QIP710: Assignment 5

Xiuzhe Luo (20812697)

Some properties of Shor's 9-qubit code

1.a

no, because an counterexample is that if there is a Z-error occurs at 2nd qubit, the syndrome is the same as occuring on 1st qubit, e.g if the input state is $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, both single qubit Z-error on 1st qubit and 2nd qubit give the same syndrome 00100100.

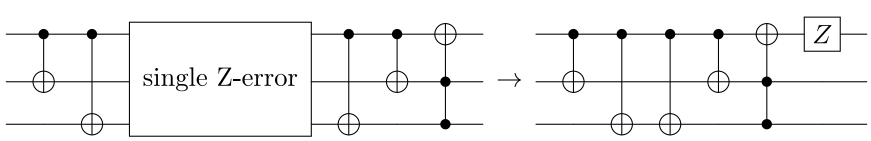
* I just check this by a script, the script is attached in Appendix.

But use the conclusion I prove in the next question, we can also show that the effect of Z-error in the same inner block on its syndrome is always to create a phase/or nothing on the syndrome instead of flip it, so we always get the same bitstring.

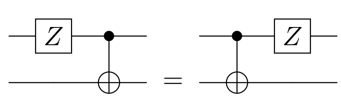
1.b

there can be one X-error occurs in the 3 blocks of the inner circuit, we pick 2 of the 3 which has 3 possibilities, then each of them can happen at 3 possible position for each block, thus $3 \times 3 \times 3 = 27$ weight 2 Pauli-X error in total.

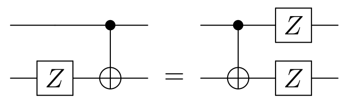
For Z error, it can only occur inside one block, this is because the inner circuiut leave Z error intact and if there is a Z-error it always moves it to the first qubit (the protected qubit), when the ancilla qubits are $|00\rangle$ (note it is not equivalent, I omit other possible Z gate on 2, 3 wires since they don't change the protected qubit, so I'm not using =, but $\rightarrow$)



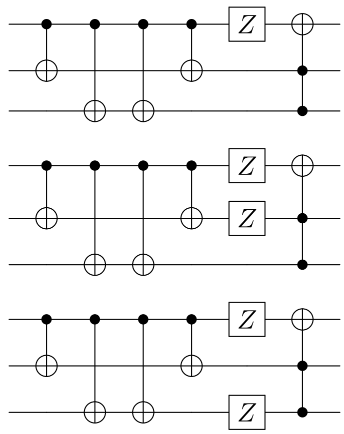
this is a statement from the slide, we can prove this easily by applying the swap rule between Pauli Z and CNOT, note we have



and



thus after we move the error gates from left to right, we always have one Pauli Z on the protected qubit.



when the ancilla qubits are $|00\rangle$, the Toffoli gate has no effect. thus if there are two Z error occurs in the inner block, we get two Z gate on the first qubit, which cancels. But if two Z error occurs at different inner block, then the outer block has to correct 2 Z-error on 1,4,7, which is not possible. Thus we can have $3 \times 3 = 9$ kinds of weight 2 Z errors.

Regarding to Y-errors, it can only happen when both Z and X error could happen, which means there cannot be double Y-error since X cannot be in the same block, while Z has to be in the same block.

Now the ZX-error can happen when single Z-error and single X-error can be corrected, which is anywhere on the qubits as long as they are not on the same qubit (since we are asking for weight 2 errors), thus we have $9 \times 8 = 72$ kinds of ZX-error.

ZY-error can only happen in the same block since single X-error can be corrected, but we cannot have two Z error in different block, thus we have $3 \times 6 = 18$ kinds of ZY-error.

XY-error can only happen in different block since single Z-error can be corrected, but we cannot have two X error in the same block, thus we have $6 \times 3 \times 3 = 54$ kinds of XY-error.

Thus we have $27 + 9 + 72 + 18 + 54 = 180$ kinds of weight 2 error.

1.c

since if Z error occurs at 2,3,5,6,8,9 which cancel itself on each protected qubit, and X error occurs at 1,4,7 which will be corrected by each inner block, the shor code corrects a weight 9 error.

Hadamard transform on uniform superposition of affine linear space

assume G is the $k \times n$ generator matrix for x, for $y \notin C^{\perp}$, when $Gy \neq 0$

$$\sum_{x \in C} (-1)^{x \cdot y} = \sum_{k \in \{0,1\}^k} (-1)^{aGy} = \sum_{k \in \{0,1\}^k} (-1)^{ab} = 0$$

since $y \notin C^{\perp}$, G can be viewed as the partity check matrix for C, thus $Gy \neq 0$. this also indicates that $|C||C^{\perp}| = 2^n$ So we have

$$\begin{aligned}
H^{\otimes n}(\frac{1}{\sqrt{C}} \sum_{x \in C} |x+z\rangle) &= \frac{1}{\sqrt{C}} \sum_{x \in C} H^{\otimes n} |x+z\rangle \\
&= \frac{1}{\sqrt{C}} \frac{1}{2^{n/2}} \sum_{x \in C} \sum_{k \in \{0,1\}^n} (-1)^{x \cdot k} (-1)^{z \cdot k} |k\rangle \\
&= \frac{1}{\sqrt{|C^{\perp}|}} \sum_{k \in C^{\perp}} (-1)^{z \cdot k} |k\rangle + \frac{1}{\sqrt{C}} \frac{1}{2^{n/2}} \sum_{k \notin C^{\perp}} \sum_{x \in C} (-1)^{x \cdot k} |k\rangle \\
&= \frac{1}{\sqrt{|C^{\perp}|}} \sum_{k \in C^{\perp}} (-1)^{z \cdot k} |k\rangle
\end{aligned}$$

Is the transpose a valid quantum operation?

3.a

this means $tr(\rho\rho^T) = 0$, $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ satisfy this condition, the output pure state is $\frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$

3.b, c

quantum channel always maps the density matrix which is a semi-definite matrix to another density matrix which is also a semi-definite matrix. However, if the transposition is a valid quantum channel, there exists a counterexample, let's apply this channel on the first qubit of a Bell state $|00\rangle + |11\rangle$:

$$\Lambda(|00\rangle\langle 00| + |11\rangle\langle 11| + |00\rangle\langle 11| + |11\rangle\langle 00|) = |00\rangle\langle 00| + |11\rangle\langle 11| + |10\rangle\langle 01| + |01\rangle\langle 10|$$

for convenience we omit the normalization factor, the eigenvalue of the output is not all non-negative (by calculating this numerically, there is one negative eigenvalue -1), thus it is not a density matrix, thus there is no such channel for all ρ.

Searching with good guessing algorithm

4.a

assume we sample k times, the probability of getting a satisfying assignment is

$$
\begin{aligned}
P(n) &= \sum_{k=1}^{n} p(1-p)^{k-1} = p\frac{1-(1-p)^n}{p} = 1 - (\frac{3}{4})^n \geq 1-\epsilon \\
\epsilon &\geq (\frac{3}{4})^n \\
\ln(\epsilon) &\geq n \ln\left(\frac{3}{4}\right) \\
n &\geq \frac{\ln(\epsilon)}{\ln\left(\frac{3}{4}\right)}
\end{aligned}
$$

4.b

(note I'm using a different notation of x and y comparing to the question, the x in question is the k here, just for convenience)

denote summation the states satisfy f as $|x\rangle = 2\sum_k A_k |k\rangle, \sum_k |A_k|^2 = 1/4$ and doesn't satisfy f: $|y\rangle = \frac{2}{\sqrt{3}}\sum_k B_k |k\rangle, \sum_k |B_k|^2 = 3/4$, we have

$$
|\omega\rangle = U_g |0^n\rangle = \frac{1}{2}|x\rangle + \frac{\sqrt{3}}{2}|y\rangle
$$

since $|x\rangle$ and $|y\rangle$ are on different basis, they are orthogonal, and we have

$$
\begin{aligned}
U_f |x\rangle |0\rangle &= |x\rangle |1\rangle \\
U_f |x\rangle |1\rangle &= |x\rangle |0\rangle \\
U_f |y\rangle |0\rangle &= |y\rangle |0\rangle \\
U_f |y\rangle |1\rangle &= |y\rangle |1\rangle
\end{aligned}
$$

$$
\begin{aligned}
&U_f(\frac{1}{2}|x\rangle |-\rangle + \frac{\sqrt{3}}{2}|y\rangle |-\rangle) \\
&= -\frac{1}{2}|x\rangle |-\rangle + \frac{\sqrt{3}}{2}|y\rangle |-\rangle
\end{aligned}
$$

Thus U_f reflects the state on $|x\rangle$ axis. Now we create the other reflection operator $2|\omega\rangle\langle\omega| - I = U_g(2|0\rangle\langle 0| - I)U_g^\dagger$, and we have

$$\langle \omega | x \rangle = \frac{1}{2}$$

$$\langle \omega | y \rangle = \frac{\sqrt{3}}{2}$$

$$(2 |\omega\rangle \langle \omega| - I) |x\rangle = |\omega\rangle - |x\rangle = -\frac{1}{2} |x\rangle + \frac{\sqrt{3}}{2} |y\rangle$$

$$(2 |\omega\rangle \langle \omega| - I) |y\rangle = \sqrt{3} |\omega\rangle - |y\rangle = \frac{\sqrt{3}}{2} |x\rangle + \frac{1}{2} |y\rangle$$

thus apply the reflection after oracle we have

$$\begin{aligned}
&(2 |\omega\rangle \langle \omega| - I)(-\frac{1}{2} |x\rangle + \frac{\sqrt{3}}{2} |y\rangle) |-\rangle \\
&= (\frac{1}{4} |x\rangle - \frac{\sqrt{3}}{4} |y\rangle + \frac{3}{4} |x\rangle + \frac{\sqrt{3}}{4} |y\rangle) |-\rangle \\
&= |x\rangle |-\rangle
\end{aligned}$$

which is the answer! The circuit looks like the following

$$\begin{quantikz}
\lstick{$|0^n\rangle$} & \gate{U_g} & \gate[2]{U_f} & \gate{U_g^\dagger} & \gate{2 |0^n\rangle \langle 0^n| - I} & \gate{U_g} & \rstick{$|x\rangle$} \\
\lstick{$|1\rangle$} & \gate{H} & & & & & \rstick{$|-\rangle$}
\end{quantikz}$$

it outputs the answer with only one query without any error.

Characterizing

5.a

assume we have some two qubit state below

$$|\psi\rangle = A_1 |00\rangle + A_2 |01\rangle + A_3 |10\rangle + A_4 |11\rangle$$

on Hadamard basis we have

$$\begin{aligned}
2H \otimes H |\psi\rangle = &(A_1 + A_2 + A_3 + A_4) |00\rangle + \\
&(A_1 - A_2 + A_3 - A_4) |01\rangle + \\
&(A_1 + A_2 - A_3 - A_4) |10\rangle + \\
&(A_1 - A_2 - A_3 + A_4) |11\rangle
\end{aligned}$$

in order to have $a = b$ on both basis, we need $A_2 = A_3 = 0$ and $(A_1 - A_2 + A_3 - A_4) = (A_1 + A_2 - A_3 - A_4) = 0$ thus, $A_1 = A_4 = C$, so the state becomes $C(|00\rangle + |11\rangle)$

5.b

similarly we have

$$2I \otimes H |\psi\rangle = (A_1 + A_2)|00\rangle + (A_1 - A_2)|01\rangle + (A_3 + A_4)|10\rangle + (A_3 - A_4)|11\rangle$$
$$2H \otimes I |\psi\rangle = (A_1 + A_3)|00\rangle + (A_2 + A_4)|01\rangle + (A_1 - A_3)|10\rangle + (A_2 - A_4)|11\rangle$$

thus $A_1 = A_2 = A_3 = C, A_4 = -C$, we get $\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$

You will need the latest develop version of `Yao` to run this, which can be installed with `add Yao#master` in Julia's `pkg` mode.

```julia
using Yao

shor(Es...) =shor(chain(Es...))

shor(E) =chain(9,
     cnot(1, 4),
     cnot(1, 7),

     put(1=>H), put(4=>H), put(7=>H),

     cnot(1,2), cnot(1,3),
     cnot(4,5), cnot(4,6),
     cnot(7,8), cnot(7,9),
     # error
     E,

     cnot(1,3), cnot(1,2),
     cnot((2, 3), 1),

     cnot(4,6), cnot(4,5),
     cnot((5, 6), 4),

     cnot(7,9), cnot(7,8),
     cnot((8, 9), 7),

     put(1=>H), put(4=>H), put(7=>H),
     cnot(1, 7), cnot(1, 4), cnot((4, 7), 1)
     )

# aasignment answers
# 1.a
@vars αβ
r =α*ket"0" +β*ket"1" |>addbits!(8) |>shor(put(1=>Z)) |>partial_tr(1) |>expand
# output: α+ β|00100100>
r =α*ket"0" +β*ket"1" |>addbits!(8) |>shor(put(2=>Z)) |>partial_tr(1) |>expand
# α+ β|00100100>

# 1.c
r =α*ket"0" +β*ket"1" |>addbits!(8) |>
     shor(kron(1=>X, 2=>Z, 3=>Z, 4=>X, 5=>Z, 6=>Z, 7=>X, 8=>Z, 9=>Z)) |>
         partial_tr(2:9) |>expand
# output: α|0> + β|1>

# question 5
@vars A1 A2 A3 A4
psi =A1*ket"00" +A2 *ket"01" +A3 *ket"10" +A4 *ket"11"
# 5.a
copy(psi) |>kron(1=>H, 2=>H) |>expand
# 5.b
copy(psi) |>put(2, 1=>H) |>expand
copy(psi) |>put(2, 2=>H) |>expand
```