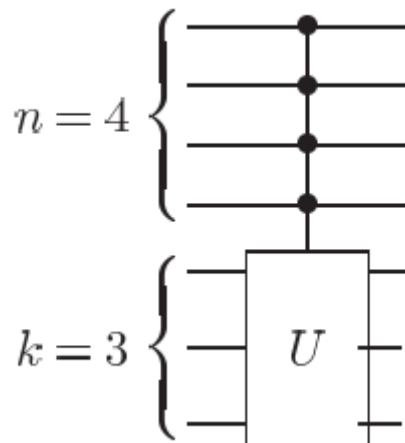


Figure 7. Sample circuit representation for the  $C^n(U)$  operation, where  $U$  is a unitary operator on  $k$  qubits, for  $n = 4$  and  $k = 3$ .

Suppose  $U$  is a single qubit unitary operator, and  $V$  is a unitary operator chosen so that  $V^2 = U$ . Then the operation  $C^2(U)$  may be implemented using the circuit shown below

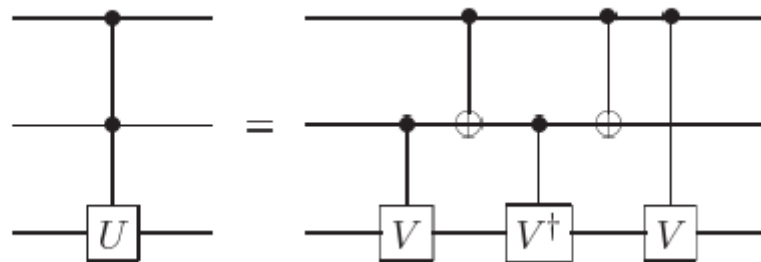


Figure 8. Circuit for the  $C^2(U)$  gate.  $V$  is any unitary operator satisfying  $V^2 = U$ . The special case  $V \equiv (I - i)(I + iX)/2$  corresponds to the Toffoli gate.

the case  $C^2(X)$ . Defining  $V \equiv (1 - i)(I + iX)/2$  and noting that  $V^2 = X$ , we see that it gives an implementation of the Toffoli gate in terms of one and two qubit operations. From a classical viewpoint this is a remarkable result. By contrast, in the quantum case we see that one and two qubit reversible gates are sufficient to implement the Toffoli gate, and will eventually prove that they suffice for universal computation. Ultimately, any unitary operation can be composed to an arbitrarily good approximation from just the Hadamard, phase, controlled-NOT and  $\pi/8$  gates. Because of the great usefulness of the Toffoli gate it is interesting to see how it can be built from just this gate set.

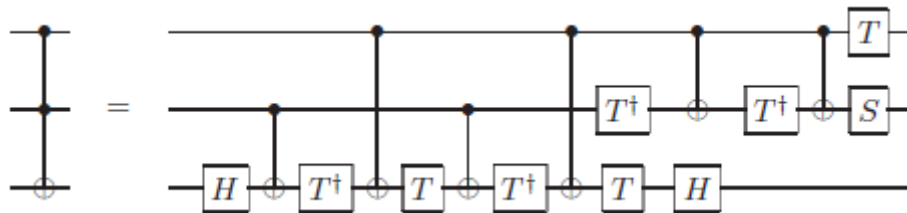


Figure 9. Implementation of the Toffoli gate using Hadamard, phase, controlled-NOT and  $\pi/8$  gates.

## Measurement

In our circuits, we always denote a projective measurement in the computational basis using a ‘meter’ symbol. Quantum circuits are always be represented by unitary transforms with ancilla qubits followed by projective measurements. There are some principles that we should follow. The first is classically conditioned operations can be replaced by quantum conditioned operations

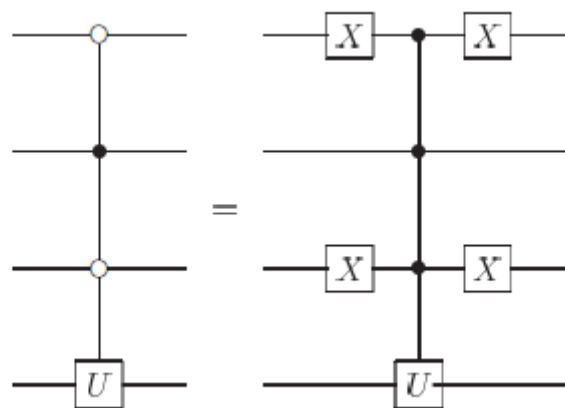


Figure 10. Controlled-U operation and its equivalent in terms of circuit elements we already know how to implement. The fourth qubit has  $U$  applied if the first and third qubits are set to zero, and the second qubit is set to one.

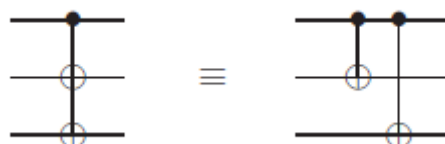


Figure 11. Controlled-NOT gate with multiple targets.

- a) **Principle of deferred measurement** - Measurements can always be moved from an intermediate stage of a quantum circuit to the end of the circuit; if the measurement

results are used at any stage of the circuit then the classically controlled operations can be replaced by conditional quantum operations.

Often, quantum measurements are performed as an intermediate step in a quantum circuit, and the measurement results are used to conditionally control subsequent quantum gates. This is the case, for example, in the teleportation circuit. However, such measurements can always be moved to the end of the circuit. It may also be done by replacing all the classical conditional operations by corresponding quantum conditional operations.

The second principle is even more obvious and surprisingly useful



Figure 12. Symbol for projective measurement on a single qubit. In this circuit nothing further is done with the measurement result, but in more general quantum circuits it is possible to change later parts of the quantum circuit, conditional on measurement outcomes in earlier parts of the circuit. Such a usage of classical information is depicted using wires drawn with double lines (not shown here).

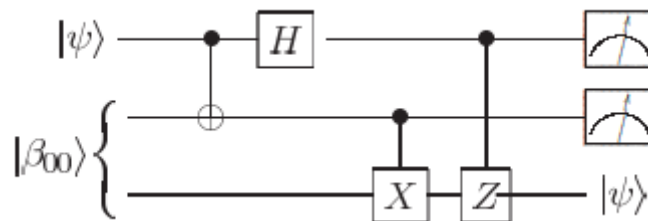


Figure 13. Quantum teleportation circuit in which measurements are done at the end, instead of in the middle of the circuit. The top two qubits belong to Alice, and the bottom one to Bob.

- b) Principle of implicit measurement** - Without loss of generality, any unterminated quantum wires (qubits which are not measured) at the end of a quantum circuit may be assumed to be measured.

To understand why this is true, imagine you have a quantum circuit containing just two qubits, and only the first qubit is measured at the end of the circuit. Then the measurement statistics observed at this time are completely determined by the reduced density matrix of the first qubit. However, if a measurement had also been performed on the second qubit, then it would be highly surprising if that measurement could change the statistics of measurement on the first qubit.

As you consider the role of measurements in quantum circuits, it is important to keep in mind that in its role as an interface between the quantum and classical worlds, measurement is generally considered to be an irreversible operation, destroying quantum information and replacing it with classical information. In certain carefully designed cases, however, this need not be true, as is vividly illustrated by teleportation and quantum error-correction. What teleportation and quantum error-correction have in common is that in neither instance does the measurement result reveal any information about the identity of the quantum state being measured. In order for a measurement to be reversible, it must reveal no information about the quantum state being measured.



## Universal Quantum Gates

A small set of gates (e.g. AND, OR, NOT ) can be used to compute an arbitrary classical function, we say that such a set of gates is universal for classical computation. A set of gates is said to be universal for quantum computation if any unitary operation may be approximated to arbitrary accuracy by a quantum circuit involving only those gates.

### **a) Two-level unitary gates are universal**

Consider a unitary matrix  $U$  which acts on a  $d$ -dimensional Hilbert space. Here, we explain how  $U$  may be decomposed into a product of two-level unitary matrices; that is, unitary matrices which act non-trivially only on two-or-fewer vector components. The essential idea behind this decomposition may be understood by considering the case when  $U$  is  $3 \times 3$ , so 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  $U_1, \dots, U_3$  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 inverses,  $U_1^\dagger, U_2^\dagger$  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

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

Note that in either case  $U_1$  is a two-level unitary matrix, and when we multiply the matrices out we get

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

The key point to note is that the middle entry in the left hand column is zero. We denote the other entries in the matrix with a generic prime ' $'$ '; their actual values do not matter. 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  $c' = 0$  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 of  $U_2 U_1 U$  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  $U = U^\dagger_1 U^\dagger_2 U^\dagger_3$ , 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  $3 \times 3$  case, we can find two-level unitary matrices  $U_1, \dots, U_{d-1}$  such that the matrix  $U_{d-1} U_{d-2} \dots U_1 U$  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  $U_{d-1} U_{d-2} \dots U_1 U$ , and so on, with the end result that an arbitrary  $d \times d$  unitary matrix may be written

$$U = V_1 \dots V_k$$

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

#### b) Single qubit and CNOT gates are universal

Single qubit and CNOT gates together can be used to implement an arbitrary two-level unitary operation on the state space of  $n$  qubits. Combining these results, we see that single qubit and CNOT gates can be used to implement an arbitrary unitary operation on  $n$  qubits, and therefore are universal for quantum computation.

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 the computational basis states  $|s\rangle$  and  $|t\rangle$ , where  $s = s_1 \dots s_n$  and  $t = t_1 \dots t_n$  are the binary expansions for  $s$  and  $t$ . Let  $\tilde{U}$  be the non-trivial  $2 \times 2$  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 members 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  $|g_1\rangle \rightarrow |g_2\rangle \rightarrow \dots \rightarrow |g_{m-1}\rangle$ , 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  $|g_{m-1}\rangle \rightarrow |g_{m-2}\rangle \rightarrow \dots \rightarrow |g_1\rangle$ . Each of these steps can be easily implemented using operations developed earlier in this chapter, and the final result is an implementation of  $U$ .

A more precise description of the implementation is as follows. The first step is to swap the states  $|g_1\rangle$  and  $|g_2\rangle$ . Suppose  $g_1$  and  $g_2$  differ at the  $i$ th digit. Then we accomplish the swap by performing a controlled bit flip on the  $i$ th qubit, conditional 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  $|g_{m-1}\rangle$ . The effect of this sequence of  $m-2$  operations is to achieve the operation

$$\begin{array}{l}
|g_1\rangle \rightarrow |g_{m-1}\rangle \\
|g_2\rangle \rightarrow |g_1\rangle \\
|g_3\rangle \rightarrow |g_2\rangle \\
\dots\dots\dots \\
|g_{m-1}\rangle \rightarrow |g_{m-2}\rangle.
\end{array}$$

All other computational basis states are left unchanged by this sequence of operations. Next, suppose  $g_{m-1}$  and  $g_m$  differ in the  $j$ th bit. We apply a controlled- $\tilde{U}$  operation with the  $j$ th 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: we swap  $|g_{m-1}\rangle$  with  $|g_{m-2}\rangle$ , then  $|g_{m-2}\rangle$  with  $|g_{m-3}\rangle$  and so on, until we swap  $|g_2\rangle$  with  $|g_1\rangle$ . A simple example illuminates the procedure further. Suppose we wish to implement the two-level unitary transformation

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

Here,  $a$ ,  $b$ ,  $c$  and  $d$  are any complex numbers such that

$$\mathcal{U} \equiv \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

is a unitary matrix.

Notice that  $U$  acts non-trivially only on the states  $|000\rangle$  and  $|111\rangle$ . 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. 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  $|000\rangle$ .

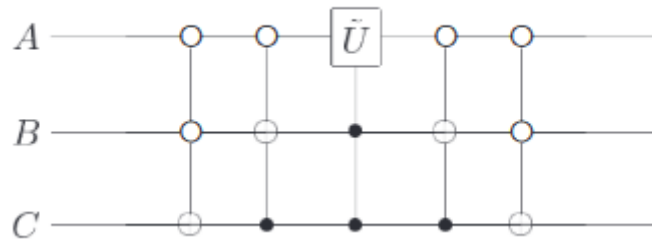


Figure 14. Circuit implementing the two-level unitary operation

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 these 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  $O(n^2)$  single qubit and CNOT gates. We saw in the previous section that an arbitrary unitary matrix on the  $2^n$ -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. We show that 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 than is taken in the universality construction.

## Simulation of Quantum Systems

### Simulation in action

The heart of simulation is the solution of differential equations which capture the physical laws governing the dynamical behavior of a system. Some examples include Newton's law.

Poisson's equation

$$\frac{d}{dt} \left( m \frac{dx}{dt} \right) = F,$$

$$-\vec{\nabla} \cdot (k \vec{\nabla} u) = \vec{Q},$$

the electromagnetic vector wave equation

$$\vec{\nabla} \cdot \vec{\nabla} \vec{E} = \epsilon_0 \mu_0 \frac{\partial^2 \vec{E}}{\partial t^2},$$

and the diffusion equation,

$$\vec{\nabla}^2 \psi = \frac{1}{a^2} \frac{\partial \psi}{\partial t},$$

The goal is generally: given an initial state of the system, what is the state at some other time and/or position? Solutions are usually obtained by approximating the state with a digital representation, then discretizing the differential equation in space and time such that an iterative application of a procedure carries the state from the initial to the final conditions. Importantly, the error in this procedure is bounded, and known not to grow faster than some small power of the number of iterations. Furthermore, not all dynamical systems can be simulated efficiently: generally, only those systems which can be described efficiently can be simulated efficiently. Simulation of quantum systems by classical computers is possible, but generally only very inefficiently. The dynamical behavior of many simple quantum systems is governed by Schrödinger's equation,

$$i\hbar \frac{d}{dt} |\psi\rangle = H |\psi\rangle.$$

We will find it convenient to absorb into H, and use this convention for the rest of this section. For a typical Hamiltonian of interest to physicists dealing with real particles in space, this reduces to

$$i \frac{\partial}{\partial t} \psi(x) = \left[ -\frac{1}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi(x),$$

using a convention known as the position representation  $\langle x | \psi \rangle = \psi(x)$ . This is an elliptical equation. So just simulating Schrödinger's equation is not the especial difficulty faced in simulating quantum systems. The key challenge in simulating quantum systems is the exponential number of differential equations which must be solved. For one qubit evolving according to the Schrödinger equation, a system of two differential equations must be solved; for two qubits, four equations; and for n qubits,  $2^n$  equations. Sometimes, insightful approximations can be made which reduce the effective number of equations involved, thus making classical simulation of the quantum system feasible. However, there are many physically interesting quantum systems for which no such approximations are known.

### **Further Reading**

## **References**

Michael A. Nielsen, I. L. (n.d.). *Quantum Computation and Quantum Information*. Cambridge.

