Multi-qubit gates

With only single-qubit gates, we could never produce entangled states like \( |\psi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \)

since \( (U \otimes V) |\psi\rangle \otimes |\psi\rangle = U |\psi\rangle \otimes V |\psi\rangle \)

Multi-qubit gates are needed to generate entanglement and access the full power of QC

Recall:

A state of \( n \) bits is a vector in \( \mathbb{C}^n \)

An \( n \)-qubit gate is thus a unitary \( U : \mathbb{C}^n \to \mathbb{C}^n \)

Ex. Important multi-qubit unitaries include

\[
\begin{align*}
\text{CNOT} &= \begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
0
\end{array}
\end{array}
\end{array}
\end{align*}
\]

Intuitive View

\( (x,y,z \in \{0,1\}) \)

\[
\begin{align*}
\text{CNOT} |x\rangle |y\rangle &= |x\rangle |x\oplus y\rangle \\
\text{CZ} |x\rangle |y\rangle &= (-1)^y |x\rangle |y\rangle \\
\text{SWAP} |x\rangle |y\rangle &= |y\rangle |x\rangle \\
\text{Toffoli} |x\rangle |y\rangle |z\rangle &= |x\rangle |y\rangle |z\oplus xy\rangle
\end{align*}
\]
Controlled gates

CNOT, CZ, Toffoli are examples of controlled gates. Controlled gates apply a unitary $U$ on the target(s) if and only if the computational basis state of the control is $|1\rangle$

Ex.

Given an $n$-qubit unitary $U$, the controlled-$U$ gate is an $n+1$-qubit gate that sends

$|0\rangle |\psi\rangle \mapsto |0\rangle |\psi\rangle$

$|1\rangle |\psi\rangle \mapsto |1\rangle (U |\psi\rangle)$

$(\alpha|0\rangle + \beta|1\rangle) |\psi\rangle \mapsto \alpha |0\rangle U |\psi\rangle + \beta |1\rangle (U |\psi\rangle)$

The controlled-$U$ gate is written

The CNOT gate is a controlled-$X$ gate

CZ gate is a controlled-$Z$ gate

Ex. The CNOT gate is entangling

$\text{Let } |\psi\rangle = |1\rangle |0\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |10\rangle)$

Then $\text{CNOT } |\psi\rangle = \frac{1}{\sqrt{2}} (\text{CNOT } |00\rangle + \text{CNOT } |10\rangle)

= \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)

= |\Phi\rangle$
Matrix of a controlled gate

Given a unitary \( U \), the controlled-\( U \) gate can be written as

\[
CU = 10 \otimes U + 11 \otimes U
\]

Ex. \( \text{CNOT} \)

\[
C_X = 10 \otimes 1 + 11 \otimes X
\]

\[
\begin{bmatrix}
1 & 0 \\
0 & 0
\end{bmatrix} \otimes 1 + \begin{bmatrix}
0 & 0 \\
0 & 1
\end{bmatrix} \otimes X
\]

\[
\begin{bmatrix}
1 & 0 \\
0 & 0
\end{bmatrix} + \begin{bmatrix}
0 & 0 \\
0 & X
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

(Negative Controls)

We sometimes want to apply a gate if and only if the control is in the \( 0 \) state. We call this a negative control and denote it by

\[
\text{\textcolor{red}{\textbf{white dot}}}
\]

Ex.

Negating the control on a CNOT flips the target bit if and only if the control is \( 0 \)

\[
\begin{array}{c}
a \\
\text{\textcolor{red}{\textbf{white dot}}} \\
b = b \oplus (1 \otimes a)
\end{array}
\]

Fact

\[
\begin{array}{c}
\text{\textcolor{red}{\textbf{white dot}}} \\
\text{\textcolor{red}{\textbf{white dot}}}
\end{array}
\]

\[
\begin{array}{c}
\text{\textcolor{red}{\textbf{white dot}}} \\
\text{\textcolor{red}{\textbf{white dot}}}
\end{array}
\]
Multiply-controlled gates

Observe that the Toffoli gate is a controlled-CNOT gate:

\[
\text{Toffoli: } 10 \uparrow 1 \uparrow 12 \downarrow \quad \rightarrow \quad 10 \uparrow 1 \uparrow 12 \uparrow \\
11 \uparrow 1 \uparrow 12 \downarrow \quad \rightarrow \quad 11 \uparrow 1 \uparrow 12 \uparrow \uparrow = 11 \uparrow (\text{CNOT} + 1) \uparrow 12 \downarrow
\]

Controlled-controlled gates like the Toffoli gate are called multi-controlled gates. We draw each control with a dot:

\[
\begin{array}{c}
\text{U} \\
\downarrow \\
\text{U is applied if and only if each control is 1}
\end{array}
\]

(Multiply-controlled Toffolis)

An important multiply-controlled gate is the multiply-controlled X gate:

\[
\begin{array}{c}
\text{y} \\
x_1 \\
\vdots \\
x_k
\end{array}
\quad \rightarrow \quad
\begin{array}{c}
\text{y} \oplus x_1, \ldots, x_k
\end{array}
\]

Such a gate is often called an MCT gate and computes the product of its \( k \) controls:

\[
| x_1, \ldots, x_k \rangle | y \rangle \quad \rightarrow \quad | y, \ldots, x_k \rangle | y \oplus x_1, \ldots, x_k \rangle
\]

It's confusing, I know.
Construction of controlled gates

The construction of (multi-)controlled gates is another fundamental problem in quantum compilation.

Barenco et al.

Elementary gates for quantum computation lays the foundation for this problem.

Our first goal is to show that multi-controlled gates may be implemented using only CNOT & single-qubit gates.

(Basic Construction of Controlled gates)

Observe that if $U = V^4 W V$, then

\[
\begin{array}{c}
\text{U} \\
\end{array}
\begin{array}{c}
\text{V} \\
\text{W} \\
\text{V}^4
\end{array}
\]

In particular, the rhs sends

\[
\begin{align*}
|0\rangle |\psi\rangle & \rightarrow |0\rangle (V + V |\psi\rangle) = |0\rangle |\psi\rangle \\
|1\rangle |\psi\rangle & \rightarrow |0\rangle (V W V |\psi\rangle) = |0\rangle (V |\psi\rangle)
\end{align*}
\]

Ex.

Recall that $X = HZH$. We can construct the controlled-$X$ (CNOT) gate as

\[
\begin{array}{c}
\text{U} \\
\end{array}
\begin{array}{c}
\text{H} \\
\end{array}
\begin{array}{c}
\text{H} \\
\end{array}
\]

Universal 1-control gate construction

**Prop.**

For any 1-qubit unitary \( U \), \( CU \) can be constructed from single-qubit and CNOT gates. Specifically,

\[
\begin{align*}
\begin{array}{c}
\text{U} \\
\text{C} \\
\text{B} \\
\text{A}
\end{array}
\end{align*}
\]

For some \( \alpha, A, B, C \) satisfying \( ABC = I \) and \( U = e^{i\alpha} AXBXC \)

**Pf.**

Observe that the above circuit sends

\[
\begin{align*}
|0\rangle \langle 0| & \rightarrow (R_z(\alpha) |0\rangle \otimes (ABC |1\rangle) = e^{i\alpha |0\rangle \langle 0|} |1\rangle \\
|1\rangle \langle 1| & \rightarrow (R_z(\alpha) |1\rangle \otimes (AXBXC |1\rangle) = e^{i\alpha |1\rangle \langle 1|} (U |1\rangle)
\end{align*}
\]

: equal up to a global phase of \( e^{i\alpha} \)

Now let \( U = e^{i\alpha} R_z(\alpha) R_y(\frac{\pi}{4}) R_z(\beta) \). Then

\[
A = R_z(\alpha) R_y(\frac{\pi}{4}), \quad B = R_y(-\frac{\pi}{2}) R_z(-\frac{\pi}{4}), \quad C = R_z(\frac{\pi}{2})
\]

satisfies the above constraints

**Ex.**

\[
T^2 = S
\]

Note that \( S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} = e^{i \frac{\pi}{4}} \begin{bmatrix} 1 & -i \\ i & 1 \end{bmatrix} = e^{i \frac{\pi}{4}} R_z(0) R_y(0) R_z(\frac{\pi}{4}) \)

Setting \( A = R_z(0) R_y(0) \)

\( B = R_y(0) R_z(-\frac{\pi}{4}) \)

\( C = R_z(\frac{\pi}{4}) \)

we get

\[
\begin{align*}
\begin{array}{c}
\text{S} \\
\end{array}
\end{align*}
\]

up to a global phase \( e^{i\pi/4} \)

Correct global phase
Construction of 2-control gates

Prop. For any 1-qubit unitary $U$, CCU can be constructed as

\[
\begin{array}{c}
\text{over } S_{U,2} \text{ CNOT}
\end{array}
\]

\[
\begin{array}{cccc}
\text{We have 4 cases: } & 1 & 00 & \square \\text{I} \\
2 & 01 & \text{I} \\
3 & 10 & \text{I} \\
4 & 11 & (\sqrt{U})^9 = U
\end{array}
\]

Ex. Recall that the Toffoli gate is an X gate with two controls. In classical computing, the Toffoli gate cannot be constructed (reversibly) by a circuit of < 3 bit gates. Quantumly, by the above

\[
\begin{array}{c}
\text{Note also that } \sqrt{X} = S \text{ and } \sqrt{S} = T, \text{ so}
\end{array}
\]

Thm. For the Toffoli gate:
- 5 two-qubit gates is minimal
- 7 T-gates is minimal over \{CNOT, H, T\}
Construction of multi-controlled gates

Multiply-controlled gates are used in most top-down universal constructions (and often in algorithms). We now look at general techniques for decomposing into fewer gates.

**Decomposition with 1 ancilla**

\[
\begin{array}{c}
\vdots \\
\vdots \\
U \\
\end{array}
\quad = 
\begin{array}{c}
\vdots \\
\vdots \\
10> \\
\end{array}
\quad 10>
\begin{array}{c}
\vdots \\
\vdots \\
U \\
\end{array}
\]

Note that \( C^h U \) “fires” if and only if all \( h \)-controls are 1.

\[
x_1 = 1 \land x_2 = 1 \land \ldots \land x_h = 1 \iff x_1 x_2 \ldots x_h = 1
\]

**Decomposition with 1 ancilla in an unknown state**

If \( U^2 = I \), then

\[
\begin{array}{c}
\vdots \\
\vdots \\
U \\
\end{array}
\quad = 
\begin{array}{c}
\vdots \\
10> \\
\end{array}
\quad 10>
\begin{array}{c}
\vdots \\
\vdots \\
U \\
\end{array}
\]

The ancilla here is called dirty and can in fact be a qubit used elsewhere in the circuit!

**E.g.**

\[
\begin{array}{c}
\vdots \\
\vdots \\
\bigvee \\
\end{array}
\quad \downarrow 
\begin{array}{c}
\vdots \\
\vdots \\
W \quad V \\
\end{array}
\]

\[
\begin{array}{c}
\vdots \\
\vdots \\
V \\
\end{array}
\quad \downarrow 
\begin{array}{c}
\vdots \\
\vdots \\
W \quad W \\
\end{array}
\]

Multi-controlled Toffoli gates

The previous constructions didn't reduce the number of controls, just shifted them to multi-controlled Toffoli (MCT) gates. To get things down to single-qubit and CNOT we need a construction which reduces controls

\[ \text{(MCT with linear clean ancillas)} \]

\[ \text{one fewer control} \]

Prop.

A \( k \)-Control Toffoli can be implemented with

- \( k-2 \) clean ancillas
- \( 2(k-2)+1 \) Toffoli gates

\[ \text{(MCT with linear dirty ancillas)} \]

A similar result holds for dirty ancillas, but its construction is slightly trickier. First observe:

\[ \text{dirty ancilla returned to its initial state} \]

This is because at ① we have \( y \oplus x_k \), so at ②

\[ y \oplus x_k \oplus (a \oplus x_1 \ldots x_{k-1}) x_k = y \oplus x_1 \ldots x_k \]

We can recursively apply this construction, giving \( k-2 \) dirty ancillas and \( 2(k-2)+1 \) Toffoli gates, but this will leave the \( k-2 \) dirty ancillas in an unclean state.
To return the ancillas to their initial state, we need to clean them by reversing the intermediate $k$-1 control Toffoli computation. The circuit for $5$ controls is

```
\[
\begin{array}{cccccccc}
    & x_1 & x_2 & x_3 & x_4 & x_5 & a_1 & a_2 & a_3 & a_4 & a_5 & y \\
1 &  &  &  &  &  &  &  &  &  &  &  \\
2 &  &  &  &  &  &  &  &  &  &  &  \\
3 &  &  &  &  &  &  &  &  &  &  &  \\
4 &  &  &  &  &  &  &  &  &  &  &  \\
5 &  &  &  &  &  &  &  &  &  &  &  \\
6 &  &  &  &  &  &  &  &  &  &  &  \\
```

- compute product
- uncompute changes to ancillas

Prop. A $k$-control MCT gate can be implemented with
- $k-2$ dirty ancillas
- $2(k-2)+1 + 2(k-3)+1$ Toffoli gates

(MCT gate with a single ancilla)

Finally we show that an MCT gate may be implemented with just 1 ancilla (dirty or clean)

```
\[
\begin{array}{cccccccc}
    &  &  &  &  &  &  &  &  &  &  \\
1 &  &  &  &  &  &  &  &  &  &  \\
2 &  &  &  &  &  &  &  &  &  &  \\
3 &  &  &  &  &  &  &  &  &  &  \\
4 &  &  &  &  &  &  &  &  &  &  \\
5 &  &  &  &  &  &  &  &  &  &  \\
6 &  &  &  &  &  &  &  &  &  &  \\
\end{array}
```

This decomposes a $k$-control MCT into $k/2$ and $k/2+1$ control MCT gates. We can then use the other $k/2-1$ bits as dirty ancillas to apply the previous linear simulation!

Prop. A $k$-control MCT gate can be implemented with
- 1 ancilla (dirty or clean)
- $O(n)$ Toffoli gates
A note about efficiency

In quantum computing, the constants matter. Physically realizable circuit depths double in years, and error-resistance means saving \( \frac{1}{2} \) on circuit depth can reduce time and space resources by orders of magnitude.

The Barenco et al. single-ancilla construction was just recently beaten by Amy & Ross 😄

With a single ancilla, the number of Toffoli gates to implement an MCT gate with the Barenco method scales roughly as

- \( 4(4k^2) = 8k \) (dirty)
- \( 3(4k^2) = 6k \) (clean)

Prep.

If \( x,z \) basis changes (conjugation by Hadamard gates) are allowed, an MCT gate can be implemented with 1 ancilla and \( \sim 4k \) CCX gates

\[ \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \quad \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \quad \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \]

\[ \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \quad \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \quad \begin{array}{c}
\text{1} \\
\vdots \\
\text{2} \\
\text{3} \\
\end{array} \]