

IBMQ를 이용한 9-qubit repetition ECC의 성능 측정

정동윤(고려대학교), 허준(고려대학교)*

milkid@naver.com, *junheo@korea.ac.kr

Performance measurement of 9-qubit repetition ECC using IBMQ

Jeong Dong Yoon, Heo Jun*

Korea Univ., *Korea Univ.

I. 서론

BB84 프로토콜은 1984년 C. H. Bennet과 G. Brassard가 제안한 양자암호통신 프로토콜로, 광자 측정 시 편광신호와 다른 편광 필터를 사용하면 편광 신호가 변질되는 특성을 가진다. 따라서 수신자와 송신자 이외의 비밀키를 가지고 있지 않은 도청자가 편광 신호를 읽으려 할 때 도청자의 편광 필터가 송신된 편광신호와 일치하지 않으면 변질된 편광신호를 통해 도청자의 존재를 알게 된다. 도청자 이외에도 편광 신호가 채널에 의해 확률적으로 변질될 경우도 고려해 본다면 이는 해당 편광 신호를 측정하고 도청자가 있을 threshold, 그렇지 않다면 해당 편광 신호를 어떤 신호인지 판별하는 threshold가 필요로 할 것이다. ECC(Error Correction Code)는 이런 threshold를 제시하는 역할을 하고, 각각의 채널 환경에 따른 error rate를 가지게 될 것이다. 이를 이해하기 위해 본 논문에서는 IBMQ를 이용해서 9-qubit repetition ECC를 구성하고 BB84 프로토콜의 Pauli channel에 따른 성능을 측정할 것이다. 또한 3-qubit repetition ECC와 9-qubit repetition ECC를 결합한 27-qubit repetition ECC를 구성하고 9-qubit repetition ECC와 비교함으로써 얼마만큼의 성능 향상을 하는지 측정할 것이다.

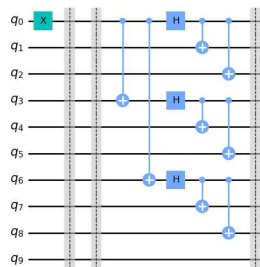
II. 본론

1. 시뮬레이션에 사용할 error channel 및 전제조건

Pauli channel은 single qubit에 대해 $N_p(\rho) = p_0\rho + p_1X\rho X + p_2Y\rho Y + p_3Z\rho Z$ 인 noise를 적용하는 채널로 표현된다. probability distribution $\mathbf{p} = (p_0, p_1, p_2, p_3)$ 는 BB84 channel에서 $\mathbf{p} = ((1-p)^2, p-p^2, p^2, p-p^2)$ 로 표현된다. BB84 프로토콜에서 qubit loss가 일어날 수 있는 채널을 가정하지만, 본 논문에서는 QEC 성능 비교를 위해 qubit loss가 없는 Pauli channel 모델을 적용할 것이다. 또한 각각의 qubit들이 independent 하다고 가정하고, qubit에 동일한 Pauli channel을 적용하여 순차적으로 p 의 값을 변화시켜 정정 이후의 오류율을 확인하고자 한다.

2. 9bit repetition code 회로 구성 및 정정 원리 설명

① initialize 및 encoding

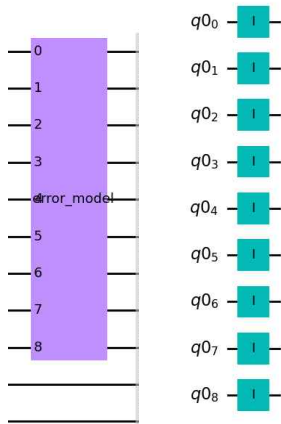


사용자가 보내고자 하는 정보가 1임을 가정하고, q0을 bitflip 시켜 1 state로 만든 후($|1\rangle$), encoding을

통해

$|1\rangle_L = \frac{1}{\sqrt{8}}(|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle)$ 인 9개의 qubit을 보낸다.

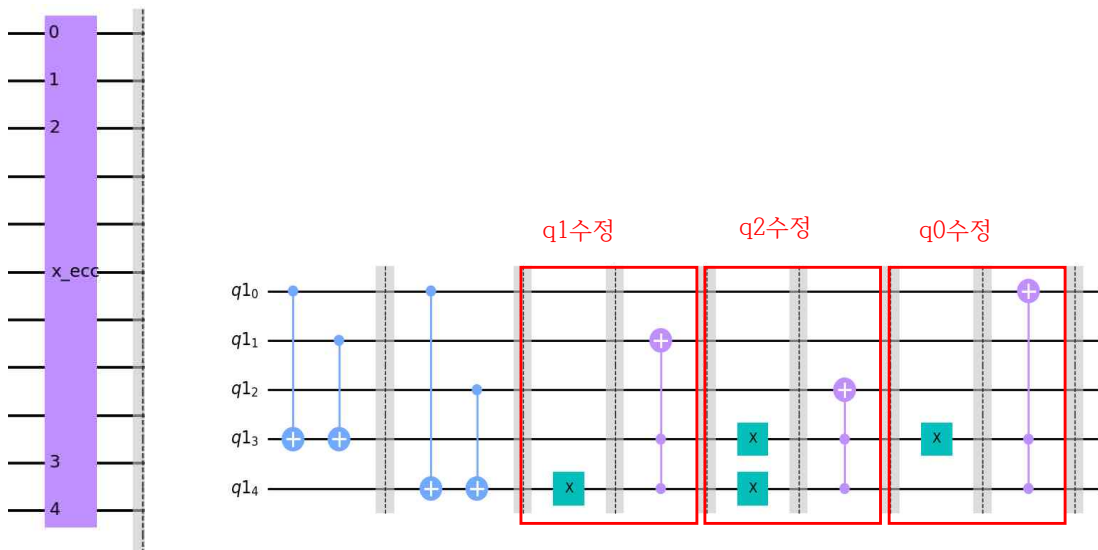
② error channel



Error channel은 error_model 이라는 보라색 상자로 표기하고, Identity gate(id gate)로 구성하였다. id gate를 거치면서, 사전에 설정한 Pauli channel 모델에 따른 에러가 발생한다.

$$\text{ex) } \frac{|000\rangle - |111\rangle}{\sqrt{2}} \otimes \frac{|010\rangle + |101\rangle}{\sqrt{2}} \otimes \frac{|000\rangle - |111\rangle}{\sqrt{2}}$$

③ x_error detecting & correction



X_error detection과 correction이 이뤄지는 회로는 x_ecc라는 보라색 상자로 표기하며, 그 구성은 오른쪽 그림과 같다.

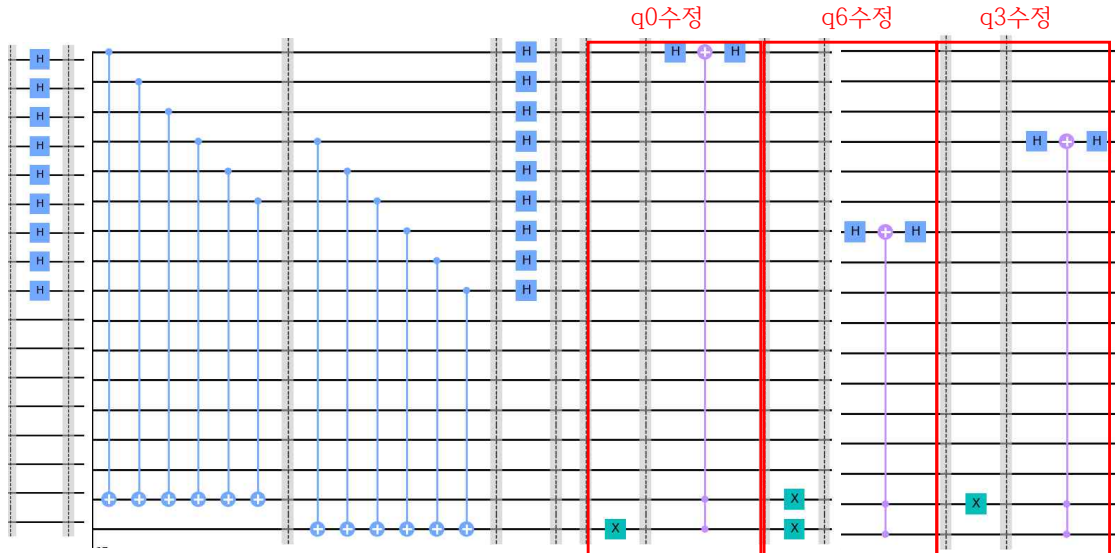
ancilla qubit (에러 발생 위치)	q1 수정	q2 수정	q0 수정
$ 00\rangle$	$ 01\rangle$	$ 10\rangle$	$ 00\rangle$
$ 01\rangle$ (q2)	$ 00\rangle$	$ 11\rangle$	$ 01\rangle$
$ 10\rangle$ (q1)	$ 11\rangle$	$ 00\rangle$	$ 10\rangle$
$ 11\rangle$ (q0)	$ 10\rangle$	$ 01\rangle$	$ 11\rangle$

ancilla qubit인 q3와 q4의 상태는 X_gate를 지나면서 3가지 변화를 확인할 수 있다. ancilla qubit이 가질 수 있는 4가지 상태가 X_gate를 지나면서 $|11\rangle$ 이 되는 경우는 각각의 X-gate를 통한 변화에서 하나씩 가짐을 확인할 수 있다. $|11\rangle$ 인 상태를 가질 때 ancilla qubit이 보여주는 error의 위치에

CCX_gate를 연결함으로써 하나의 qubit에서 발생한 bit_flip error(X_error)을 수정하게 된다. (ex)

$$\frac{|010\rangle + |101\rangle}{\sqrt{2}} \rightarrow \frac{|000\rangle + |111\rangle}{\sqrt{2}}$$

④ z_error detecting & correction

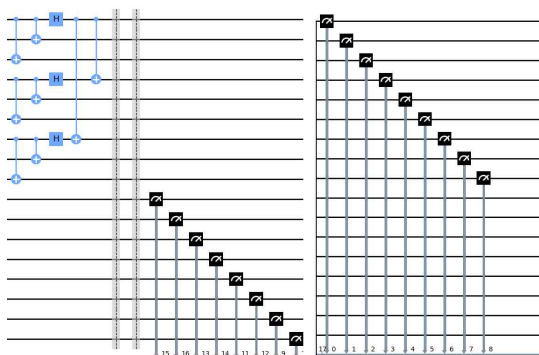


phase_flip error(Z_error)을 detecting 하는 과정은 ancilla qubit에 X-gate가 적용되기 이전이다. H-gate가 적용된 후, 들어오게 되는 state는 q3~q5 중 하나의 qubit에 Z_error가 발생했을 시, $|111\rangle|000\rangle|111\rangle$ 와 같은 state가 들어오게 된다. 이는 ancilla qubit이 $|11\rangle$ 인 경우로 나타난다. 앞서 ③에서와 같은 방식을 적용해서 phase_flip error를 수정하게 된다. 이때 CCZ를 구현하기 위해 $Z = HXH$ 인 관계를 이용하여 CCX-gate의 앞뒤에 H-gate를 연결해 구현하였다.

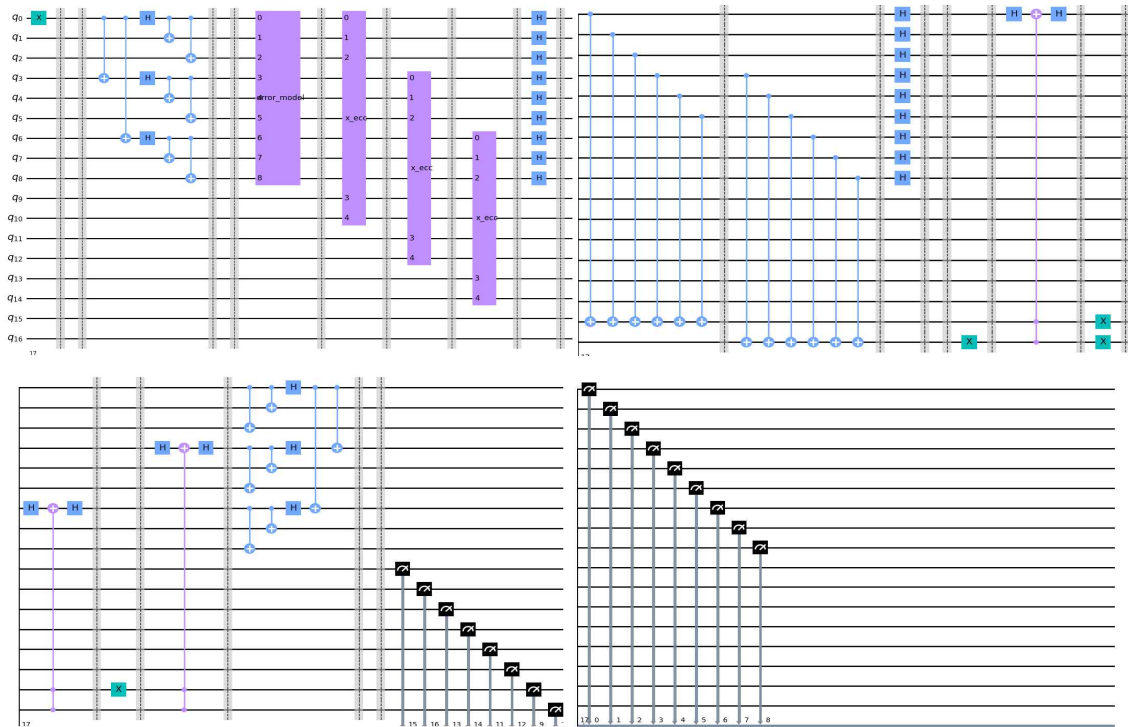
$$\begin{aligned} \text{(ex)} \quad & \frac{|000\rangle - |111\rangle}{\sqrt{2}} \otimes \frac{|000\rangle + |111\rangle}{\sqrt{2}} \otimes \frac{|000\rangle - |111\rangle}{\sqrt{2}} \rightarrow \\ & \frac{|000\rangle - |111\rangle}{\sqrt{2}} \otimes \frac{|000\rangle - |111\rangle}{\sqrt{2}} \otimes \frac{|000\rangle - |111\rangle}{\sqrt{2}} \end{aligned}$$

ancilla qubit(에러 발생 위치)	q0 수정	q6 수정	q3 수정
$ 00\rangle$	$ 01\rangle$	$ 10\rangle$	$ 00\rangle$
$ 01\rangle$ (q6~q8)	$ 00\rangle$	$ 11\rangle$	$ 01\rangle$
$ 10\rangle$ (q0~q2)	$ 11\rangle$	$ 00\rangle$	$ 10\rangle$
$ 11\rangle$ (q3~q5)	$ 10\rangle$	$ 01\rangle$	$ 11\rangle$

⑤ decoding & measurement

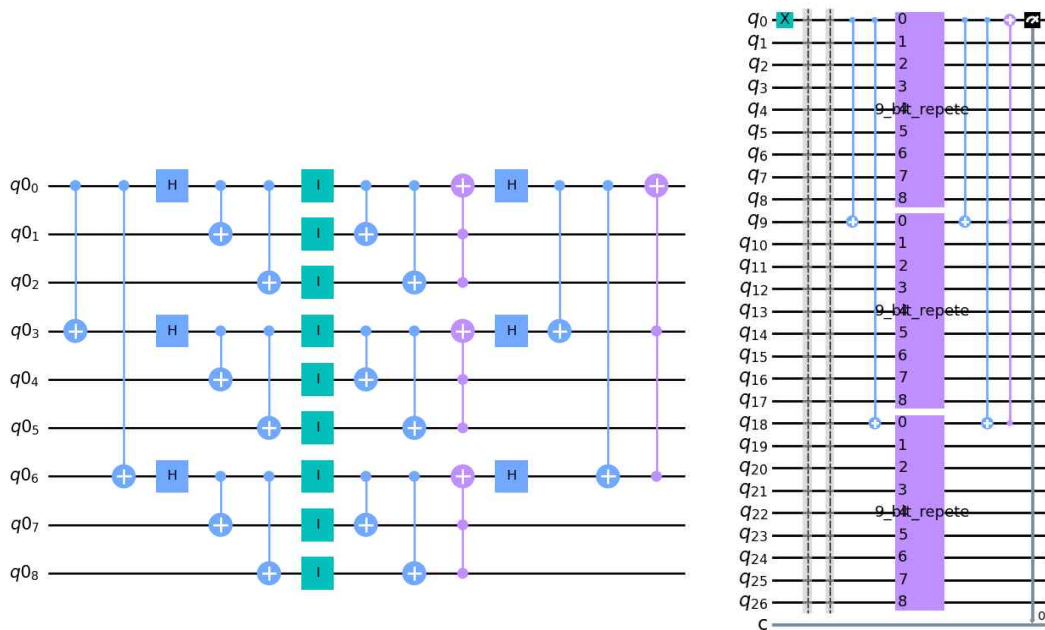


에러가 정상적으로 정정될 경우, decoding을 거치면 encoding 이전 상태인 q_0 만 $|1\rangle$ 이고 ancilla qubit을 제외한 $q_1 \sim q_8$ 은 $|0\rangle$ 을 가지게 된다. ancilla qubit과 9개의 physical qubit을 모두 측정하여 error 정정 위치와 decoder 이후 얻어지는 결과를 시뮬레이션에서 동시에 확인할 것이다.



9-bit repetition ECC 전체 회로도

시뮬레이션에서 사용할 수 있는 최대 qubit의 수는 30개로 제한되어 있다. 따라서 81개 이상의 qubit을 갖는 concatenate 된 circuit의 시뮬레이션은 불가능하다. 이를 대체하여 더 나은 error correction을 하는 circuit을 구성하기 위해 ancilla qubit을 사용하지 않는 9-qubit repetition code와 3-qubit repetition code를 결합하여 27-qubit circuit을 구성하였다.



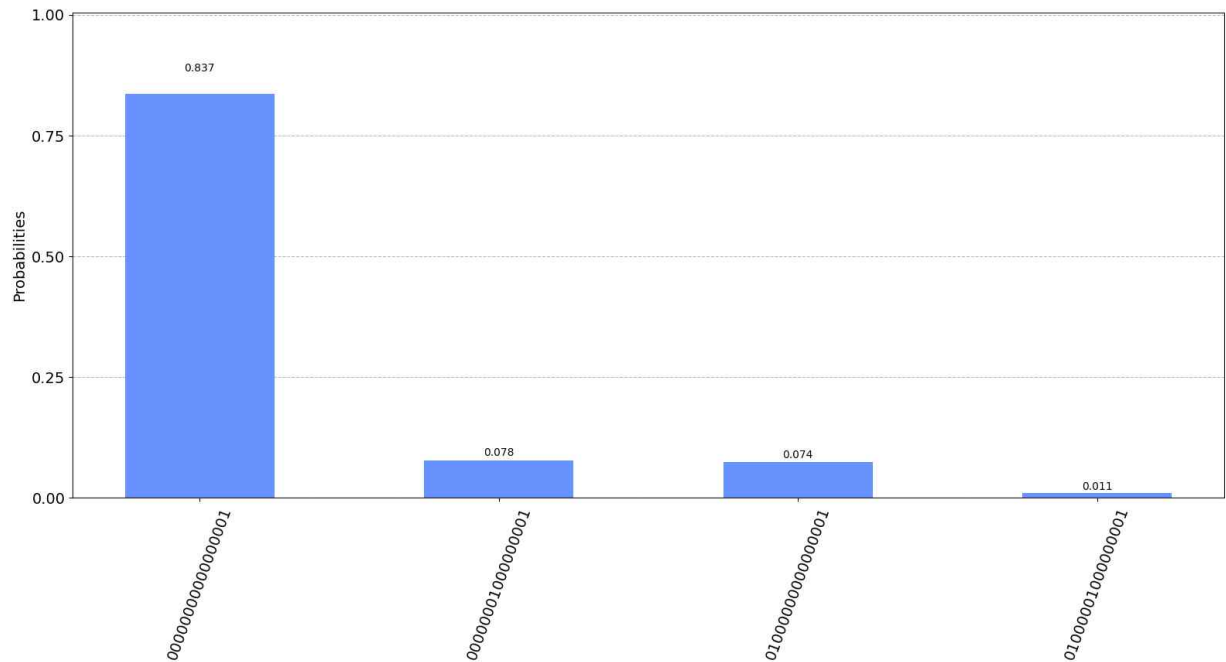
왼쪽에 있는 회로는 9-qubit repetition code의 ECC 회로이다. 이 회로는 q_0 이외의 qubit에 대해 encoding 이전의 state로 돌려주진 않는다. 예를 들어 I_gate(error channel) 이후의 $q_0 \sim q_2$ 의 state가 $\frac{|001\rangle - |110\rangle}{\sqrt{2}}$ 라 할 때, H_gate 이후의 $q_0 \sim q_2$ state는 $|1\rangle|01\rangle$ 이 되어 q_2 를 $|0\rangle$ 로 돌려주진 않는 것을 확인할 수 있다. 따라서 q_0 값에 대해서만 정정이 이뤄진다.

오른쪽에 있는 회로는 3-qubit repetition code를 encoding 하고 각각의 qubit에 대해 9-qubit repetition code의 error correction을 하고, 각각의 q_0 값에 대해 decoding을 진행한 회로이다. 측정은 q_0 에서만 이뤄졌다.

2.시뮬레이션 결과

① 9bit repetition code의 1-qubit에 대한 오류정정확인

$p = 0.1$, $\mathbf{p} = (0.81, 0.09, 0.01, 0.09)$, q_1 에 noise 적용



00_00_00_00_000000001 : error 가 발생하지 않는 경우

00_00_00_01_000000001 : Z_error가 발생한 경우

01_00_00_00_000000001 : X_error가 발생한 경우

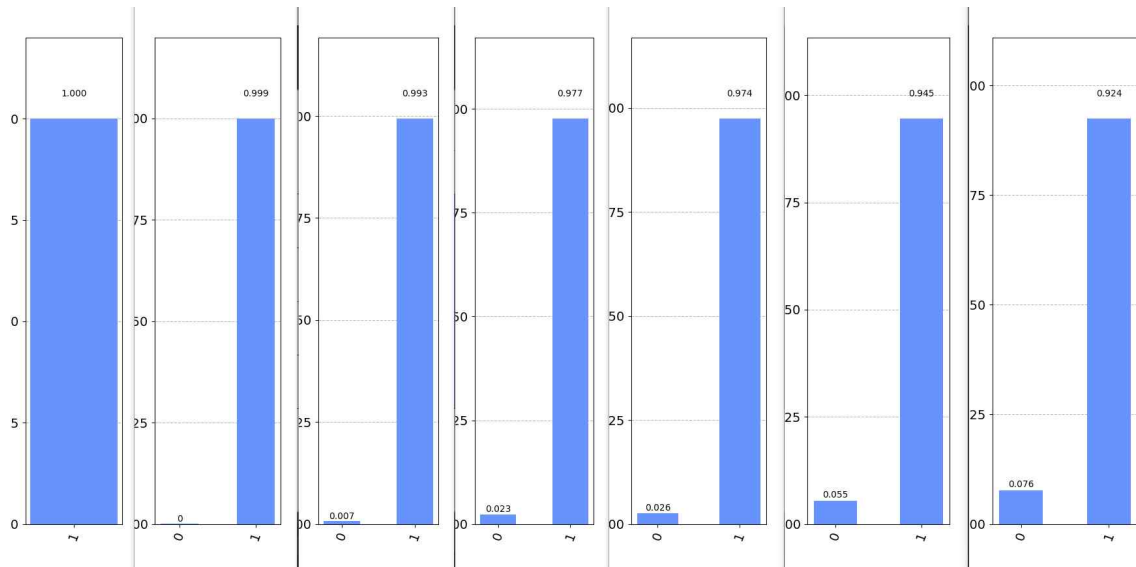
01_00_00_01_000000001 : Y_error가 발생한 경우($Y = -iZX$, X_error와 Z_error가 동시에 발생한 경우)

x_ecc(q10q9)(2qubit)_x_ecc(q12q11)(2qubit)_x_ecc(q14q13)(2qubit)_z_ecc(q16q15)(2qubit)_decoder_output(q8q7q6q5q4q3q2q1q0)(9qubit) 으로 출력된다. 위의 결과는 하나의 qubit에 일어난 error에 대해 정정하는 것과 error 발생 시 정정 위치와 어떤 error가 일어났는지를 보여준다.

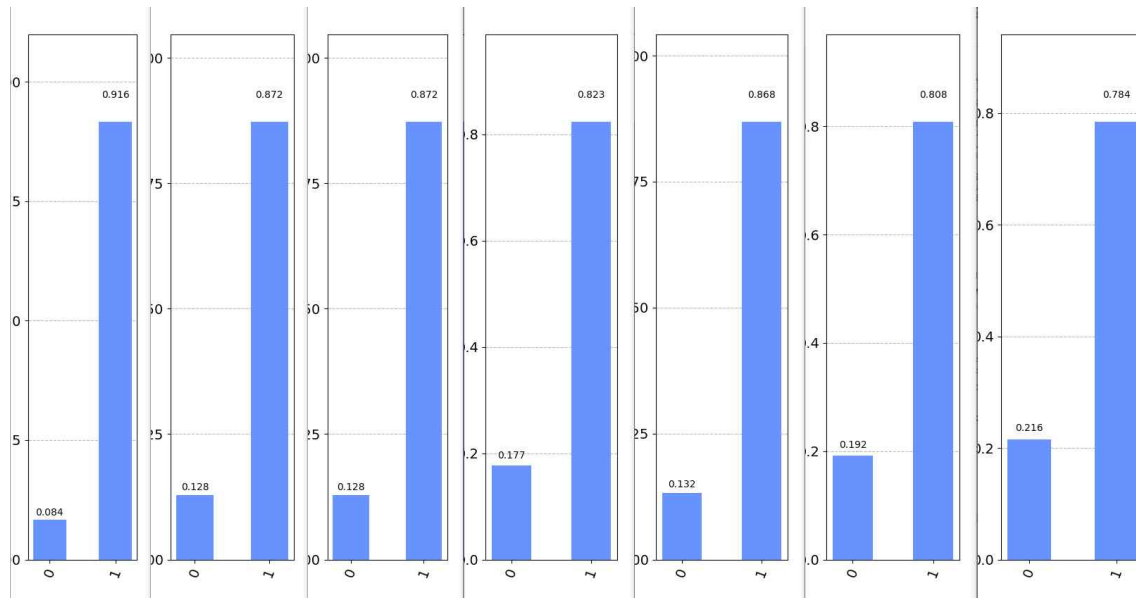
① 9bit repetition code의 p 값에 따른 출력

p = 0, 0.01, 0.02, ... 0.2 까지의 q0 출력

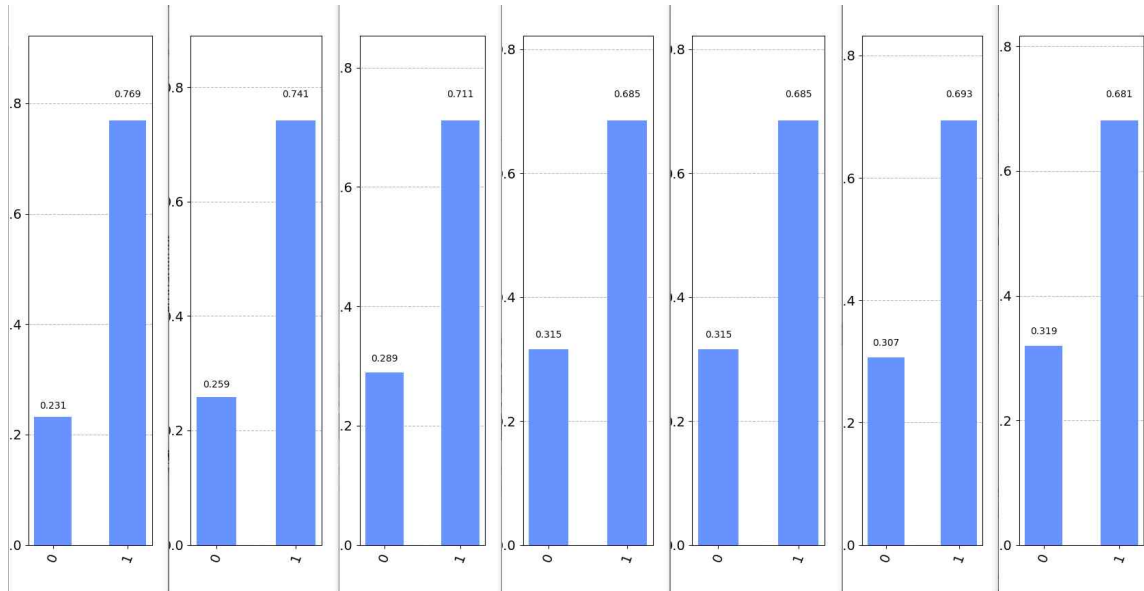
p = 0 ~ 0.06



$p = 0.07 \sim 0.13$

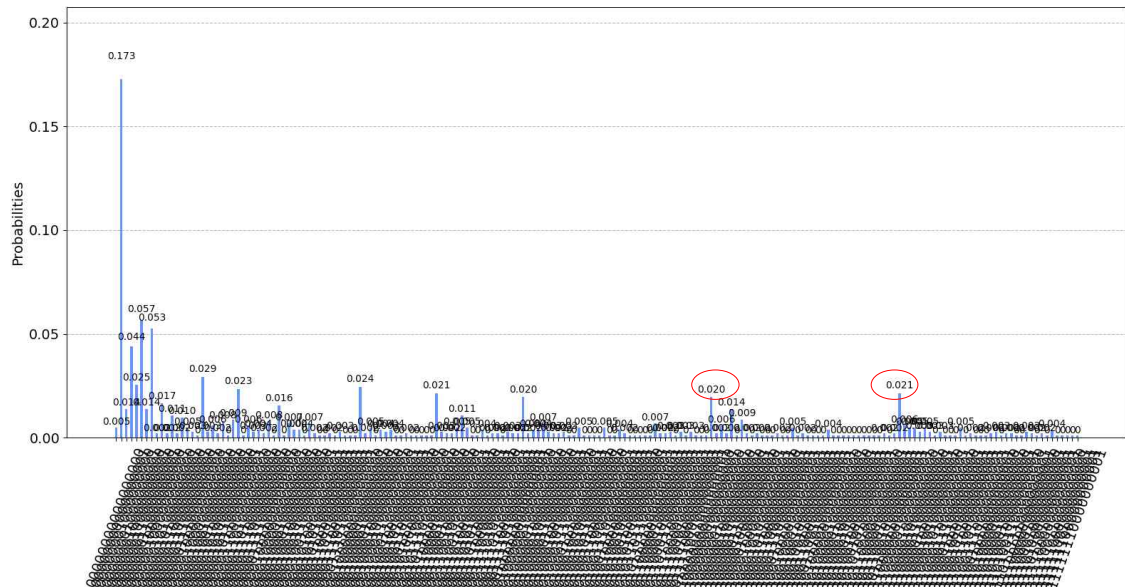


$p = 0.14 \sim 0.20$



오류발생 확률이 거의 $1-(1-p)^2$ 에 비례함을 확인할 수 있다.(ex) $p = 0.1$ 일 때 0.19, $p = 0.2$ 일 때 0.36)

$p = 0.1$, $\mathbf{p} = (0.81, 0.09, 0.01, 0.09)$,

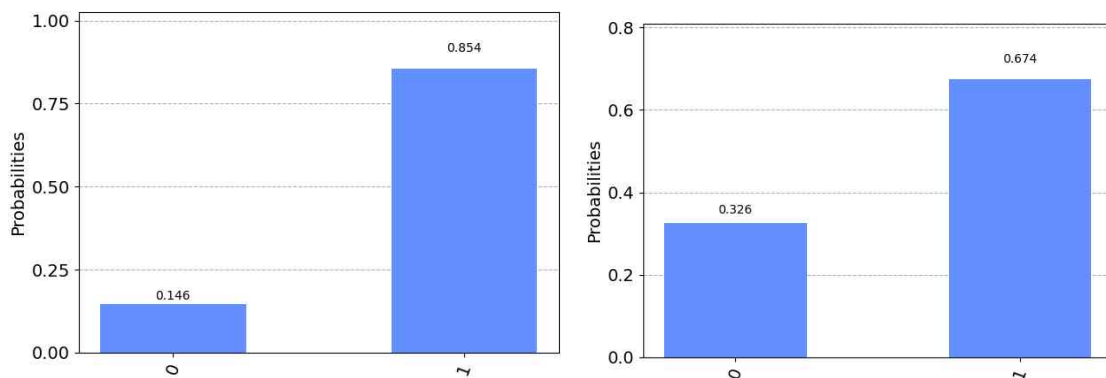


17-qubit에 대한 histogram에서 빨간색으로 표시된 부분과 같이 다른 경우에 비해 비교적 큰 확률을 갖는 패턴에 대해 확인하면 다음과 같다. (1024번 시뮬레이션 한 확률로, 20번 이상인 패턴을 확인하였다.)

'00_00_00_10_000000000': 26,	correction failure, q0~q2, q3~q5 Z_error
'00_11_00_00_000000001': 22,	q3 X_error
'00_00_11_00_000000001': 24,	q6 X_error
'00_00_10_00_000000001': 30,	q8 X_error
'00_10_00_00_000000001': 25,	q5 X_error
'00_00_00_11_000000001': 54,	q3~q5 Z_error

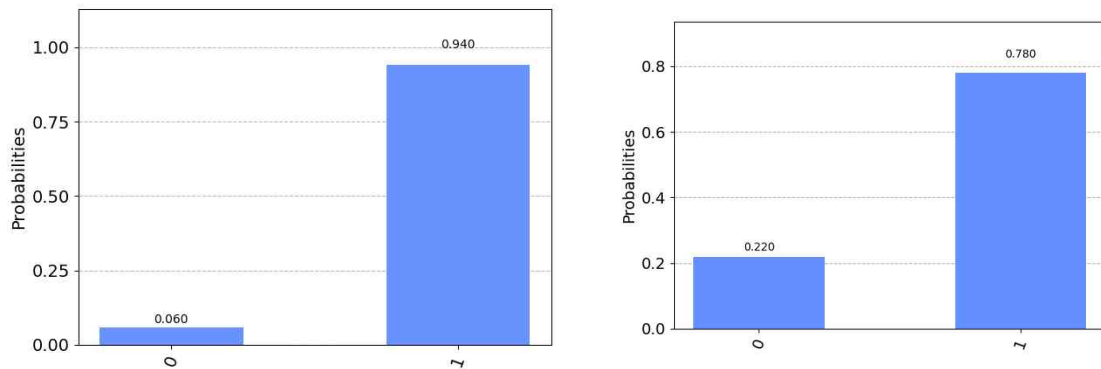
'00_00_00_01_000000001': 27,	q0~q3 Z_error
'10_00_00_11_000000001': 9,	q2 X_error, q3~q5 Z_error
'01_00_00_01_000000001': 8,	q1 Y_error(or q1 X_error (q0, q2) Z_error)
'00_00_00_00_000000001': 33,	no error
'10_00_00_01_000000001': 8,	q2 Y_error(or q2 X_error (q0~q1) Z_error)
'00_10_00_00_000000001': 8,	q5 X_error
'00_00_10_00_000000001': 12,	q8 X_error
'00_11_00_01_000000001': 8,	q3 X_error, q0~q2 Z_error
'01_00_00_10_000000001': 8,	q1 X_error, q6~q8 Z_error
'00_00_00_00_000000000': 8,	correction failure, q0~q2, q3~q5, q6~q8 Z_error
'00_00_01_00_000000001': 8,	q7 X_error
'00_11_00_10_000000001': 8,	q3 X_error, q6~q8 Z_error
'00_01_00_10_000000001': 11,	q4 X_error, q6~q8 Z_error
'00_00_00_11_000000001': 18,	q3~q5 Z_error
'00_10_00_01_000000001': 8,	q5 X_error, q0~q2 Z_error
'00_00_11_00_000000001': 8,	q6 X_error

1-qubit에서 error가 발생하는 패턴과 2-qubit에서 error가 발생하는 패턴이 서로 비슷하게 세어진다. 2-qubit에서 error가 발생하는 비율이 증가하여 정정할 수 없는 패턴의 비율이 늘어나 9-qubit repetition ECC의 성능저하가 확인된다.



9-qubit repetition code의 $p=0.1$, $p=0.2$ 일 때의 error correction 비율 확인.

③ 9-qubit repetition code와 27-qubit repetition code의 error correction 비교



27-qubit repetition code $p = 0.1$

$p = 0.2$

9-qubit repetition code와 비교해 보았을 때, 27-qubit repetition code가 10% 내외로 error correction을 더 잘 수행함을 확인할 수 있다.

III. 결론

BB84의 Pauli channel 모델을 이용하여 error가 있는 channel 모델을 구성하고, 9-qubit repetition code와 함께 9-qubit repetition code와 3-qubit repetition code를 결합한 27-qubit repetition code에 대해 error detect 및 correction 회로를 구현해 보았다. error correction 이후의 error 확률에 대해 p 값이 변화함에 따른 경향성을 확인하고, 9-qubit repetition code와 27-qubit repetition code의 비교를 통해 repetition qubit 수를 늘림에 따른 성능 향상을 확인했다. 하지만 시뮬레이션에서 확인한 것처럼 사용할 수 있는 qubit의 수는 제한되어 있으므로, 실제 채널 조건에 따른 error rate와 decoder 이후의 quantum circuit에서의 error tolerant, 사용 가능한 qubit 자원 등 복합적인 요인을 고려하여 최적화된 ECC를 설계해야 할 것이다.

ACKNOWLEDGMENT

이 성과는 2021년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.2019R1A2C2010061)

참고문헌

[1] Simon J. Devitt, William J. Munro, and Kae Neoto “Quantum Error Correction for Beginners” pp. 8 ~ 11, pp. 20 ~ 21
Johannes Bausch, “Error Thresholds for Arbitrary Pauli Noise”, pp. 8.

[2] Chia_Chun Lin, Amlan Chakrabarti, and Niraj K. Jha, Fellow, IEEE “Optimized Quantum Gate Library for Various Physical Machine Descriptions”, pp. 2063.

https://qiskit.org/documentation/locale/ko_KR/tutorials/circuits_advanced/01_advanced_circuits.html - 합성 게이트

[3] https://qiskit.org/documentation/tutorials/simulators/3_building_noise_models.html -noise model

[4]

https://www.researchgate.net/figure/Shors-nine-qubit-error-correction-code_fig5_334634646
- 27 qubit repetition

[5]

https://ko.wikipedia.org/wiki/%EC%96%91%EC%9E%90_%ED%82%A4_%EB%B6%84%EB%B0%B0 -
서론