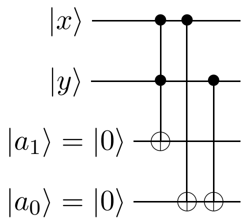
1 Classical Computers

Q.1 *Give a reversible circuit to add two single-bit numbers x and y, giving the output as a two-bit number.*

S.1 Note that there is never a unique circuit but in this case the obvious simple circuit is



In this case the order of the gates doesn't matter. The CCNOT get sets $a_1 = 1$ if $x = y = 1$ while the two CNOT gates set $a_0 = 1$ if precisely one of $x = 1$ or $y = 1$.

Q.2 *List all possible single-bit functions of a two-bit input x (so $f(x_1x_0)$ is 0 or 1 for each input). Give reversible circuit representations using the universal gate set $\{NOT, CNOT, CCNOT\}$ for all such functions with $f(00) = 0$. State a simple modification of these circuits to produce circuits for all such functions with $f(00) = 1$. Given that $\{NOT, CNOT\}$ is not a universal gate set, is it possible to construct all the functions without using CCNOT?*

S.2 There are four possible values for x, and $f(x)$ has two possible values for each choice, so there are $2^4 = 16$ functions which we can label $f_0, f_1, \ldots, f_{15}$. For the circuits we can take 3 bits in total, the two input bits and another bit initialised to 0 which will give the output bit – it is not necessary to include any further (ancillary) bits. Taking $x = x_1x_0$, we can write $CCNOT$ to mean a CCNOT gate with the output bit as the target and the two input bit as the controls, $CNOT_0$ ($CNOT_1$) to mean $CNOT$ acting on the output bit controlled by x_0 (x_1), and NOT to mean a NOT acting on the output. You can then easily draw the circuits by placing these gates in the same order left to right. (Actually, if you use these gates only the order does not matter – in general the order is important!) These are not unique circuits so you may find different correct circuits. The following table summarises all the details. Note that the list of all outputs for f_N is just N written as a 4-digit binary in these conventions. You could define the functions in other ways such as by using combinations of logical operations, but this way it is manifest that we have included all possible functions exactly once – the description in terms of logical operations is not unique.

x	00	01	10	11	Representation	Logic output
f_0	0	0	0	0	$Trivial$	0
f_1	0	0	0	1	$CCNOT$	$x_0\ AND\ x_1$
f_2	0	0	1	0	$CCNOT\ CNOT_1$	$(NOT\ x_0)\ AND\ x_1$
f_3	0	0	1	1	$CNOT_1$	x_1
f_4	0	1	0	0	$CCNOT\ CNOT_0$	$x_0\ AND\ (NOT\ x_1)$
f_5	0	1	0	1	$CNOT_0$	x_0
f_6	0	1	1	0	$CNOT_0\ CNOT_1$	$x_0\ XOR\ x_1$
f_7	0	1	1	1	$CCNOT\ CNOT_0\ CNOT_1$	$x_0\ OR\ x_1$
f_8	1	0	0	0	$CCNOT\ CNOT_0\ CNOT_1\ NOT$	$x_0\ NOR\ x_1$
f_9	1	0	0	1	$CNOT_0\ CNOT_1\ NOT$	$x_0\ NXOR\ x_1$
f_{10}	1	0	1	0	$CNOT_0\ NOT$	$NOT\ x_0$
f_{11}	1	0	1	1	$CCNOT\ CNOT_0\ NOT$	$x_0\ NAND\ (NOT\ x_1)$
f_{12}	1	1	0	0	$CNOT_1\ NOT$	$NOT\ x_1$
f_{13}	1	1	0	1	$CCNOT\ CNOT_1\ NOT$	$(NOT\ x_0)\ NAND\ x_1$
f_{14}	1	1	1	0	$CCNOT\ NOT$	$x_0\ NAND\ x_1$
f_{15}	1	1	1	1	NOT	1

Note that the second half (those with $f(00) = 1$) are the NOT of a function from the first half, specifically f_{15-N} is related to f_N in this way. So, if you have constructed circuits for the functions with $f(00) = 0$, you can simply include a NOT gate at the end of the output to produce the remaining circuits. If you have used the circuits described above, the NOT gate can be placed anywhere on the output line – but note this is not true in general.

These realisations are not unique, but we cannot avoid using $CCNOT$ for all of them. In terms of the information given in the question, the simple argument is that f_1 implements $CCNOT$. If we could construct it from just $\{NOT, CNOT\}$ then we would be able to use that circuit anywhere we wanted a $CCNOT$ gate. Hence, we would have shown that $\{NOT, CNOT\}$ is a universal gate set, since we are told $\{NOT, CNOT, CCNOT\}$ is. Clearly this contradicts the statement in the question so it must not be possible to construct a circuit for f_1 without using any $CCNOT$ gates.

Actually, this argument is not quite correct since we only require the circuit to behave as CCNOT when the target is initialised to 0. This leaves the possibility that we could construct such a circuit without a CCNOT gate and it would behave as a CCNOT gate if the target was initially 0, but differently if the target was initially 1. However, it is easy to see that if we could construct any such a circuit, we could construct a CCNOT gate. To do this, take the circuit with the output bit initialised to 0. Then the output will be 0 unless both inputs were 1. This means that the output indicates whether or not the CCNOT gate with these two inputs as control bits should act trivially (if output is 0) or as a NOT gate (if output is 1) on the target of the CCNOT gate. So we can now take this output and use it as the control bit for a CNOT gate acting on another bit which is the target bit of the CCNOT gate which we have then constructed.

Q.3 *Give definitions of the complexity classes P, NP, PSPACE and EXP, and prove the inclusions* $P \subseteq NP \subseteq PSPACE \subseteq EXP$.

S.3 The definitions are bookwork. We interpret the inclusions in terms of problems. Any problem in P is clearly in NP; we can check that a solution is correct in polynomial time simply by solving the problem in polynomial time to see if the actual solution matches the proposed solution. Any problem in NP is in PSPACE as we can simply check all the possible solutions one after the other until one works. This may take a very long time, but it will only require polynomial space since any algorithm in NP requires only polynomial resource. And everything is in EXP.

2 Quantum Computers

Q.4 *Show that*

$$R_{\hat{n}}(\theta) = \cos(\theta/2)I - i\sin(\theta/2)(n_x X + n_y Y + n_z Z),$$

where $\hat{n} = (n_x, n_y, n_z)$ *is a unit vector in* $\mathbb{R}^3$*, is a unitary operator. Show that if a single qubit has the state*

$$\hat{\rho} = \frac{1}{2}(I + \mathbf{r}\cdot\boldsymbol{\sigma}) = \frac{1}{2}(I + xX + yY + zZ),$$

where $\mathbf{r} = (x, y, z)$ *is a unit vector (that is, this is a pure state), then the effect of the unitary operator* $R_{\hat{n}}(\theta)$ *is to rotate* $\mathbf{r}$ *about the axis* $\hat{n}$ *in the Bloch sphere by an angle* θ*.*

S.4 To show it is unitary is just a calculation, but to show that we have a rotation can be done in different ways. Note first that conceptually we know the result must be a rotation since this is a unitary transformation of a single-qubit pure state – hence it must map the Bloch sphere to itself and preserve inner products (which are determined by relative positions on the Bloch sphere). The question is then, precisely what rotation is taking place.

There are several ways to approach this problem. A nice, but slightly abstract approach is to construct an argument by showing that $R_{(0,0,1)}(\theta)$ rotates by angle θ around the z-axis (which is a straightforward calculation), and then use symmetry to argue for the result. More precisely, we use the fact that we can always choose our coordinates or our basis vectors in $\mathbb{R}^3$ so that any given vector, taking $\hat{\mathbf{n}}$ in this case, is pointing along the new z-axis which we could label the z'-axis. Then, since the statement is not dependent on any specific choice of coordinates or basis, we have the result provided the operators $R_{\hat{n}}$ and $\hat{\rho}$ take the same form in any orthonormal basis. This is almost true. Under a change of basis we have $n_i \to n'_i = M_{ij}n_j$ and $r_i \to r'_i = M_{ij}r_j$ where M is an orthogonal matrix implementing the rotation. Now, if we also define $\sigma'_i = M_{ij}\sigma_j$ then $\mathbf{r}\cdot\boldsymbol{\sigma} = r_i\sigma_i = r'_i\sigma'_i$ so the operators take the same form in any orthonormal basis provided we can interpret the σ'_i as Pauli σ-matrices. It is straightforward to check that indeed we have $\sigma'_i\sigma'_j + \sigma'_j\sigma'_i = 2\delta_{ij}I$ etc.

Below we outline a direct calculation.

We know X, Y, and Z are unitary, so

$$R^\dagger_{\hat{n}}(\theta) = \cos(\theta/2)I + i\sin(\theta/2)(n_x X + n_y Y + n_z Z) = R_{\hat{n}}(-\theta).$$

Multiplying,

$$\begin{aligned} R_{\hat{n}}(\theta)R_{\hat{n}}(-\theta) &= \cos^2(\theta/2)I + \sin^2(\theta/2)(n_x^2X^2 + n_xn_y(XY+YX) + n_xn_z(XZ+ZX) \\ &\quad + n_y^2Y^2 + n_yn_z(YZ+ZY) + n_z^2Z^2. \end{aligned}$$

Now the Pauli matrices satisfy $XY + YX = XZ + ZX = YZ + ZY = 0$, and $X^2 = Y^2 = Z^2 = I$, so

$$R_{\hat{n}}(\theta)R_{\hat{n}}(-\theta) = [\cos^2(\theta/2) + \sin^2(\theta/2)(n_x^2 + n_y^2 + n_z^2)]I = I$$

as $\hat{\mathbf{n}}$ is a unit vector. Thus, $R^\dagger = R^{-1}$, and this is a unitary operator.

Applying this transformation to $\hat{\rho}$,

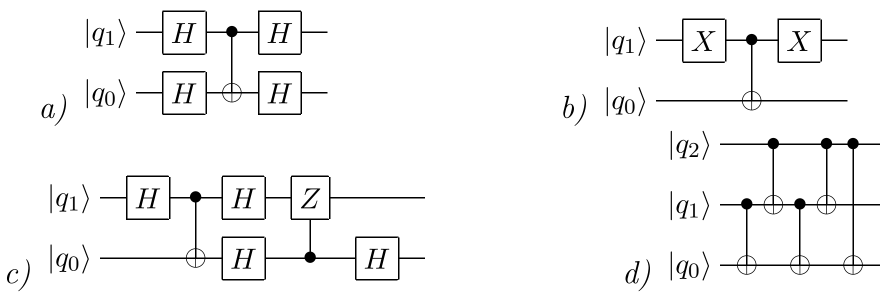
$$\begin{aligned}
\hat{\rho}' &= R^\dagger \hat{\rho} R = \frac{1}{2}[\cos(\theta/2)I + i\sin(\theta/2)(n_x X + n_y Y + n_z Z)](I + xX + yY + zZ) \\
&\quad \times[\cos(\theta/2)I - i\sin(\theta/2)(n_x X + n_y Y + n_z Z)] \\
&= \frac{1}{2}\left\{\cos^2(\theta/2)(I + xX + yY + zZ) + i\cos(\theta/2)\sin(\theta/2)[n_x X + n_y Y + n_z Z, xX + yY + zZ]\right. \\
&\quad \left. + \sin^2(\theta/2)[I + (n_x X + n_y Y + n_z Z)(xX + yY + zZ)(n_x X + n_y Y + n_z Z)]\right\} \\
&= \frac{1}{2}\left\{I + \cos^2(\theta/2)(xX + yY + zZ)\right. \\
&\quad -2\cos(\theta/2)\sin(\theta/2)[(n_x y - n_y x)Z + (n_y z - n_z y)X + (n_z x - n_x z)Y] \\
&\quad + \sin^2(\theta/2)(n_x x I + i n_x y Z - i n_x z Y - i n_y x Z + n_y y I + i n_y z X + i n_z x Y - i n_z y X + n_z z I) \\
&\quad \left. \times\, (n_x X + n_y Y + n_z Z)]\right\} \\
&= \frac{1}{2}\left\{I + \cos^2(\theta/2)(xX + yY + zZ) - \sin(\theta)[(n_x y - n_y x)Z + (n_y z - n_z y)X + (n_z x - n_x z)Y]\right. \\
&\quad \left. + \sin^2(\theta/2)[(2n_x \hat{\mathbf{n}}\cdot\mathbf{r} - x)X + (2n_y \hat{\mathbf{n}}\cdot\mathbf{r} - y)Y + (2n_z \hat{\mathbf{n}}\cdot\mathbf{r} - z)Z]\right\}
\end{aligned}$$

If we write $\mathbf{r} = (\hat{\mathbf{n}}\cdot\mathbf{r})\hat{\mathbf{n}} + \mathbf{r}_\perp$, where $\mathbf{r}_\perp$ is the component of $\mathbf{r}$ which is orthogonal to $\hat{\mathbf{n}}$, this becomes

$$\hat{\rho}' = \frac{1}{2}[I + (\hat{\mathbf{n}}\cdot\mathbf{r})\hat{\mathbf{n}}\cdot\mathbf{X} + \cos\theta\mathbf{r}_\perp\cdot\mathbf{X} + \sin\theta(\mathbf{r}_\perp\times\hat{\mathbf{n}})\cdot\mathbf{X}],$$

which indeed gives a rotation about $\hat{\mathbf{n}}$ by an angle θ.

Q.5 *Compute the action of the circuits below on states in the computational basis. Give simpler equivalent circuits where possible.*

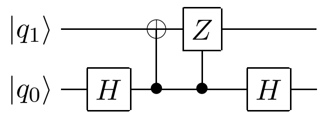


S.5 (a) First, recall the states $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$. Then $|00\rangle \to |++\rangle \to |++\rangle \to |00\rangle$, $|01\rangle \to |+-\rangle \to |--\rangle \to |11\rangle$, $|10\rangle \to |-+\rangle \to |-+\rangle \to |10\rangle$, $|11\rangle \to |--\rangle \to |+-\rangle \to |01\rangle$. This is equivalent to CNOT with q_0 as the control bit.

(b) This is very straightforward to calculate directly for each computational basis state. $|00\rangle \to |10\rangle \to |11\rangle \to |01\rangle$. $|01\rangle \to |11\rangle \to |10\rangle \to |00\rangle$. $|10\rangle \to |00\rangle \to |00\rangle \to |10\rangle$. $|11\rangle \to |01\rangle \to |01\rangle \to |11\rangle$.

Alternatively, note that two NOT gates act on q_1 so it is unchanged. As it is used as the control after the first NOT, q_0 is changed precisely when initially $q_1 = 0$.

(c) This is easier to do if we use the result in part (a), together with the fact that $H^2 = I$ which allow us to add two Hadamard gates to q_0 to the left of the $CNOT$ gate, to write it as



Then

$$|00\rangle \to |0+\rangle \to \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \to \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \to \frac{1}{\sqrt{2}}(|0+\rangle - |1-\rangle)$$

$$|01\rangle \to |0-\rangle \to \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \to \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \to \frac{1}{\sqrt{2}}(|0+\rangle + |1-\rangle)$$

$$|10\rangle \to |1+\rangle \to \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle) \to \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle) \to \frac{1}{\sqrt{2}}(|1+\rangle + |0-\rangle)$$

$$|11\rangle \to |1-\rangle \to \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle) \to \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle) \to \frac{1}{\sqrt{2}}(|1+\rangle - |0-\rangle)$$

(d) If q_2 is zero, the circuit simplifies to just two CNOTs (control q_1, target q_0), which is trivial. For $q_2 = 1$ you can directly calculate $|10q_0\rangle \to |10q_0\rangle \to |11q_0\rangle \to |11(q_0 \oplus 1)\rangle \to |10(q_0 \oplus 1)\rangle \to |10q_0\rangle$. $|11q_0\rangle \to |11(q_0 \oplus 1)\rangle \to |10(q_0 \oplus 1)\rangle \to |10(q_0 \oplus 1)\rangle \to |11(q_0 \oplus 1)\rangle \to |11q_0\rangle$.

Alternatively, for $q_2 = 1$ note that q_2 and q_1 are not changes, since for q_1 we have two NOTs which gives the identity. For q_0 since q_1 has a NOT between the two $CNOT$s where q_1 is the control, exactly one of them will act as NOT on q_0. However, the final $CNOT$ with control $q_2 = 1$ acts as another NOT on q_0, so it is also unchanged.

Thus, the action in the computational basis is completely trivial. This is a trivial unitary. The circuit can then be simplified to simply 3 horizontal lines.

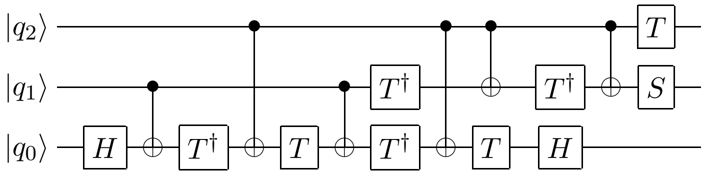
Q.6 *Show that $S = \frac{1}{2}(1 + X_iX_j + Y_iY_j + Z_iZ_j)$ defines a swap operator, interchanging the state of qubits i and j.*

S.6 Consider the action on computational basis states:

- $X_iX_j|00\rangle = |11\rangle$, $Y_iY_j|00\rangle = -|11\rangle$, $Z_iZ_j|00\rangle = |00\rangle$, so $S|00\rangle = |00\rangle$.
- $X_iX_j|01\rangle = |10\rangle$, $Y_iY_j|01\rangle = |10\rangle$, $Z_iZ_j|01\rangle = -|01\rangle$, so $S|01\rangle = |10\rangle$.
- $X_iX_j|10\rangle = |01\rangle$, $Y_iY_j|10\rangle = |01\rangle$, $Z_iZ_j|10\rangle = -|10\rangle$, so $S|10\rangle = |01\rangle$.
- $X_iX_j|11\rangle = |00\rangle$, $Y_iY_j|11\rangle = -|00\rangle$, $Z_iZ_j|11\rangle = |11\rangle$, so $S|11\rangle = |11\rangle$.

Alternatively, you could multiply out the matrices.

Q.7 *By considering the action on computational basis states, show that the circuit given in lectures (and reproduced below) does implement the Toffoli gate (CCNOT).*



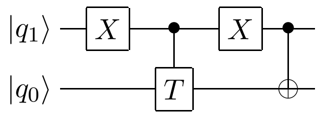
S.7 For $q_2 = 0$ T does nothing to $|q_2\rangle$ while the phase gates on q_1 are $ST^\dagger T^\dagger = I$. For $q_1 = 0$, the action on q_0 is $HTT^\dagger TT^\dagger H = I$. For $q_1 = 1$, the action on q_0 is $HTT^\dagger XTT^\dagger XH = I$.

For $q_2 = 1$, the action on q_1 is $SXT^\dagger XT^\dagger$; for $q_1 = 0$ this is an $e^{-i\pi/4}$ phase which cancels the phase from the T acting on q_0. For $q_1 = 1$ it is an $e^{i\pi/4}$ phase, so the upper two qubits contribute a $e^{i\pi/2}$ phase. For $q_2 = 1$, $q_1 = 0$ the action on q_0 is $HTXT^\dagger TXT^\dagger H = I$. For $q_2 = 1, q_1 = 1$, the action on q_0 is $HTXT^\dagger XTXT^\dagger XH$. It seems easiest at this stage to multiply out explicitly: $TX = \begin{pmatrix} 0 & 1 \\ e^{i\pi/4} & 0 \end{pmatrix}$ and $T^\dagger X = \begin{pmatrix} 0 & 1 \\ e^{-i\pi/4} & 0 \end{pmatrix}$ so $TXT^\dagger X = \begin{pmatrix} e^{-i\pi/4} & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$, hence $TXT^\dagger XTXT^\dagger X = \begin{pmatrix} e^{-i\pi/2} & 0 \\ 0 & e^{i\pi/2} \end{pmatrix} = -iZ$. So, the action on q_0 is $-iHZH = -iX$.

The overall phase cancels with the phase from the gates on q_2, q_1. So this circuit acts as the identity on the states with $q_2 = 0$ or $q_1 = 0$, and when $q_2 = q_1 = 1$, it acts as NOT on q_0, realising the Toffoli gate.

Q.8 *Consider a two-qubit system. Construct a circuit to realise the operation* $U = \begin{pmatrix} T & 0 \\ 0 & X \end{pmatrix}$, *where* T, X *are the standard* 2×2 *matrices.*

S.8 Acting on 2-qubit computational basis states $|q_1 q_0\rangle$, this is T on $|q_0\rangle$ if the $q_1 = 0$, and X on $|q_0\rangle$ if the $q_1 = 1$. Hence we want



It is also correct to have the CNOT gate on the left.

Q.9 *Consider a two-qubit system. Construct a circuit to realise the operation* $U = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$

S.9 This is just a NOT on both bits which you can see from the action of U on the computational basis states.

If you don't spot the simple solution above, the methodical approach is to write U as a product of unitary matrices which are each 2×2 unitary matrices $U_{ij}^\dagger$ embedded in the 4×4 identity matrix, where $U_{ij} = U_{ji}$ has non-trivial entries in the ii, ij, ji and jj components only. We do this by multiplying U by suitable U_{ij} so that, working left to right and up to down, we set the off-diagonal components of U to 0, essentially by doing row reduction (but constrained since we can only use unitary matrices).

So, we start by choosing U_{14} to make the 4th element in the 1st row of $U_{14}U$ vanish. This requires the component $(U_{14})_{44} = 0$, so for unitarity we need $(U_{14})_{41} = (U_{14})_{14} = 1$ and then we see $(U_{14})_{11} = 0$. (Actually, we could have arbitrary phases for the 14 and 41 components, but we fix the 14 component to 1 so that the 11 component of $U_{14}U$ is 1, and it doesn't matter what the other phase is so we choose it to be simply 1.) So, we have

$$U_{14} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad U_{14}U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

To continue we could choose U_{23} so that the 32 component of $U_{23}U_{14}U$ vanishes. However, we see that $U_{14}U$ is already a unitary matrix with only a 2×2 non-trivial block so we define this to be $U_{23}^\dagger$ and have $U_{14}U = U_{23}^\dagger$ leading to the result $U = U_{14}^\dagger U_{23}^\dagger$.

The unitary matrices U_{14} and U_{23} do not act on single qubits so we need to use Gray codes to convert the basis so that they do act on single qubits. Since U_{14} acts on the basis states $|00\rangle$ and $|11\rangle$ we can use the Gray code $00 \to 01 \to 11$. Similarly for U_{23} we can use $01 \to 00 \to 10$. These are both the same transformation where we use $CNOT$ on $|q_0\rangle$ when the control bit $q_1 = 0$ which we may write as $C_1 NOT_0$. This is implemented in the circuit by $X_1 C_1 NOT_0 X_1$.

In the new basis U_{14} is NOT on $|q_1\rangle$ when $q_0 = 1$ while U_{23} is also NOT on $|q_1\rangle$ but when $q_0 = 0$. Therefore the overall effect is just NOT on $|q_1\rangle$, i.e. X_1. Finally we must transform back to the original basis, again using $X_1 C_1 NOT_0 X_1$.

So, the final circuit is $(X_1 C_1 NOT_0 X_1) X_1 (X_1 C_1 NOT_0 X_1) = X_1 (C_1 NOT_0 X_1 C_1 NOT_0) X_1 = X_1 (X_1 X_0) X_1 = X_1 X_0$.

Q.10 *Consider a two-qubit system. We wish to construct a circuit to realise the operation*

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{2} & \frac{1}{2} & -\frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{pmatrix}$$

(a) *First decompose this operator in terms of unitary operators U_1, U_2, U_3 which each act non-trivially on a two-dimensional subspace of the Hilbert space, $U = U_1 U_2 U_3$.*

(b) *Use CNOTs to convert the operators which do not act on a subspace corresponding to a single qubit into ones that do.*

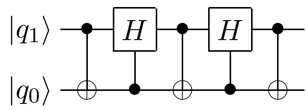
(c) *Draw the resulting quantum circuit.*

S.10 (a) As in the previous question, choose unitaries U_{ij} to transform U into the identity by row reduction. In this example only the lower right 3×3 block is non-trivial so really it is a 3×3 problem embedded into 4×4 matrices. The matrices we need are U_{23}, U_{24} and U_{34} which in the notation of the question can be chosen to be (note this is not unique so if you have 3 other matrices that are unitary, non-trivial only in 2×2 submatrices and multiply to give U, that is a valid alternative solution – you will end up with a different but equivalent quantum circuit, and it may or may not be obvious how to relate the different circuits)

$$U_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad U_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \end{pmatrix}, \quad U_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

(b) U_2 is a controlled-Hadamard with target $|q_1\rangle$ and control $|q_0\rangle$. U_3 is a CNOT with target $|q_0\rangle$ and control $|q_1\rangle$. So it is only U_1 we need to address: it acts on the subspace spanned by $|01\rangle$ and $|10\rangle$. Acting with CNOT, this is $|01\rangle$ and $|11\rangle$, so it's CNOT U_2 CNOT.

(c)



Q.11 *Defining the error $E(U,V) \equiv \max_\psi ||(U-V)|\psi\rangle||$, show that $E(R_{\hat{n}}(\alpha), R_{\hat{n}}(\beta)) = \frac{1}{\sqrt{2}}|1 - e^{i(\alpha-\beta)}|$.*

S.11 Without loss of generality, change our basis so that $\hat{n} = (0, 0, 1)$, so $R_{\hat{n}}(\alpha) = R_z(\alpha)$. In the Bloch sphere representation, this is represented as a rotation in the $x - y$ plane, and the error is maximised if we consider vectors in the $x - y$ plane, that is, we take $\mathbf{r}$ orthogonal to $\mathbf{n}$. In terms of the state, this is

$$|\psi(\theta)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle),$$

where θ is the angle in the $x - y$ plane. $R_z(\alpha)$ acts as $\theta \to \theta + \alpha$.

$$||(R_z(\alpha) - R_z(\beta)|\psi(\theta)\rangle|| = |||\psi(\theta+\alpha)\rangle - |\psi(\theta+\beta)\rangle|| = \frac{1}{\sqrt{2}}|e^{i(\theta+\alpha)} - e^{i(\theta+\beta)}| = \frac{1}{\sqrt{2}}|1 - e^{i(\alpha-\beta)}|.$$

Q.12 *Delayed measurement: In the discussion of quantum teleportation, observers were often required to perform operations which depended on the result of a measurement. In a quantum circuit, we would represent such actions by performing a measurement on one qubit and then acting with a unitary on another if the result of the measurement was 1.*

Show that such an operation can always be replaced by a controlled-unitary gate, with the measurement postponed to the end of the computation.

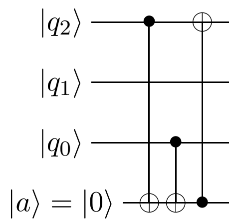
S.12 If the first qubit is initially in a state $|q_1\rangle = \alpha|0\rangle + \beta|1\rangle$, and the second qubit is in a state $|q_2\rangle$, acting with a controlled-unitary gate will put the system in the state $\alpha|0\rangle \otimes |q_2\rangle + \beta|1\rangle \otimes U|q_2\rangle$. Measuring the first qubit, we either measure 0, leaving the second qubit in the state $|q_2\rangle$, or we measure 1, leaving the second qubit in the state $U|q_2\rangle$. Mathematically, this is equivalent to measuring the first qubit and then acting on the second qubit with U if the measurement result is 1. Also, in both cases the probabilities of these outcomes are $|\alpha|^2$ and $|\beta|^2$.

Actually, we should consider the more general case when the two qubits may be entangled. In that case we can always write the initial state as $\alpha|0\rangle \otimes |\phi\rangle + \beta\ |1\rangle \otimes |\psi\rangle$ but by exactly the same argument, either way we will measure 0 with probability $|\alpha|^2$ and get final state $|0\rangle \otimes |\phi\rangle$ or 1 with probability $|\beta|^2$ and get final state $|1\rangle \otimes |\psi\rangle$.

Of course, if the two qubits are spatially separated, it is very difficult to perform the joint quantum operation necessary to implement the controlled unitary. It is therefore often advantageous to actually perform measurements first and communicate the classical information instead. However, theoretically we can always do measurements at the end and this simplifies our discussion of quantum circuits since we can always first implement a unitary transformation and then at the end make measurements.

3 Error-correcting codes

Q.13 *Suppose three qubits were initially in some state $|\psi\rangle = \alpha|000\rangle + \beta|111\rangle$ in the usual code subspace for single qubit bit-flip error correction, and have subsequently become entangled with an environment, such that the joint state is $|e_1\rangle \otimes |\psi\rangle + |e_2\rangle \otimes X_2|\psi\rangle$. Show that the circuit below will return the qubits to their original state, transferring the entanglement with the environment to the ancillary qubit $|a\rangle$.*



S.13 After the first gate, the state is

$$|e_1\rangle \otimes (\alpha|000\rangle \otimes |0\rangle + \beta|111\rangle \otimes |1\rangle) + |e_2\rangle \otimes (\alpha|100\rangle \otimes |1\rangle + \beta|011\rangle \otimes |0\rangle).$$

After the second gate, the state is

$$|e_1\rangle \otimes (\alpha|000\rangle \otimes |0\rangle + \beta|111\rangle \otimes |0\rangle) + |e_2\rangle \otimes (\alpha|100\rangle \otimes |1\rangle + \beta|011\rangle \otimes |1\rangle).$$

Finally, the state is

$$|e_1\rangle\otimes(\alpha|000\rangle\otimes|0\rangle+\beta|111\rangle\otimes|0\rangle)+|e_2\rangle\otimes(\alpha|000\rangle\otimes|1\rangle+\beta|111\rangle\otimes|1\rangle) = |e_1\rangle\otimes|\psi\rangle\otimes|0\rangle+|e_2\rangle\otimes|\psi\rangle\otimes|1\rangle,$$

so $|\psi\rangle$ is an overall factor, and the state of the environment is entangled with the ancilla, as desired.

Q.14 *Construct a 3-qubit code subspace protecting against single phase errors, that is against the random action of Z on any single qubit, by showing that the error syndromes X_0X_1 and X_0X_2 will diagnose single phase errors, and finding their +1, +1 eigenspace.*

S.14 The error Z_0 anticommutes with both error syndromes, mapping the $+1,+1$ eigenspace to the $-1,-1$ eigenspace. Z_1 anticommutes with the first error syndrome, mapping the $+1,+1$ eigenspace to the $-1,+1$ eigenspace. Z_2 anticommutes with the second error syndrome, mapping the $+1,+1$ eigenspace to the $+1,-1$ eigenspace. Thus if we take the $+1,+1$ eigenspace as the code subspace, the errors will each map to a distinct eigenspace, and the errors can be distinguished by these syndromes.

The $+1,+1$ eigenspace is most easily constructed by using $HZ = XH$, so $H^{\otimes 3}$ will map the $+1,+1$ eigenspace of Z_0Z_1 and Z_0Z_2 to the $+1,+1$ eigenspace of X_0X_1 and X_0X_2. Thus, suitable codewords are

$$|\bar{0}\rangle = H^{\otimes 3}|000\rangle = \frac{1}{\sqrt{8}}(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)$$

and

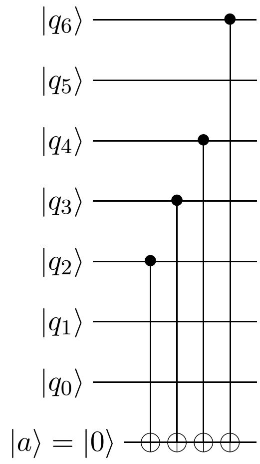
$$|\bar{1}\rangle = H^{\otimes 3}|111\rangle = \frac{1}{\sqrt{8}}(|000\rangle - |001\rangle - |010\rangle + |011\rangle - |100\rangle + |101\rangle + |110\rangle - |111\rangle.)$$

Q.15 *In classical codes, greater redundancy reduces the risk of errors; if we have five bits for each logical bit, we are protected against two single bit errors. Consider the 5 qubit code* $|\bar{0}\rangle = |00000\rangle$, $|\bar{1}\rangle = |11111\rangle$. *Does this protect against any two single bit flip errors? Justify your answer.*

S.15 Yes; suitable error syndrome operators are $M_0 = Z_1Z_2Z_3Z_4$, $M_1 = Z_0Z_2Z_3Z_4$, $M_2 = Z_0Z_1Z_3Z_4$, $M_3 = Z_0Z_1Z_2Z_3$ (note $Z_1Z_2Z_3Z_4$ is not independent). These define 16 two-dimensional eigenspaces which make up the five-qubit Hilbert space. There are 5 possible single-qubit bit flip errors X_i, and 10 possible double bit flip errors X_iX_j, which all map to distinct eigenspaces of these error syndromes.

Q.16 *Suppose we have a state $|\psi\rangle$ which was encoded using the Steane code, and we want to check whether a Y_2 error has acted on it. Identify an appropriate error syndrome to diagnose this error, and draw a quantum circuit to measure this syndrome.*

S.16 We could detect this by measuring either M_2 or N_2, which both anticommute with Y_2. Suppose we measure N_2; the circuit is



Q.17 *How many distinct subspaces do we need to encode a single logical qubit to allow for recovery from independent single qubit errors acting on up to two qubits in an n-qubit system? What is the smallest number of qubits where such an encoding could exist?*

S.17 We need a code subspace, $3n$ subspaces for single errors, and $\frac{9}{2}n(n-1)$ subspaces for double errors: $n(n-1)$ each for X_iY_j, X_iZ_j and Y_iZ_j, and $\frac{1}{2}n(n-1)$ each for X_iX_j, Y_iY_j and Z_iZ_j. So in total $\frac{1}{2}(9n^2 - 3n + 2)$ subspaces. $2^n \geq 9n^2 - 3n + 2$ for $n \geq 10$.

Q.18 *Demonstrate that if we have two logical qubits encoded using the Steane code, $\overline{CNOT} = \prod_{i=1}^{7} CNOT_{ii}$ implements the CNOT operation on the logical qubits, where $CNOT_{ii}$ is the CNOT operation between the ith physical qubit of the first codeword and the ith physical qubit of the second codeword.*

Hint: This can be solved elegantly using the representation of the logical $|\bar{0}\rangle$ and $|\bar{1}\rangle$ in terms of the M_a.

S.18 Assume the state of the control qubit is

$$|\bar{0}\rangle = \frac{1}{2^{3/2}}(1 + M_0)(1 + M_1)(1 + M_2)|0000000\rangle.$$

the $\overline{CNOT}$ flips every bit in the target where the bit in the control is 1. So if the control is $M_a|0000000\rangle$, the $\overline{CNOT}$ acts as M_a on the target, etc.

Thus, when the control is $|\bar{0}\rangle$, the $\overline{CNOT}$ acts as

$$|\bar{0}\rangle|\psi\rangle = \frac{1}{2^{3/2}}(1+M_0)(1+M_1)(1+M_2)|0000000\rangle|\psi\rangle \to \frac{1}{2^{3/2}}(|0000000\rangle|\psi\rangle + M_0|0000000\rangle M_0|\psi\rangle + \ldots)$$

But we assume the state $|\psi\rangle$ is in the code subspace, which is the $+1$ eigenspace of all the M_a, so this state is just

$$\frac{1}{2^{3/2}}(|0000000\rangle|\psi\rangle + M_0|0000000\rangle|\psi\rangle + \ldots) = |\bar{0}\rangle|\psi\rangle.$$

Similarly, if the state of the control qubit is

$$|\bar{1}\rangle = \frac{1}{2^{3/2}}(1 + M_0)(1 + M_1)(1 + M_2)\bar{X}|0000000\rangle.$$

the $\overline{CNOT}$ flips every bit in the target where the bit in the control is 1. So if the control is $\bar{X}|0000000\rangle$, the $\overline{CNOT}$ acts as $\bar{X}$ on the target, and if the control is $M_a\bar{X}|0000000\rangle$, the

$\overline{CNOT}$ acts as $M_a\bar{X}$ on the target, etc. Acting on the code subspace, $M_a\bar{X} = \bar{X}$. Thus, when the control is $|\bar{1}\rangle$, and the target $|\psi\rangle$ is in the code subspace,

$$\begin{aligned} |\bar{1}\rangle|\psi\rangle &= \frac{1}{2^{3/2}}(1+M_0)(1+M_1)(1+M_2)\bar{X}|0000000\rangle|\psi\rangle \\ &\to \frac{1}{2^{3/2}}(\bar{X}|0000000\rangle\bar{X}|\psi\rangle + M_0\bar{X}|0000000\rangle M_0\bar{X}|\psi\rangle + \ldots) \\ &= \frac{1}{2^{3/2}}(\bar{X}|0000000\rangle\bar{X}|\psi\rangle + M_0\bar{X}|0000000\rangle\bar{X}|\psi\rangle + \ldots) = |\bar{1}\rangle\bar{X}|\psi\rangle, \end{aligned}$$

as desired.

Q.19 *We wish to construct a 5 qubit error correcting code.*

(a) Show that

$$M_0 = Z_1X_2X_3Z_4, \quad M_1 = Z_0Z_2X_3X_4, \quad M_2 = X_0Z_1Z_3X_4, \quad M_3 = X_0X_1Z_2Z_4$$

are a good set of error syndromes, by showing that they all commute, and that the possible errors will map the $(+1,+1,+1,+1)$ eigenspace to distinct orthogonal subspaces.

(b) Find a basis for the $(+1,+1,+1,+1)$ eigenspace.

(c) Show that for an appropriate choice of encoding, $\bar{Z} = Z_0Z_1Z_2Z_3Z_4$ acts as Pauli Z on the logical qubit, and $\bar{X} = X_0X_1X_2X_3X_4$ acts as Pauli X on the logical qubit.

S.19 (a) In each case, there is a X_i and Z_j in M_a with a corresponding Z_i and X_j in M_b. The two minus signs from the anticommutation of these two operators imply that M_a commutes with M_b. Write $+1$ as 0 and -1 as 1; then the code subspace is 0000. X_0 anticommutes only with the Z_0 in M_1, so it maps to 0010. Similarly X_1 maps to 0101, X_2 maps to 1010, X_3 maps to 0100, X_4 maps to 1001. Z_0 maps to 1100, Z_1 maps to 1000, Z_2 maps to 0001, Z_3 maps to 0011, Z_4 maps to 0110. Y_0 anticommutes with the Z_0 in M_1 and the X_0 in M_2, M_3, so it maps to 1110. Similarly Y_1 maps to 1101, Y_2 maps to 1011, and Y_3 maps to 0111. These are all distinct, so there are good error syndromes.

(b) This can be constructed by starting with two convenient states and projecting to the eigenspace. Let's take

$$|\bar{0}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)|00000\rangle$$

$$|\bar{1}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)|11111\rangle$$

(c) $\bar{Z}$ and $\bar{X}$ commute with all the M_a, so their action on a vector in the code subspace will give a vector in the code subspace. The commutation also implies

$$\bar{Z}|\bar{0}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)\bar{Z}|00000\rangle = |\bar{0}\rangle,$$

$$\bar{Z}|\bar{1}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)\bar{Z}|11111\rangle = -|\bar{1}\rangle,$$

$$\bar{X}|\bar{0}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)\bar{X}|00000\rangle = |\bar{1}\rangle,$$

$$\bar{X}|\bar{1}\rangle = \frac{1}{4}(1+M_0)(1+M_1)(1+M_2)(1+M_3)\bar{X}|11111\rangle = |\bar{0}\rangle$$

as desired.

4 Quantum Algorithms

Q.20 *A general state of an n-qubit system can be written as*

$$|\psi\rangle = \sum_{y=0}^{2^n-1} \psi(y)|y\rangle.$$

Find the condition on $\psi(y)$ for this to be a product state, so that

$$|\psi\rangle = \prod_{i=0}^{n}[a(i)|0\rangle + b(i)|1\rangle]$$

for some functions a, b.

S.20 For a product state, the functions $\psi(y)$ must be a product of functions of the individual bit values, $\psi(y) = \prod_{i=1}^{n} \psi(y_i)$. The functions $\psi(y_i)$ are defined by $\psi(y_i) = a(i)$ if $y_i = 0$ and $\psi(y_i) = b(i)$ if $y_i = 1$. Note that not all functions take this form; to specify a general function $\psi(y)$ we must give 2^n function values, while a product function is determined by only $2n$ values $a(i)$ and $b(i)$.

Q.21 *Consider the Bernstein-Vazirani problem: Given a unitary operator U_f acting on n input bits x and one output bit m such that*

$$U_f|x\rangle|m\rangle = |x\rangle|m \oplus f(x)\rangle,$$

where $f(x) = a \cdot x$, we want to find the value of a. Here $a \cdot x$ is the bitwise multiplication introduced in lectures, with $x \cdot y = x_{n-1}y_{n-1} \oplus \ldots \oplus x_0 y_0$, where $\oplus$ denotes addition mod 2 (or equivalently XOR when acting on single bit values).

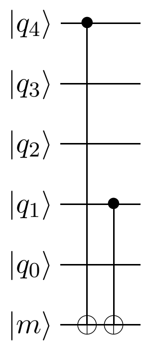
(a) *Describe how to construct a quantum circuit realising U_f for specific values of n and a. Illustrate this explicitly for $n = 5$ and $a = (10010)_2$.*

(b) *Using this quantum circuit and the result of question 5 a), or otherwise, show that*

$$H^{\otimes(n+1)} U_f H^{\otimes(n+1)} |0\rangle_n |1\rangle_1 = |a\rangle_n |1\rangle_1.$$

Hence, using this quantum operation, we can learn the value of a with a single application of U_f.

S.21 (a) U_f adds to m every bit of x where the corresponding bit of a is 1, so a quantum circuit can be constructed by taking a CNOT with control x_i and target m for each bit i with $a_i = 1$. For example, if $n = 5$ and $a = 10010$, the circuit is



(b) Insert an $H^2 = I$ in between each pair of CNOT operations. Then we can use the result of 5 a) to replace the circuit with CNOTs with control $|m\rangle$ and target x_i where $a_i = 1$, giving the indicated result.

Q.22 *Determine the action of U_{FT}^2. Hence show that $U_{FT}^4 = I$.*

S.22

$$U_{FT}|x\rangle = \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{2\pi ixy/2^n} |y\rangle,$$

so

$$\begin{aligned} U_{FT}^2|x\rangle &= \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{2\pi ixy/2^n} U_{FT}|y\rangle \\ &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} e^{2\pi ixy/2^n} \sum_{z=0}^{2^n-1} e^{2\pi iyz/2^n} |z\rangle \\ &= \frac{1}{2^n} \sum_{z=0}^{2^n-1} \sum_{y=0}^{2^{n-1}} e^{2\pi i(x+z)y/2^n} |z\rangle. \end{aligned}$$

The sum over y is a geometric series,

$$\sum_{y=0}^{2^{n-1}} e^{2\pi i(x+z)y/2^n} = \frac{(1 - e^{2\pi i(x+z)})}{(1 - e^{2\pi i(x+z)/2^n})},$$

which gives zero unless $x + z = 0 \bmod 2^n$, in which case the sum is 2^n. So

$$U_{FT}^2|x\rangle = |2^n - x\rangle.$$

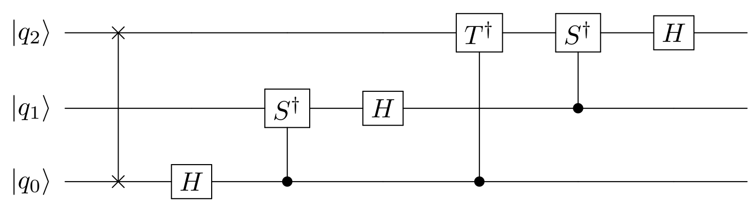
Hence $U_{FT}^4 = I$.

Q.23 *Give the inverse for U_{FT}, and give the explicit quantum circuit for the inverse for three qubits.*

S.23 U_{FT} is a unitary operator, so $U_{FT}^{-1} = U_{FT}^\dagger$, so

$$U_{FT}^{-1}|x\rangle = \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{-2\pi ixy/2^n} |y\rangle.$$

The circuit is given by taking the circuit for U_{FT} given in lectures and conjugating all the operators and reversing the order, giving



Q.24 *Consider the Quantum Fourier Transform, defined as the linear operator U_{FT} on an n qubit Hilbert space whose action on basis states $|x\rangle$, $x = 0, \ldots 2^n - 1$ is*

$$U_{FT}|x\rangle = \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{2\pi i xy/N} |y\rangle \ ,$$

where $N = 2^n$.

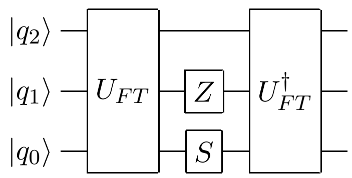
(a) Show that we can rewrite the transform as a product of states for the individual qubits,

$$U_{FT}|x\rangle = \frac{1}{2^{n/2}} \otimes_{l=0}^{n-1} [|0\rangle + \alpha_l |1\rangle],$$

where you should give a formula for the phases α_l.

(b) Show directly (that is, without assuming the unitarity of U_{FT}) that for $x \neq z$, $U_{FT}|x\rangle$ is orthogonal to $U_{FT}|z\rangle$.

(c) Consider a 3-qubit system, and consider the unitary transform $U_{FT}^\dagger S_0 Z_1 U_{FT}$, represented by the quantum circuit below.



Show that this circuit implements the operation $x \to x + 2$ mod 8.

S.24 (a) Writing y as a bit string, $y = y_{n-1}2^{n-1} + \ldots + y_0$, $e^{2\pi i xy/2^n} = \prod_{l=0}^{n-1} e^{2\pi i x y_l/2^{n-l}}$. Thus

$$U_{FT}|x\rangle = \frac{1}{2^{n/2}} \otimes_{l=0}^{n-1} (|0\rangle + e^{2\pi i x/2^{n-l}}|1\rangle).$$

(b) If $|\alpha\rangle = U_{FT}|z\rangle$ and $|\beta\rangle = U_{FT}|x\rangle$,

$$\langle\alpha|\beta\rangle = \frac{1}{2^n} \prod_{l=0}^{n-1} (1 + e^{2\pi i(x-z)/2^{n-l}}).$$

If $x - z$ is odd, that is, if they differ in the least significant bit, the term with $l = n - 1$ vanishes. If $x - z$ is even, but $(x - z)/2$ is odd, the term with $l = n - 2$ vanishes, and so on. So long as they differ in some bit, one of the terms in the product will vanish. Hence the states are orthogonal for $x \neq z$. It is also clear that if $x = z$ we have $\langle\alpha\,|\beta\rangle = 1$ so the states $U_{FT}|x\rangle$ are orthonormal, showing that U_{FT} is unitary.

An alternative derivation is to calculate

$$\langle\alpha\,|\beta\rangle = \frac{1}{2^n} \sum_{u,v} e^{2\pi i(xu - zv)} \langle v\,|u\rangle = \frac{1}{2^n} \sum_{u=0}^{2^n-1} e^{2\pi i(x-z)u/2^n}$$

and note that this is a geometric series. If $x \neq z$ the sum gives

$$\langle\alpha\,|\beta\rangle = \frac{1}{2^n} \frac{1 - e^{2\pi i(x-z)}}{1 - e^{2\pi i(x-z)/2^n}} = 0$$

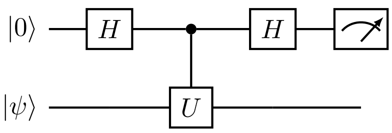
since $x - z$ is an integer. However, if $x = z$ we have

$$\langle\alpha\,|\beta\rangle = \frac{1}{2^n} \sum_{u=0}^{2^n-1} 1 = 1 \ .$$

(c)
$$U_{FT}|x\rangle = \frac{1}{\sqrt{8}}(|0\rangle + e^{i\pi x}|1\rangle)_2(|0\rangle + e^{i\pi x/2}|1\rangle)_1(|0\rangle + e^{i\pi x/4}|1\rangle)_0 \,.$$
Applying S_0 and Z_1 gives
$$\begin{aligned} S_0 Z_1 U_{FT}|x\rangle &= \frac{1}{\sqrt{8}}(|0\rangle + e^{i\pi x}|1\rangle)_2(|0\rangle - e^{i\pi x/2}|1\rangle)_1(|0\rangle + e^{i\pi x/4}e^{i\pi/2}|1\rangle)_0 \\ &= (|0\rangle + e^{i\pi(x+2)}|1\rangle)_2(|0\rangle + e^{i\pi(x+2)/2}|1\rangle)_1(|0\rangle + e^{i\pi(x+2)/4}|1\rangle)_0 \end{aligned}$$
where for qubit 2 we note that $\exp(2\pi i) = 1$, so
$$U_{FT}^\dagger S_0 Z_1 U_{FT}|x\rangle = |x + 2 \pmod 8\rangle$$
with the mod 8 being due to $\exp(2\pi i) = 1$ again.

Q.25 *Suppose we have a unitary operator U on a one-qubit Hilbert space, with an eigenvector $|\psi\rangle$ such that $U|\psi\rangle = e^{2\pi i\varphi}|\psi\rangle$, and we want to find the phase φ.*

(a) Show that if the qubit q_0 is initially set to 0, the measurement



produces a result 0 with probability $p = \cos^2(\pi\varphi)$.

(b) Find the probability for a 0 result when U is replaced by U^k. Hence give a procedure for estimating φ.

S.25 (a) After the first two steps, the state is $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i\varphi}|1\rangle) \otimes |\psi\rangle$, so the further H gives $\frac{1}{2}[(1+e^{2\pi i\varphi})|0\rangle + (1-e^{2\pi i\varphi})|1\rangle] \otimes |\psi\rangle$, so the probability of measuring 0 is as given (and the probability of measuring 1 is $\sin^2(\pi\varphi)$).

(b) If we use U^k, $p(0) = \cos^2(\pi k\varphi)$. We could just measure $\cos^2(\pi\varphi)$ by repeatedly measuring U, obtaining better estimates of $\cos^2(\pi\varphi)$, but the accuracy of the estimate improves slowly. Instead we can speed up the process by using circuits with increasing values of k.

First, note that due to periodicity we can take $\varphi \in (-1/2, 1/2]$ but we cannot distinguish between φ and $-\varphi$ since all the probabilities are even functions of φ. So, up to this ambiguity, let's determine φ with the assumption that $\varphi \in [0, 1/2]$. Note that such values can be written in binary as $\varphi = 0.0b_2b_3b_4\cdots = \sum_{j=2}^{\infty} b_j/2^j$. (The value 1/2 would normally be written in binary as 0.1 but this is also equal to $0.01111\cdots$.)

Now, to determine the value of b_2 we just need to determine of φ is less than 1/4 (so $b_2 = 0$) or not (so $b_2 = 1$). Hence, starting with $k = 1$, we only need to determine p to sufficient accuracy to determine if $p < \cos^2(\pi/4) = 1/2$ or not.

Once we have done that we can set $k = 2$ and measure to estimate the probability $p = \cos^2(\pi 2\varphi)$ which is equivalent to the previous case of $k = 1$ but replacing φ with 2φ. If we had determined $b_2 = 0$ then we have exactly the same process to determine b_3. If instead we had found $b_2 = 1$ we would want to distinguish between $2\varphi \in [1/2, 3/4)$ giving $b_3 = 0$ for $p < 1/2$ and $2\varphi \in [3/4, 1]$ giving $b_3 = 1$ for $p \geq 1/2$.

We can then continue the process by taking $k = 2^2 = 4$ to determine b_4 with the details depending on the value of b_3, as for b_3 and b_2 above. Note that the value of b_2 is irrelevant here since b_2 will affect the integer part of 4φ and that does not matter due to periodicity. In general taking $k = 2^j$ will determine b_{j+2} with the details depending on the value of b_{j+1}.

Q.26 *Find the period of the function $f(a) = y^a \bmod N$ for $N = 713$, for some y of your choosing (if the period is odd, choose again). Use the result to find a prime factor of N.*

S.26 If we take $y = 3$ for example, $r = 330$. $\gcd(3^{165} - 1, 713) = 23$, which gives us $713 = 23 \times 31$.

Q.27 *The diffusion operator is defined by*

$$D = 2|\psi\rangle\langle\psi| - I,$$

where $|\psi\rangle = \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} |y\rangle$ is the uniform superposition of all the computational basis states.

(a) Show that D is a unitary operator.

(b) Show that the action of this operator on an arbitrary state $|\chi\rangle = \sum_x \chi_x |x\rangle$ is

$$D|\chi\rangle = \sum_x (2\bar{\chi} - \chi_x)|x\rangle,$$

where $\bar{\chi} = \frac{1}{2^n} \sum_x \chi_x$ is the average value of the coefficients. It is for this reason that D is also referred to as inversion about the mean.

(c) Construct a quantum circuit to realise this operator.

S.27 (a) Since $D = H^{\otimes n}(2|0\rangle\langle 0| - I)H^{\otimes n}$, it suffices to show that $U = 2|0\rangle\langle 0| - I$ is a unitary operator. But this is immediate, as $U^\dagger = U$ and $U^2 = I$.

(b) $D|\chi\rangle = 2|\psi\rangle\langle\psi|\chi\rangle - |\chi\rangle$, and $\langle\psi|\chi\rangle = \frac{1}{2^{n/2}} \sum_x \chi_x = 2^{n/2}\bar{\chi}$, giving

$$D|\chi\rangle = 2\langle\psi|\chi\rangle \frac{1}{2^{n/2}} \sum_x |x\rangle - |\chi\rangle = \sum_x (2\bar{\chi} - \chi_x)|x\rangle$$

as required.

(c) The unitary $U = 2|0\rangle\langle 0| - I$ is $+1$ on $|0\rangle$ and -1 on all other basis states. We don't care about the overall phase, so we could also take -1 on $|0\rangle$ and $+1$ on all other basis states. Taking the NOT of all bits, this is -1 on $|1\ldots 1\rangle$ and $+1$ on all other basis states. This is $C^{n-1}Z$ acting on one of the qubits conditioned on all the other ones. Thus $D = H^{\otimes n}X^{\otimes n}(C^{n-1}Z)X^{\otimes n}H^{\otimes n}$. The $C^{n-1}Z$ can be reduced to simpler gates using the general procedure for controlled unitaries.

Q.28 *Suppose we have a quantum circuit implementing a unitary operator U such that $U|0\rangle = |\psi\rangle$. Using this, give a circuit implementing the operator*

$$U_\psi = I - 2|\psi\rangle\langle\psi|.$$

S.28

$$U_\psi = U(I - 2|0\rangle\langle 0|)U^\dagger,$$

so using the results of the previous question, $U_\psi = UX^{\otimes n}(C^{n-1}Z)X^{\otimes n}U^\dagger$.

Q.29 *Consider a function $f(x)$, where x is a 3-bit number, which has two values a_1, a_2 such that $f(a_1) = f(a_2) = 1$, and $f(x) = 0$ for all other values.*

(a) The state

$$|\psi\rangle = H^{\otimes 3}|0\rangle = \frac{1}{\sqrt{8}}\sum_{i=0}^{7}|i\rangle$$

can be decomposed into a component $|\psi\rangle_a$ in the subspace $\mathcal{H}_a$ spanned by $|a_1\rangle$, $|a_2\rangle$, and a component $|\psi\rangle_\perp$ in the orthogonal subspace $\mathcal{H}_\perp$. Give explicit expressions for the unit normalised vectors

$$|a\rangle = \frac{|\psi\rangle_a}{\||\psi\rangle_a\|}, \quad |\perp\rangle = \frac{|\psi\rangle_\perp}{\||\psi\rangle_\perp\|}.$$

(b) Given a unitary U_f such that

$$U_f|x\rangle \otimes |m\rangle = |x\rangle \otimes |m \oplus f(x)\rangle,$$

where $|m\rangle$ is the state of a single ancillary qubit, construct an operation V which reflects vectors in the Hilbert space about the subspace $\mathcal{H}_\perp$. That is, if $|\chi\rangle = |\chi\rangle_a + |\chi\rangle_\perp$ with $|\chi\rangle_a \in \mathcal{H}_a$ and $|\chi\rangle_\perp \in \mathcal{H}_\perp$,

$$V|\chi\rangle = -|\chi\rangle_a + |\chi\rangle_\perp.$$

(c) Show that if we have a vector in the two-dimensional subspace spanned by $|a\rangle$ and $|\perp\rangle$, applying V and

$$D = 2|\psi\rangle\langle\psi| - I$$

rotates the state in this subspace, and find the rotation angle.

(d) Give an algorithm to use this rotation to find one of the special values a_1, a_2.

S.29 (a) The component $|\psi\rangle_a$ is just the part of $|\psi\rangle$ in the subspace spanned by $\{|a_1\rangle, |a_2\rangle\}$,

$$|\psi\rangle_a = \frac{1}{\sqrt{8}}(|a_1\rangle + |a_2\rangle).$$

The orthogonal component is then

$$|\psi\rangle_\perp = \frac{1}{\sqrt{8}}\sum_{i\neq a_1,a_2}|i\rangle.$$

Note that by definition $|\psi\rangle = |\psi\rangle_a + |\psi\rangle_\perp$.
Since $\||\psi\rangle_a\|^2 = 1/4$ and $\||\psi\rangle_\perp\|^2 = 3/4$, the unit normalised vectors are

$$|a\rangle = \frac{1}{\sqrt{2}}(|a_1\rangle + |a_2\rangle), \quad |\perp\rangle = \frac{1}{\sqrt{6}}\sum_{i\neq a_1,a_2}|i\rangle.$$

Note

$$|\psi\rangle = \frac{1}{2}|a\rangle + \frac{\sqrt{3}}{2}|\perp\rangle.$$

(b) We take the ancillary qubit in the superposition $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Then

$$U_f|a_i\rangle \otimes |-\rangle = -|a_i\rangle \otimes |-\rangle,$$

while

$$U_f|i\rangle \otimes |-\rangle = |i\rangle \otimes |-\rangle$$

for $i \neq a_1, a_2$. This realises the required operation V since it acts as $-I$ in $\mathcal{H}_a$ while as I in $\mathcal{H}_\perp$:

$$U_f|\chi\rangle \otimes |-\rangle = (-|\chi\rangle_a + |\chi\rangle_\perp) \otimes |-\rangle.$$

Since the state of the ancillary qubit is unchanged, we can think of this as a transformation in the 3-qubit Hilbert space.

Comments: Recalling the description of Grover's algorithm, in that case we had in the 2d subspace $V = 2|\perp\rangle\langle\perp| - I = I - 2|a\rangle\langle a|$. Now the operator V we constructed above also acts in this way in the 2d subspace. However, in the full Hilbert space it is given by $V = I - 2|a_1\rangle\langle a_1| - 2|a_2\rangle\langle a_2|$, not $V = I - 2|a\rangle\langle a|$. Now, it would be fine to construct any operator which reduced to the required form on the 2d subspace, but given U_f our options are limited. Indeed, ignoring the ancillary qubit, we see that U_f is proportional to I when acting in either $\mathcal{H}_a$ or $\mathcal{H}_\perp$ so it cannot act as say $I - 2|a\rangle\langle a|$ which clearly involves a specific state $|a\rangle \in \mathcal{H}_a$ so does not act proportional to I in $\mathcal{H}_a$. However since in $\mathcal{H}_a$ $|a_1\rangle\langle a_1| + |a_2\rangle\langle a_2|$ is the identity operator, we can use U_f is implement $V = I - 2|a_1\rangle\langle a_1| - 2|a_2\rangle\langle a_2|$.

Note also that for a generic state $|\chi\rangle$, $|\chi\rangle_a$ is a linear combination of $|a_1\rangle$ and $|a_2\rangle$ but not proportional to $|a\rangle$. Similarly $|\chi\rangle_\perp$ is generically not proportional to $|\perp\rangle$.

(c) If $|\chi\rangle = \cos\alpha|a\rangle + \sin\alpha|\perp\rangle$, $V|\chi\rangle = -\cos\alpha|a\rangle + \sin\alpha|\perp\rangle$, and

$$\begin{aligned} DV|\chi\rangle &= 2|\psi\rangle(-\frac{1}{2}\cos\alpha + \frac{\sqrt{3}}{2}\sin\alpha) + \cos\alpha|a\rangle - \sin\alpha|\perp\rangle \\ &= \left(\frac{1}{2}\cos\alpha + \frac{\sqrt{3}}{2}\sin\alpha\right)|a\rangle + \left(-\frac{\sqrt{3}}{2}\cos\alpha + \frac{1}{2}\sin\alpha\right)|\perp\rangle \\ &= \cos(\alpha - \pi/3)|a\rangle + \sin(\alpha - \pi/3)|\perp\rangle\ . \end{aligned}$$

This is a rotation through an angle θ with $\cos\theta = 1/2$, that is $\theta = \pi/3$.

Comments: It is not sufficient to just calculate $DV|\psi\rangle$ as that is just a single example so does not show that in general DV acts as a rotation in the 2d subspace.

In general we should also include phases in the coefficients of χ but that doesn't alter anything in this question.

(d) i. Start with $|000\rangle|0\rangle$.

ii. Act with $H^{\otimes 3} \otimes HX$ to produce the state $|\psi\rangle|-\rangle$. This is at an angle of $\pi/3$ to $|a\rangle$ so we have $\alpha = \pi/3$ above.

iii. So, applying DV once, we will obtain precisely $|a\rangle$.

iv. Measuring the state in the computational basis will then give us one of the special values a_1, a_2, each with probability $1/2$, so the probability of a wrong answer is 0 (assuming no errors).

Comments: To find both values a_1 and a_2 we have to repeat the algorithm. Each time we get a random one of a_1 or a_2, so after t tries the probability that we have not found both values is 2^{1-t}.

Note that giving a generic description of Grover's algorithm (or a generalisation of it) does not answer this question. In particular, using estimates which are valid for large $N = 2^n$, where n is the number of qubits, to estimate the number of applications of DV or the probabilities of finding a_1 or a_2 is not sufficient.

Q.30 *Generalise the Grover search algorithm to the case where the function $f(x)$ has more than one value where $f(x) = 1$; that is, to find one of a number of special items. If x has n digits* *and there are r special values, how many times should we apply the Grover iteration? How many searches will it typically take to find all the special values? [You can give estimations with the assumptions $N = 2^n \gg r \geq 1$.]*

S.30 There are now r special values a_i, $i = 1, \dots r$ such that $f(a_i) = 1$, with $f(x) = 0$ otherwise. The $|a_i\rangle$ span an r-dimensional subspace A of the Hilbert space. The uniform superposition $|\psi\rangle$ can be written as

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle = \frac{\sqrt{r}}{\sqrt{N}} |a\rangle + \frac{\sqrt{N-r}}{\sqrt{N}} |a_\perp\rangle,$$

where

$$|a\rangle = \frac{1}{\sqrt{r}} \sum_{i=1}^{r} |a_i\rangle \in A,$$

$$|a_\perp\rangle = \frac{1}{\sqrt{N-r}} \sum_{x \neq a_i} |x\rangle.$$

If we apply the operators D and V as before, they will generate a rotation in the two-dimensional space spanned by $|a\rangle$ and $|a_\perp\rangle$, and the algorithm proceeds in the same way as for a single special item. The only difference is the value of the angle θ between $|\psi\rangle$ and $|a_\perp\rangle$,

$$\cos\theta = \langle\psi|a_\perp\rangle = \frac{\sqrt{N-r}}{\sqrt{N}}.$$

If we assume $r \ll N$, $\theta \approx \frac{\sqrt{r}}{\sqrt{N}}$, so we want to run the algorithm Q times where $(2Q+1)\theta \approx \pi/2$, that is

$$Q \approx \frac{\sqrt{N}}{\sqrt{r}} \frac{\pi}{4} - \frac{1}{2}.$$

After iterating, the state is nearly along $|a_\perp\rangle$. Measuring the state will give at random one of the special values a_i.

For moderate values of r, we will need to take roughly $2r$ to $4r$ samples to typically get one instance of each value. The precise answer to this problem (known as the Coupon collector's problem – how many coupons do you need to collect to get one of each type, assuming equal probability of getting each type?) is $r \sum_{j=1}^{r} 1/j$ and for very large r this is approximately $r \log r$.