Today's outline - February 03, 2022

Today's outline - February 03, 2022

- Multiply controlled operators

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators
- Implementing general operators

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators
- Implementing general operators
- Universally approximating gates

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators
- Implementing general operators
- Universally approximating gates

Reading Assignment: Chapter 7.1-7.2

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators
- Implementing general operators
- Universally approximating gates

Reading Assignment: Chapter 7.1-7.2

Homework Assignment #04:
Chapter 5:4,6,9,15,16,17
due Tuesday, February 15, 2022

Today's outline - February 03, 2022

- Multiply controlled operators
- Arbitrary controlled operators
- Implementing general operators
- Universally approximating gates

Reading Assignment: Chapter 7.1-7.2

Homework Assignment #04:
Chapter 5:4,6,9,15,16,17
due Tuesday, February 15, 2022

No class on Tuesday, Feb 08, 2022

Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

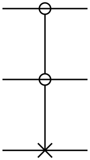
The CC_{not}, also called the Toffoli gate, $\bigwedge_2 X$ negates the last bit if the first two are 1

Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

The CC_{not}, also called the Toffoli gate, $\bigwedge_2 X$ negates the last bit if the first two are 1



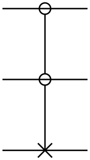
Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

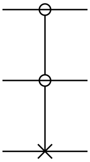
$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

The CC_{not}, also called the Toffoli gate, $\bigwedge_2 X$ negates the last bit if the first two are 1

The arbitrary Q transformation can also be controlled by multiple qubits



Multiply controlled transformations



Controlled operations can be generalized to more than one control bit

$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

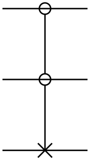
The CC_{not}, also called the Toffoli gate, $\bigwedge_2 X$ negates the last bit if the first two are 1

The arbitrary Q transformation can also be controlled by multiple qubits

The $\bigwedge_2 Q$ three-qubit gate can be obtained by adding control of the Q_0, Q_1, and Q_2 by the third qubit

Multiply controlled transformations

Controlled operations can be generalized to more than one control bit

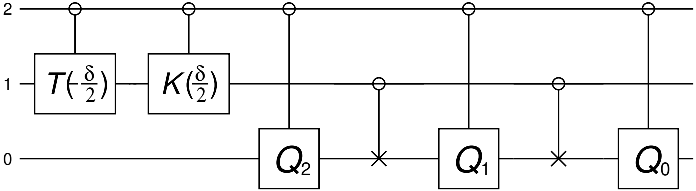


$\bigwedge_k Q$ represents a $(k+1)$-qubit transformation that applied Q to the low order qubit if all of the other qubits are 1

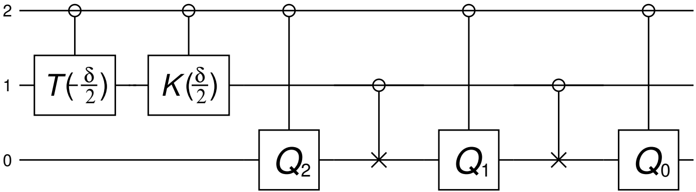
The CC_{not}, also called the Toffoli gate, $\bigwedge_2 X$ negates the last bit if the first two are 1

The arbitrary Q transformation can also be controlled by multiple qubits

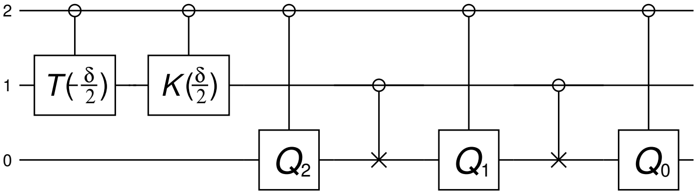
The $\bigwedge_2 Q$ three-qubit gate can be obtained by adding control of the Q_0, Q_1, and Q_2 by the third qubit



Multiply controlled transformations (cont.)

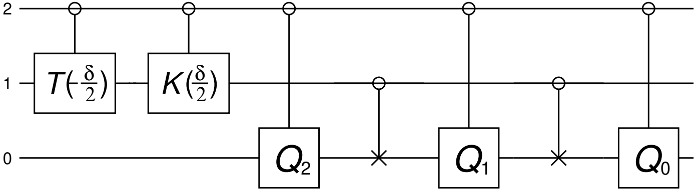


Multiply controlled transformations (cont.)



This circuit can be expanded in terms of the general phase shift and rotation gates plus C_{not}, however it requires 25 single qubit gates and 12 C_{not} gates

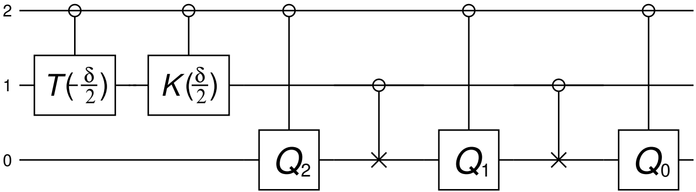
Multiply controlled transformations (cont.)



This circuit can be expanded in terms of the general phase shift and rotation gates plus C_{not}, however it requires 25 single qubit gates and 12 C_{not} gates

For a general k-qubit controlled arbitrary gate, one needs 5^k single qubit gates plus $\frac{1}{2}(5^k - 1)$ C_{not} gates which is not the most efficient implementation

Multiply controlled transformations (cont.)

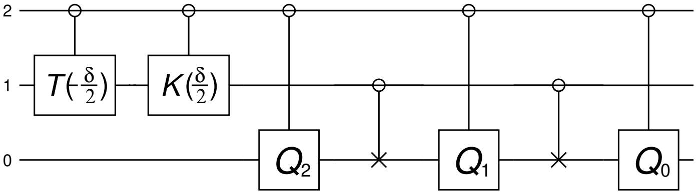


This circuit can be expanded in terms of the general phase shift and rotation gates plus C_{not}, however it requires 25 single qubit gates and 12 C_{not} gates

For a general k-qubit controlled arbitrary gate, one needs 5^k single qubit gates plus $\frac{1}{2}(5^k - 1)$ C_{not} gates which is not the most efficient implementation

Suppose we want to apply a transformation when the control qubit is 0 or a specific combination of 1's and 0's

Multiply controlled transformations (cont.)

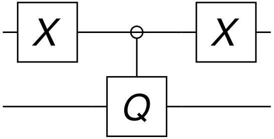


This circuit can be expanded in terms of the general phase shift and rotation gates plus C_{not}, however it requires 25 single qubit gates and 12 C_{not} gates

For a general k-qubit controlled arbitrary gate, one needs 5^k single qubit gates plus $\frac{1}{2}(5^k - 1)$ C_{not} gates which is not the most efficient implementation

Suppose we want to apply a transformation when the control qubit is 0 or a specific combination of 1's and 0's

This is possible by adding two X gates to the control bit



Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k + 1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\left\{ \left| s_k \dots s_{i+1} x_i s_{i-1} \dots s_0 \right\rangle, \left| s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0 \right\rangle \right\},$$

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\left\{ |s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle \right\}, \ \hat{x}_i = x_i \oplus 1 \ (\mathrm{XOR})$$

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\left\{ |s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle \right\}, \ \hat{x}_i = x_i \oplus 1 \ (\text{XOR})$$

It will be useful to define two different transformations using a k-qubit string and a single qubit transformation, Q, on a separate qubit, both of which can be represented as $\bigwedge_x^i Q$

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\{|s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle\}, \ \hat{x}_i = x_i \oplus 1 \ (\text{XOR}) \ \longrightarrow \ \{|x\rangle, |\hat{x}\rangle\} \ \hat{x} = x \oplus 2^i$$

It will be useful to define two different transformations using a k-qubit string and a single qubit transformation, Q, on a separate qubit, both of which can be represented as $\bigwedge_x^i Q$

x is a $(k+1)$-qubit string where the i^{th} qubit $|x_i\rangle$ is either $|0\rangle$ or $|1\rangle$ and the other qubits are defined as $s_k \dots s_{i+1} s_{i-1} \dots s_0$

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\{|s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle\}, \; \hat{x}_i = x_i \oplus 1 \text{ (XOR)} \longrightarrow \{|x\rangle, |\hat{x}\rangle\} \; \hat{x} = x \oplus 2^i$$

It will be useful to define two different transformations using a k-qubit string and a single qubit transformation, Q, on a separate qubit, both of which can be represented as $\bigwedge_x^i Q$

x is a $(k+1)$-qubit string where the i^{th} qubit $|x_i\rangle$ is either $|0\rangle$ or $|1\rangle$ and the other qubits are defined as $s_k \dots s_{i+1} s_{i-1} \dots s_0$

If $|x_i\rangle = |0\rangle$, $Q|x_i\rangle$ is applied

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\left\{|s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle\right\}, \ \hat{x}_i = x_i \oplus 1 \ (\text{XOR}) \ \longrightarrow \ \left\{|x\rangle, |\hat{x}\rangle\right\} \ \hat{x} = x \oplus 2^i$$

It will be useful to define two different transformations using a k-qubit string and a single qubit transformation, Q, on a separate qubit, both of which can be represented as $\bigwedge_x^i Q$

x is a $(k+1)$-qubit string where the i^{th} qubit $|x_i\rangle$ is either $|0\rangle$ or $|1\rangle$ and the other qubits are defined as $s_k \dots s_{i+1} s_{i-1} \dots s_0$

If $|x_i\rangle = |0\rangle$, $Q|x_i\rangle$ is applied but if $|x_i\rangle = |1\rangle$, $XQX|x_i\rangle$ is applied

Arbitrary controlled transformations

These multiply controlled qubit gates will permit arbitrary circuits

Suppose we have a $(k+1)$-qubit system to which we wish to apply transformation Q on the i^{th} qubit when all the other qubits are in a specific basis state

The transformation Q is thus applied to a 2-dimensional subspace spanned by the vector with x_i and its flipped state in the standard basis, $\hat{x}_i$

$$\left\{|s_k \dots s_{i+1} x_i s_{i-1} \dots s_0\rangle, |s_k \dots s_{i+1} \hat{x}_i s_{i-1} \dots s_0\rangle\right\}, \; \hat{x}_i = x_i \oplus 1 \text{ (XOR)} \;\longrightarrow\; \left\{|x\rangle, |\hat{x}\rangle\right\} \; \hat{x} = x \oplus 2^i$$

It will be useful to define two different transformations using a k-qubit string and a single qubit transformation, Q, on a separate qubit, both of which can be represented as $\bigwedge_x^i Q$

x is a $(k+1)$-qubit string where the i^{th} qubit $|x_i\rangle$ is either $|0\rangle$ or $|1\rangle$ and the other qubits are defined as $s_k \dots s_{i+1} s_{i-1} \dots s_0$

If $|x_i\rangle = |0\rangle$, $Q|x_i\rangle$ is applied but if $|x_i\rangle = |1\rangle$, $XQX|x_i\rangle$ is applied

This operator has the property that: $\bigwedge_{\hat{x}}^i Q = \bigwedge_x^i \hat{Q} = \bigwedge_x^i XQX$

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$				

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
$\bigwedge_{10}^0 X$				

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
$\bigwedge_{10}^0 X$	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}:\ \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$				

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \ \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	C_{not}: $\|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}:\ \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^{0} X$	$\vert 00\rangle$	$I\vert b_0\rangle$	$\vert 00\rangle$	$C_{not}: \; \vert b_1\rangle_{ctl} \rightarrow \vert b_0\rangle_{tgt}$
	$\vert 01\rangle$	$I\vert b_0\rangle$	$\vert 01\rangle$	
	$\vert 10\rangle$	$X\vert b_0\rangle$	$\vert 11\rangle$	
	$\vert 11\rangle$	$XXX\vert b_0\rangle$	$\vert 10\rangle$	
$\bigwedge_{11}^{0} X$	$\vert 00\rangle$	$I\vert b_0\rangle$	$\vert 00\rangle$	$C_{not}: \; \vert b_1\rangle_{ctl} \rightarrow \vert b_0\rangle_{tgt}$
	$\vert 01\rangle$	$I\vert b_0\rangle$	$\vert 01\rangle$	
	$\vert 10\rangle$	$XXX\vert b_0\rangle$	$\vert 11\rangle$	
	$\vert 11\rangle$	$X\vert b_0\rangle$	$\vert 10\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$				

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}:\ \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}:\ \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	
	$\|01\rangle$	$XXX\|b_0\rangle$	$\|00\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	
	$\|01\rangle$	$XXX\|b_0\rangle$	$\|00\rangle$	
	$\|10\rangle$	$I\|b_0\rangle$	$\|10\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \rightarrow \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	
	$\|01\rangle$	$XXX\|b_0\rangle$	$\|00\rangle$	
	$\|10\rangle$	$I\|b_0\rangle$	$\|10\rangle$	
	$\|11\rangle$	$I\|b_0\rangle$	$\|11\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	$C_{not}: \|\hat{b}_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$XXX\|b_0\rangle$	$\|00\rangle$	
	$\|10\rangle$	$I\|b_0\rangle$	$\|10\rangle$	
	$\|11\rangle$	$I\|b_0\rangle$	$\|11\rangle$	

Example 5.4.1

A simplified example of the general $\bigwedge_x^i Q$ transformation is that of a 2-qubit system $|b_1 b_0\rangle$

Operator	Initial State	Action	Final State	Overall Effect
$\bigwedge_{10}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$X\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$XXX\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{11}^0 X$	$\|00\rangle$	$I\|b_0\rangle$	$\|00\rangle$	$C_{not}: \|b_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$I\|b_0\rangle$	$\|01\rangle$	
	$\|10\rangle$	$XXX\|b_0\rangle$	$\|11\rangle$	
	$\|11\rangle$	$X\|b_0\rangle$	$\|10\rangle$	
$\bigwedge_{00}^0 X$	$\|00\rangle$	$X\|b_0\rangle$	$\|01\rangle$	$C_{not}: \|\hat{b}_1\rangle_{ctl} \to \|b_0\rangle_{tgt}$
	$\|01\rangle$	$XXX\|b_0\rangle$	$\|00\rangle$	
	$\|10\rangle$	$I\|b_0\rangle$	$\|10\rangle$	
	$\|11\rangle$	$I\|b_0\rangle$	$\|11\rangle$	

Note that $\bigwedge_{01}^1 X$ has the effect of $C_{not}: |b_0\rangle_{ctl} \to |b_1\rangle_{tgt}$

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \leq i \leq N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \leq i \leq N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \leq i \leq N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

$$U_m = \begin{pmatrix} I^{(m)} & 0 \\ 0 & V_{N-m} \end{pmatrix}$$

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \dots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \le i \le N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

$$U_m = \begin{pmatrix} I^{(m)} & 0 \\ 0 & V_{N-m} \end{pmatrix}$$

where $I^{(m)}$ is the $m \times m$ identity matrix and V_{N-m} is an $(N-m) \times (N-m)$ unitary matrix with $0 \le m \le N-2$

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \leq i \leq N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

$$U_m = \begin{pmatrix} I^{(m)} & 0 \\ 0 & V_{N-m} \end{pmatrix}$$

where $I^{(m)}$ is the $m \times m$ identity matrix and V_{N-m} is an $(N-m) \times (N-m)$ unitary matrix with $0 \leq m \leq N-2$

Start with $m = N-2$ at its maximum value and the smallest possible unitary matrix V_2 representing only 2 qubits

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \leq i \leq N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

$$U_m = \begin{pmatrix} I^{(m)} & 0 \\ 0 & V_{N-m} \end{pmatrix}$$

where $I^{(m)}$ is the $m \times m$ identity matrix and V_{N-m} is an $(N-m) \times (N-m)$ unitary matrix with $0 \leq m \leq N-2$

Start with $m = N-2$ at its maximum value and the smallest possible unitary matrix V_2 representing only 2 qubits

$$U_{N-2} = \begin{pmatrix} I^{(N-2)} & 0 \\ 0 & V_2 \end{pmatrix}$$

Implementing general unitary transformations

As we have seen, any unitary transformation is just a rotation of the 2^n-dimensional vector space associated with an n-qubit system

Let $N = 2^n$ and define the standard basis as $\{|x_0\rangle, \ldots, |x_{N-1}\rangle\}$ such that $|x_i\rangle$ and $|x_{i+1}\rangle$ differ only by a single bit (called Gray code)

We can define a suitable Gray code by saying that for $0 \le i \le N-2$, define j_i as the bit that differs between $|x_i\rangle$ and $|x_{i+1}\rangle$ and B_i as the shared pattern of all the the other bits in the two vectors

U_m is an operator defined as

$$U_m = \begin{pmatrix} I^{(m)} & 0 \\ 0 & V_{N-m} \end{pmatrix}$$

where $I^{(m)}$ is the $m \times m$ identity matrix and V_{N-m} is an $(N-m) \times (N-m)$ unitary matrix with $0 \le m \le N-2$

$$U_{N-2} = \begin{pmatrix} I^{(N-2)} & 0 \\ 0 & V_2 \end{pmatrix}$$

Start with $m = N-2$ at its maximum value and the smallest possible unitary matrix V_2 representing only 2 qubits

Applying this operator is identical to applying $\bigwedge_x^j V_2$ where $x = x_{N-2}$ and $j = j_{N-2}$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \ldots, |x_{m-1}\rangle, \ldots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \ldots, |x_{m-1}\rangle, \ldots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \dots, |x_{m-1}\rangle$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \dots, |x_{m-1}\rangle$

This transformation will then have the property that

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \dots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1}$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \ldots, |x_{m-1}\rangle, \ldots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \ldots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1} \longrightarrow C_m = W_m^{-1}$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \dots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1} \longrightarrow C_m = W_m^{-1} \longrightarrow U_{m-1} = C_m U_m$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \ldots, |x_{m-1}\rangle, \ldots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \ldots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1} \longrightarrow C_m = W_m^{-1} \longrightarrow U_{m-1} = C_m U_m \longrightarrow U = U_0 = C_1 \cdots C_{N-2} U_{N-2}$$

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \ldots, |x_{m-1}\rangle, \ldots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \ldots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1} \longrightarrow C_m = W_m^{-1} \longrightarrow U_{m-1} = C_m U_m \longrightarrow U = U_0 = C_1 \cdots C_{N-2} U_{N-2}$$

W_m is defined iteratively starting by rewriting $|v_{m-1}\rangle$ as

Generating the general unitary operator

Given the unitary matrix U_{m-1}, and the basis $\{|x_0\rangle, \dots, |x_{m-1}\rangle, \dots, |x_{N-1}\rangle\}$, the basis vector $|x_{m-1}\rangle$ is the first on which the operator has a non-trivial action since the identity matrix is $(m-1) \times (m-1)$ and $V_{N-(m-1)}$ mixes the last $N-(m-1)$ basis vectors

$$|v_{m-1}\rangle = U_{m-1}|x_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$

The coefficient a_{N-1} can be made real by applying a global phase shift so we need to find a unitary transformation W_m that takes $|v_{m-1}\rangle$ to $|x_{m-1}\rangle$ and does not affect basis elements $|x_0\rangle, \dots, |x_{m-1}\rangle$

This transformation will then have the property that

$$U_m = W_m U_{m-1} \longrightarrow C_m = W_m^{-1} \longrightarrow U_{m-1} = C_m U_m \longrightarrow U = U_0 = C_1 \cdots C_{N-2} U_{N-2}$$

W_m is defined iteratively starting by rewriting $|v_{m-1}\rangle$ as

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

Generating the general unitary operator (cont.)

$$\begin{aligned}|v_{m-1}\rangle &= a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle \\ &= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle\end{aligned}$$

Generating the general unitary operator (cont.)

$$\begin{aligned} |v_{m-1}\rangle &= a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle \\ &= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle \end{aligned}$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}}$$

Generating the general unitary operator (cont.)

$$\begin{aligned}|v_{m-1}\rangle &= a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle \\ &= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle\end{aligned}$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2}$$

Generating the general unitary operator (cont.)

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$
$$= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2}$$

Generating the general unitary operator (cont.)

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$
$$= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2} \qquad \sin(\theta_{N-2}) = \frac{|a_{N-1}|}{c_{N-2}}$$

Generating the general unitary operator (cont.)

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$
$$= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2} \qquad \sin(\theta_{N-2}) = \frac{|a_{N-1}|}{c_{N-2}}$$

With these definitions, we can write a multiply controlled set of single qubit operators that acts on $|v_{m-1}\rangle$ to eliminate the $|x_{N-1}\rangle$ term

Generating the general unitary operator (cont.)

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$
$$= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2} \qquad \sin(\theta_{N-2}) = \frac{|a_{N-1}|}{c_{N-2}}$$

With these definitions, we can write a multiply controlled set of single qubit operators that acts on $|v_{m-1}\rangle$ to eliminate the $|x_{N-1}\rangle$ term

$$\bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a'_{N-2}|x_{N-2}\rangle, \quad a'_{N-2} = c_{N-2}$$

Generating the general unitary operator (cont.)

$$
\begin{aligned}
|v_{m-1}\rangle &= a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle \\
&= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle
\end{aligned}
$$

$$
a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}
$$

$$
c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2} \qquad \sin(\theta_{N-2}) = \frac{|a_{N-1}|}{c_{N-2}}
$$

With these definitions, we can write a multiply controlled set of single qubit operators that acts on $|v_{m-1}\rangle$ to eliminate the $|x_{N-1}\rangle$ term

$$
\bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a'_{N-2}|x_{N-2}\rangle, \quad a'_{N-2} = c_{N-2}
$$

The $K(-\phi_{N-2})$ eliminates the phase factor in front of $|x_{N-2}\rangle$

Generating the general unitary operator (cont.)

$$|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a_{N-1}|x_{N-1}\rangle$$
$$= a_{m-1}|x_{m-1}\rangle + \cdots + c_{N-2}\cos(\theta_{N-2})e^{i\phi_{N-2}}|x_{N-2}\rangle + c_{N-2}\sin(\theta_{N-1})|x_{N-1}\rangle$$

$$a_{N-2} = |a_{N-2}|e^{i\phi_{N-2}} \qquad \cos(\theta_{N-2}) = \frac{|a_{N-2}|}{c_{N-2}}$$

$$c_{N-2} = \sqrt{|a_{N-2}|^2 + |a_{N-1}|^2} \qquad \sin(\theta_{N-2}) = \frac{|a_{N-1}|}{c_{N-2}}$$

With these definitions, we can write a multiply controlled set of single qubit operators that acts on $|v_{m-1}\rangle$ to eliminate the $|x_{N-1}\rangle$ term

$$\bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})|v_{m-1}\rangle = a_{m-1}|x_{m-1}\rangle + \cdots + a'_{N-2}|x_{N-2}\rangle, \quad a'_{N-2} = c_{N-2}$$

The $K(-\phi_{N-2})$ eliminates the phase factor in front of $|x_{N-2}\rangle$ and the $R(\theta_{N-2})$ rotates amplitude from $|x_{N-1}\rangle$ to $|x_{N-2}\rangle$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m|x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m|x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m |x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i| e^{i\phi_i},$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m|x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i|e^{i\phi_i}, \quad a'_i = c_i,$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m |x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i| e^{i\phi_i}, \quad a'_i = c_i, \quad c_i = \sqrt{|a_i|^2 + |a_{i+1}|^2},$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m |x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i| e^{i\phi_i}, \quad a'_i = c_i, \quad c_i = \sqrt{|a_i|^2 + |a_{i+1}|^2}, \quad \cos\theta_i = \frac{|a_i|}{c_i},$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m|x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i|e^{i\phi_i}, \quad a'_i = c_i, \quad c_i = \sqrt{|a_i|^2 + |a_{i+1}|^2}, \quad \cos\theta_i = \frac{|a_i|}{c_i}, \quad \sin\theta_i = \frac{|a'_{i+1}|}{c_i}$$

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m|x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i|e^{i\phi_i}, \quad a'_i = c_i, \quad c_i = \sqrt{|a_i|^2 + |a_{i+1}|^2}, \quad \cos\theta_i = \frac{|a_i|}{c_i}, \quad \sin\theta_i = \frac{|a'_{i+1}|}{c_i}$$

This procedure guarantees a general unitary transformation but it is exponentially expensive and therefore is of limited value

Generating the general unitary operator (cont.)

The multiply controlled gate ensures that only the two basis vectors with the identical qubit pattern B_{N-2} are affected by this transformation

This same procedure is repeated for the next two lowest order qubit states until $|v_{m-1}\rangle = a'_m |x_{m-1}\rangle \equiv |x_{m-1}\rangle$ and this results in a composite operator

$$W_m = \bigwedge_{x_{m-1}}^{j_{m-1}} R(\theta_{m-1}) \bigwedge_{x_{m-1}}^{j_{m-1}} K(-\phi_{m-1}) \cdots \bigwedge_{x_{N-2}}^{j_{N-2}} R(\theta_{N-2}) \bigwedge_{x_{N-2}}^{j_{N-2}} K(-\phi_{N-2})$$

$$a_i = |a_i| e^{i\phi_i}, \quad a'_i = c_i, \quad c_i = \sqrt{|a_i|^2 + |a_{i+1}|^2}, \quad \cos\theta_i = \frac{|a_i|}{c_i}, \quad \sin\theta_i = \frac{|a'_{i+1}|}{c_i}$$

This procedure guarantees a general unitary transformation but it is exponentially expensive and therefore is of limited value

Making a practical quantum computer requires a more clever approach to take advantage of the inherent efficiency in the computations

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \quad |111\rangle, \quad |011\rangle, \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \quad |111\rangle, \quad |011\rangle, \quad |001\rangle, \qquad\qquad\qquad\qquad\qquad\qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \quad |111\rangle, \quad |011\rangle, \quad |001\rangle, \quad |000\rangle, \qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \quad |111\rangle, \quad |011\rangle, \quad |001\rangle, \quad |000\rangle, \quad |010\rangle, \quad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{ \quad |111\rangle, \quad |011\rangle, \quad |001\rangle, \quad |000\rangle, \quad |010\rangle, \quad |110\rangle, \qquad\qquad\qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\{\quad |111\rangle, \quad |011\rangle, \quad |001\rangle, \quad |000\rangle, \quad |010\rangle, \quad |110\rangle, \quad |100\rangle, \qquad \}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

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

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$
\begin{array}{ccccccccccc}
\{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\
\{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \}
\end{array}
$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{array}{ccccccccccc} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{array}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{array}{lllllllll} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle \quad \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle \quad \} \end{array}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{matrix} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{matrix}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \left(\quad \right)$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{array}{ccccccccc} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle \} \end{array}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & & \\ 0 & 1 & 0 & 0 & 0 & 0 & & \\ 0 & 0 & 1 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 1 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 1 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 1 & & \\ & & & & & & & \\ & & & & & & & \end{pmatrix}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{matrix} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{matrix}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & \end{pmatrix}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{matrix} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{matrix}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{pmatrix} \qquad U_5 = \begin{pmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{pmatrix}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{array}{cccccccccc} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{array}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{pmatrix} \qquad U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & & & \\ 0 & 1 & 0 & 0 & 0 & & & \\ 0 & 0 & 1 & 0 & 0 & & & \\ 0 & 0 & 0 & 1 & 0 & & & \\ 0 & 0 & 0 & 0 & 1 & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{pmatrix}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{matrix} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle & \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle & \} \end{matrix}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{pmatrix} \qquad U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & & & \\ 0 & 0 & 0 & 0 & 0 & & & \\ 0 & 0 & 0 & 0 & 0 & & & \end{pmatrix}$$

A 3-bit example

Consider a 3-qubit system where we wish to establish a Grey code basis

$$\begin{array}{lllllllll} \{ & |111\rangle, & |011\rangle, & |001\rangle, & |000\rangle, & |010\rangle, & |110\rangle, & |100\rangle, & |101\rangle \quad \} \\ \{ & |x_0\rangle, & |x_1\rangle, & |x_2\rangle, & |x_3\rangle, & |x_4\rangle, & |x_5\rangle, & |x_6\rangle, & |x_7\rangle \quad \} \end{array}$$

In this case, $n = 3$, $N = 2^n = 8$, and $0 \leq m \leq N - 2 = 6$

Let's look at the U_6 and U_5 operators

$$U_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{pmatrix} \qquad U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5 |x_5\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using $a_6 = |r| e^{i\phi_6}$,

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using $a_6 = |r|e^{i\phi_6}, \quad c_6 = \sqrt{|r|^2 + |u|^2}$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using $a_6 = |r|e^{i\phi_6}$, $c_6 = \sqrt{|r|^2 + |u|^2}$

$\cos\theta_6 = \frac{|r|}{c_6}$,

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using
$a_6 = |r|e^{i\phi_6}, \qquad c_6 = \sqrt{|r|^2 + |u|^2}$
$\cos\theta_6 = \frac{|r|}{c_6}, \qquad \sin\theta_6 = \frac{|u|}{c_6}$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$|v_5\rangle = U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using

$a_6 = |r|e^{i\phi_6}, \qquad c_6 = \sqrt{|r|^2 + |u|^2}$

$\cos\theta_6 = \frac{|r|}{c_6}, \qquad \sin\theta_6 = \frac{|u|}{c_6}$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$\begin{aligned} |v_5\rangle &= U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle \\ &= o|x_5\rangle + c_6\cos\theta_6|x_6\rangle + c_6\sin\theta_6|x_7\rangle \end{aligned}$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using

$a_6 = |r|e^{i\phi_6}, \qquad c_6 = \sqrt{|r|^2 + |u|^2}$

$\cos\theta_6 = \frac{|r|}{c_6}, \qquad \sin\theta_6 = \frac{|u|}{c_6}$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$\begin{aligned} |v_5\rangle &= U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle \\ &= o|x_5\rangle + c_6\cos\theta_6|x_6\rangle + c_6\sin\theta_6|x_7\rangle \end{aligned}$$

$$\bigwedge_{x_6}^{j_0} R(\theta_6) \bigwedge_{x_6}^{j_0} K(-\phi_6)|v_5\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using

$a_6 = |r|e^{i\phi_6}, \qquad c_6 = \sqrt{|r|^2 + |u|^2}$

$\cos\theta_6 = \frac{|r|}{c_6}, \qquad \sin\theta_6 = \frac{|u|}{c_6}$

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$\begin{aligned} |v_5\rangle &= U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle \\ &= o|x_5\rangle + c_6\cos\theta_6|x_6\rangle + c_6\sin\theta_6|x_7\rangle \end{aligned}$$

$$\bigwedge_{x_6}^{j_0} R(\theta_6) \bigwedge_{x_6}^{j_0} K(-\phi_6)|v_5\rangle = o|x_5\rangle + c_6|x_6\rangle$$

A 3-bit example (cont.)

Our goal is to generate a universal operator

$$U = U_0 = C_1 \cdots C_6 U_6$$

Starting with the U_5 matrix, we want an operator W_6 that satisfies $W_6 U_5 = U_6$

The U_5 operator leaves all the basis vectors from $|x_0\rangle \cdots |x_4\rangle$ alone so we can write

U_5 mixes the last three basis vectors

Now rewrite the coefficients using
$a_6 = |r|e^{i\phi_6}, \qquad c_6 = \sqrt{|r|^2 + |u|^2}$
$\cos\theta_6 = \frac{|r|}{c_6}, \qquad \sin\theta_6 = \frac{|u|}{c_6}$

This eliminates the $|x_7\rangle$ term and can be repeated to eliminate the $|x_6\rangle$ term

$$U_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & o & p & q \\ 0 & 0 & 0 & 0 & 0 & r & s & t \\ 0 & 0 & 0 & 0 & 0 & u & v & w \end{pmatrix}$$

$$\begin{aligned} |v_5\rangle &= U_5|x_5\rangle = o|x_5\rangle + r|x_6\rangle + u|x_7\rangle \\ &= o|x_5\rangle + c_6\cos\theta_6|x_6\rangle + c_6\sin\theta_6|x_7\rangle \end{aligned}$$

$$\bigwedge_{x_6}^{j_0} R(\theta_6) \bigwedge_{x_6}^{j_0} K(-\phi_6)|v_5\rangle = o|x_5\rangle + c_6|x_6\rangle$$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

$$P_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix}$$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

$$P_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|,$$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

$$P_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|, \quad P_{\frac{\pi}{4}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

$$P_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|, \quad P_{\frac{\pi}{4}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix} = |0\rangle\langle 0| + e^{i\frac{\pi}{4}}|1\rangle\langle 1|$$

Universally approximating set of gates

The problem we encountered in trying to make a general unitary operator out of simple gates cannot be solved exactly, however the Solovay-Kitaev theorem states that there are finite sets of gates that can approximate any unitary transformation to arbitrary accuracy efficiently

If we desire accuracy to a level of 2^{-d}, there exists a polynomial $p(d)$ such that any single-qubit unitary transformation can be approximated to the desired accuracy by a sequence of no more than $p(d)$ gates

We want to find a finite set of gates that can approximate all single-qubit transformations so that with the addition of the C_{not}, we can prepare any unitary operator

Take the Hadamard and the C_{not} gates and add two phase gates $P_{\frac{\pi}{2}}$ and $P_{\frac{\pi}{4}}$

$$P_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{pmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|, \quad P_{\frac{\pi}{4}} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix} = |0\rangle\langle 0| + e^{i\frac{\pi}{4}}|1\rangle\langle 1| = e^{i\frac{\pi}{8}}T(-\frac{\pi}{8})$$

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Why do we only use rational rotations, those where for some integer m, $R^m = I$?

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Why do we only use rational rotations, those where for some integer m, $R^m = I$?

When working on the Bloch sphere, rational rotations around single axes are able to construct irrational rotations in three dimensions

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Why do we only use rational rotations, those where for some integer m, $R^m = I$?

When working on the Bloch sphere, rational rotations around single axes are able to construct irrational rotations in three dimensions

$P_{\frac{\pi}{4}}$ is a rotation of $\frac{\pi}{4}$ about the z-axis of the Bloch sphere

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Why do we only use rational rotations, those where for some integer m, $R^m = I$?

When working on the Bloch sphere, rational rotations around single axes are able to construct irrational rotations in three dimensions

$P_{\frac{\pi}{4}}$ is a rotation of $\frac{\pi}{4}$ about the z-axis of the Bloch sphere

$S = HP_{\frac{\pi}{4}}H$ is a rotation of $\frac{\pi}{4}$ about the x-axis

Making approximating set of gates

An arbitrary transformation can be viewed as a rotation of the qubit on the Bloch sphere by any amount

In order to make an approximating set to an arbitrary precision, we need to be able to combine the 4 gates to get as close as desired to any rotation, even irrational ones

Why do we only use rational rotations, those where for some integer m, $R^m = I$?

When working on the Bloch sphere, rational rotations around single axes are able to construct irrational rotations in three dimensions

$P_{\frac{\pi}{4}}$ is a rotation of $\frac{\pi}{4}$ about the z-axis of the Bloch sphere

$S = HP_{\frac{\pi}{4}}H$ is a rotation of $\frac{\pi}{4}$ about the x-axis

It can be shown that simply combining these two rational rotations $V = P_{\frac{\pi}{4}}S$ gives an irrational rotation

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

By transforming V as HVH we create an irrational rotation about an orthogonal axis and with these two, any operation can be constructed, recall that another way of generating an arbitrary transformation was

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

By transforming V as HVH we create an irrational rotation about an orthogonal axis and with these two, any operation can be constructed, recall that another way of generating an arbitrary transformation was

$$W = K(\delta)T(\alpha)R(\beta)T(\gamma)$$

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

By transforming V as HVH we create an irrational rotation about an orthogonal axis and with these two, any operation can be constructed, recall that another way of generating an arbitrary transformation was

$$W = K(\delta)T(\alpha)R(\beta)T(\gamma)$$

$T(\alpha)$ rotates about the z-axis and $R(\beta)$ rotates about the y-axis

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

By transforming V as HVH we create an irrational rotation about an orthogonal axis and with these two, any operation can be constructed, recall that another way of generating an arbitrary transformation was

$$W = K(\delta)T(\alpha)R(\beta)T(\gamma)$$

$T(\alpha)$ rotates about the z-axis and $R(\beta)$ rotates about the y-axis

So we have that these 4 gates constitute a universally approximating set of gates

Making approximating set of gates (cont.)

Since V is irrational, any rotation can be approximated by a suitable power of V to within arbitrary precision

By transforming V as HVH we create an irrational rotation about an orthogonal axis and with these two, any operation can be constructed, recall that another way of generating an arbitrary transformation was

$$W = K(\delta)T(\alpha)R(\beta)T(\gamma)$$

$T(\alpha)$ rotates about the z-axis and $R(\beta)$ rotates about the y-axis

So we have that these 4 gates constitute a universally approximating set of gates

This is one possible set, others also exist