C.3 Quantum circuits

1. **Quantum circuit**: A *quantum circuit* is a sequential series of quantum transformations on a quantum register.

2. The inputs are usually computational basis states (all $|0\rangle$ unless stated otherwise).

3. *Quantum circuit diagrams* are drawn with time going from left to right, with the quantum gates crossing one or more “wires” (qubits) as appropriate.

4. **Unique features**: Acyclic: loops (feedback) are not allowed.

5. **Fan-In** (equivalent to OR) is not allowed, since it is not reversible or unitary.

6. **Fan-Out** is not allowed, because it would violate the No-cloning Theorem.
   (N.B.: This does not contradict the universality of the Toffoli or Fredkin gates, which are universal only with respect to classical states.)

7. **CNOT**: Fig. III.9 (right) shows the symbol for CNOT and its effect.

8. **Swap**: The swap operation is defined $|xy\rangle \mapsto |yx\rangle$, or explicitly

\[
\sum_{x,y \in \mathbb{Z}_2} |yx\rangle|x\rangle.
\]

9. We can put three CNOTs in series to swap two qubits (Exer. III.21). It has a special symbol as shown in Fig. III.11.
10. **Controlled-U**: In general, any unitary operator (on any number of qubits) can be controlled (see Fig. III.12). If the control bit is 0, it does nothing, otherwise it does $U$.

11. This is implemented by $|0\rangle|0\rangle \otimes I + |1\rangle|1\rangle \otimes U$.
   Effectively, the operators are entangled.

12. **Example**: Suppose the control bit is in superposition, $|\psi\rangle = a|0\rangle + b|1\rangle$.

\[
(|0\rangle|0\rangle \otimes I + |1\rangle|1\rangle \otimes U)|\psi\rangle
= (|0\rangle|0\rangle \otimes I + |1\rangle|1\rangle \otimes U)(a|0\rangle + b|1\rangle) \otimes |\psi\rangle
= |0\rangle(a|0\rangle + b|1\rangle) \otimes I|\psi\rangle + |1\rangle(a|0\rangle + b|1\rangle) \otimes U|\psi\rangle
= a|0\rangle \otimes |\psi\rangle + b|1\rangle \otimes U|\psi\rangle
= a|0, \psi\rangle + b|1, U\psi\rangle.
\]

We have a superposition of entangled outputs.

13. Recall that CNOT = controlled $X$.

14. **Conditional or controlled transformation**: If $U_0$ and $U_1$ are unitary operators, then we can make the choice between them conditional on a control bit as follows:

\[
|0\rangle|0\rangle \otimes U_0 + |1\rangle|1\rangle \otimes U_1.
\]

15. For example,

\[
\text{CNOT} = |0\rangle|0\rangle \otimes I + |1\rangle|1\rangle \otimes X. \quad \text{(III.17)}
\]
The state input into the circuit is understood via the equations

\[ \begin{align} 
|00\rangle & \rightarrow (|00\rangle + |11\rangle)/\sqrt{2} \equiv |\beta_{00}\rangle \\
|01\rangle & \rightarrow (|01\rangle + |10\rangle)/\sqrt{2} \equiv |\beta_{01}\rangle \\
|10\rangle & \rightarrow (|00\rangle - |11\rangle)/\sqrt{2} \equiv |\beta_{10}\rangle \\
|11\rangle & \rightarrow (|01\rangle - |10\rangle)/\sqrt{2} \equiv |\beta_{11}\rangle
\end{align} \]

Figure III.13: Quantum circuit for generating Bell states. [from NC fig. 1.12]

\[ |\psi\rangle \xrightarrow{M} \]

Figure III.14: Symbol for measurement of a quantum state (from NC).

\[ \begin{align} 
\text{¶16. Other special gates: } & \text{ The symbol for the CCNOT gate is show in Fig. III.10,} \\
& \text{ or with } \bullet \text{ for top two connections and } \oplus \text{ for bottom, representing } \\
& \text{CCNOT} \langle x, y, z \rangle = \langle x, y, xy \oplus z \rangle, \\
& \text{or put “CCNot” in a box.}
\end{align} \]

\[ \begin{align} 
\text{¶17. Other operations may be shown by putting a letter or symbol in a box,} \\
& \text{ for example “H” for the Hadamard gate.}
\end{align} \]

\[ \begin{align} 
\text{¶18. } H \text{ can be used to generate Bell states (Exer. III.20):} \\
\text{CNOT}(H \otimes I)|xy\rangle = |\beta_{xy}\rangle. \tag{III.18}
\end{align} \]

\[ \begin{align} 
\text{¶19. The circuit for generating Bell states (Eq. III.18) is shown in Fig. III.13.}
\end{align} \]

\[ \begin{align} 
\text{¶20. Measurement: } & \text{ It’s also convenient to have a symbol for quantum} \\
& \text{state measurement, such as Fig. III.14.}
\end{align} \]
C. QUANTUM INFORMATION

\[ |c\rangle \quad |x\rangle \quad |y\rangle \quad |0\rangle \quad |0\rangle \]

Figure III.15: Quantum circuit for 1-bit full adder [from IQC]. “\(x\) and \(y\) are the data bits, \(s\) is their sum (modulo 2), \(c\) is the incoming carry bit, and \(c'\) is the new carry bit.”

\[ \Phi \quad \text{CNOT} \quad \Phi^{-1} \]

\(U_f\)

Figure III.16: Quantum gate array for reversible quantum computation.

C.4 Quantum gate arrays

1. **Full adder:** Fig. III.15 shows a quantum circuit for a 1-bit full adder.

2. As we will see (Sec. C.5), it is possible to construct reversible quantum gates for any classically computable function. In particular the Fredkin and Toffoli gates are universal.

3. **Reversibility:** Because quantum computation is a unitary operator, it must be reversible.
   
   You know that an irreversible computation \(x \mapsto f(x)\) can be embedded in a reversible computation \((x, c) \mapsto (f(x), g(x))\), where \(c\) are suitable constants and \(g(x)\) represents the garbage bits.
4. Note that throwing away the garbage bits (dumping them in the environment) will collapse the state (equivalent to measurement) by entangling them in the many degrees of freedom of the environment.

5. Since NOT is reversible, each 1 bit in \( c \) can be replaced by a 0 bit followed by a NOT, so we need only consider \((x, 0) \mapsto (f(x), g(x))\).

6. The garbage must be produced in a standard state independent of \( x \), “because garbage bits whose value depends upon \( x \) will in general destroy the interference properties crucial to quantum computation.”

7. **Uncomputation:** This is accomplished by uncomputing. Specifically, perform the computation on four registers \((data, result, workspace, target)\):

\[
(x, 0, 0, y) \mapsto (x, f(x), g(x), y).
\]

Notice that \( x \) and \( y \) (data and target) are passed through.

8. Now use CNOTs to compute \( y \oplus f(x) \), where \( \oplus \) represents bitwise exclusive-or, in the fourth register:

\[
(x, 0, 0, y) \mapsto (x, f(x), g(x), y \oplus f(x)).
\]

9. Now we uncompute \( f \), but since the data and target registers is passed through, we get \((x, 0, 0, y \oplus f(x))\).

Ignoring the result and workspace registers, we write

\[
(x, y) \mapsto (x, y \oplus f(x)).
\]

10. **Quantum gate array:** Therefore, for any computable \( f : 2^m \to 2^n \), there is a reversible quantum gate array \( U_f : \mathcal{H}^{m+n} \to \mathcal{H}^{m+n} \) such that for \( x \in 2^m \) and \( y \in 2^n \),

\[
U_f |x, y\rangle = |x, y \oplus f(x)\rangle,
\]

See Fig. III.17.

11. The first \( m \) qubits are called the data register and the last \( n \) are called the target register.

12. In particular, \( U_f |x, 0\rangle = |x, f(x)\rangle \).
C. QUANTUM INFORMATION

Introduction to Quantum Computing

Deutsch has shown [Deutsch 1985] that it is possible to construct reversible quantum gates for any classically computable function. In fact, it is possible to conceive of a universal quantum Turing machine [Bernstein and Vazirani 1997]. In this construction we must assume a sufficient supply of bits that correspond to the tape of a Turing machine.

Knowing that an arbitrary classical function $f$ with $m$ input and $k$ output bits can be implemented on a quantum computer, we assume the existence of a quantum gate array $U_f$ that implements $f$. $U_f$ is a $m + k$ bit transformation of the form

$$U_f : |x, y_i \mapsto |x, y_i f(x)$$

where $\oplus$ denotes the bitwise exclusive-OR.

Quantum arrays $U_f$, defined in this way, are unitary for any function $f$. To compute $f(x)$ we apply $U_f$ to $|x\rangle$ tensored with $|0\rangle^k$. Since $f(x) = 0$ we have

$$U_f U_f = I.$$  

Graphically the transformation $U_f : |x, y_i \mapsto |x, y_i f(x)$ is depicted as

$\begin{array}{c}
|x\rangle \quad \rightarrow \quad |x\rangle \\
|y\rangle \quad \rightarrow \quad |y \oplus f(x)\rangle
\end{array}$

While the $T$ and $F$ gates are complete for combinatorial circuits, they cannot achieve arbitrary quantum state transformations. In order to realize arbitrary unitary transformations, single bit rotations need to be included. Barenco et. al. [Barenco et al. 1995] show that $C$ not together with all 1-bit quantum gates is a universal gate set. It suffices to include the following one-bit transformations

$\begin{array}{c}
\cos \theta & \sin \theta \\
\sin \theta & \cos \theta
\end{array}$

for all $0 \leq \theta \leq \frac{\pi}{2}$ together with the $C$ not to obtain a universal set of gates. As we shall see, such non-classical transformations are crucial for exploiting the power of quantum computers.

5.2 Quantum Parallelism

What happens if $U_f$ is applied to input which is in a superposition? The answer is easy but powerful: since $U_f$ is a linear transformation, it is applied to all basis vector simultaneously and will generate a superposition of the results. In this way, it is possible to compute $f(x)$ for $n$ values of $x$ in a single application of $U_f$. This effect is called quantum parallelism.

The power of quantum algorithms comes from taking advantage of quantum parallelism and entanglement. So most quantum algorithms begin by computing a function of interest on a superposition of all values as follows. Start with an $n$-qubit state $|00\ldots0\rangle$. Apply the

$\begin{array}{c}
|x\rangle \quad \rightarrow \quad |x\rangle \\
|y\rangle \quad \rightarrow \quad |y \oplus f(x)\rangle
\end{array}$

Figure III.17: Computation of function by quantum gate array [from IQC].