

과 IBMQ를 사용하며, Mitiq 파이션 툴킷을 사용하여 양자 컴퓨터에서 ZNE 기법을 통한 Error mitigation을 수행한다. Error mitigation을 적용할 알고리즘은 15큐비트를 사용한 Grover 알고리즘이다.

양자 컴퓨터를 활용한 양자 계산에서 노이즈의 영향을 줄이기 위해 Error mitigation 방법을 적용하고자 한다. 우리는 실험을 위해 Qiskit과 IBMQ를 사용하며, Mitiq 파이션 툴킷을 사용하여 양자 컴퓨터에서 ZNE 기법을 통한 Error mitigation을 수행한다. Error mitigation을 적용할 알고리즘은 15큐비트를 사용한 Grover 알고리즘이다. 노이즈가 없는 환경인 시뮬레이터에서의 Grover 알고리즘 성능과 노이즈가 있는 환경인 양자 컴퓨터에서의 Grover 알고리즘 성능을 비교하며, 더불어 노이즈가 있는 환경인 양자 컴퓨터에서의 Error mitigation인 ZNE를 적용한 Grover 알고리즘 성능을 함께 비교한다.

노이즈가 있는 환경에서 Error mitigation을 적용한 경우(with error mitigation)와 하지 않은 경우(without error mitigation)을 비교하기 위해 동일한 환경에서 Grover 알고리즘 실험을 진행한다. noise가 있는 환경에서 ibmq_executor 함수로 qcircuit 회로 인자를 넘겨주어, job을 실행하고 job.result()를 얻는다. job.result()는 job이 실행된 후 즉시 결과를 반환하며, 결과값 통계를 get_counts() 함수를 통해 얻을 수 있고 이를 통해 현재까지 실행했던 실험의 집계된 결과를 확인할 수 있다. get_counts()는 딕셔너리 형태이며, 딕셔너리에는 이진 형식의 문자열과 회로의 레지스터에 따라 구분된 키를 사용하여 각 큐비트에 대한 카운트가 있다. 예를 들어 result.get_counts(qcircuit)의 결과가 {'0': 498, '1': 502}일 경우, 0과 1은 큐비트의 측정값이고 498과 502는 각 결과가 반복된 횟수이다.

노이즈가 있는 환경에서 Error mitigation을 적용한 Grover 알고리즘을 실행하며, Mitiq의 ZNE를 적용하여 실험한다. Mitiq은 ZNE 방법을 결정하는 Factory를 사용하여 ZNE를 구현하며, 그 중 PolyFactoryMitiq Factory를 사용한다. Factory를 선택한 후, mitiq.zne.zne.execute_with_zne() 함수를 통해 ZNE를 구현한다. Mitiq의 zne.execute_with_zne에 회로 qcircuit과 ibmq_executor 함수, 선택한 factory를 인자로 넘겨주어 job을 실행하여 job.result() 결과를 얻는다.

IV. 실험 결과

세 가지 실험을 각 1000번 수행하여 결과를 도출하였고, 실험 결과를 그래프로 표현한 그림은 그림 2와 같다. x축은 큐비트 측정값이며, y축은 큐비트 측정값이 반복된 횟수이다. 노이즈가 없는 시뮬레이터에서 Grover 알고리즘 실험 결과로 나온 0000, 0111, 1001, 1110 측정값을 제외하고는 오류로 판단한다. 각각의 측정값이 반복된 횟수를 실험마다 비교한다.

A. Noiseless 환경에서 Grover 알고리즘

[('0000', 239), ('0111', 276), ('1001', 234), ('1110', 251)]

B. Noise 환경에서 Grover 알고리즘

[('0000', 53), ('0001', 72), ('0010', 80), ('0011', 67), ('0100', 53), ('0101', 75), ('0110', 64), ('0111', 67), ('1000', 77), ('1001', 73), ('1010', 53), ('1011', 51), ('1100', 49), ('1101', 53), ('1110', 47), ('1111', 66)]

오류가 없는 시뮬레이터와는 다르게 0000, 0111, 1001, 1110 외에 오류로 인해 나올 수 있는 모든 조합이 나온 것을 확인할 수 있다.

C. Noise 환경에서 Error mitigation을 적용한 Grover 알고리즘

[('0000', 71), ('0001', 68), ('0010', 60), ('0011', 84), ('0100', 55),

('0101', 67), ('0110', 66), ('0111', 83), ('1000', 45), ('1001', 64), ('1010', 48), ('1011', 56), ('1100', 41), ('1101', 72), ('1110', 54), ('1111', 66)]

빨간색별은 Error mitigation을 적용하였을 때 올바른 측정값이 검색된 횟수가 증가한 경우를 표시하였으며, 검정별은 해당 측정값이 검색된 경우가 오히려 감소한 경우를 나타낸다. 결과를 통해 올바른 측정값이 나온 경우는 노이즈가 있는 환경에서 Grover 알고리즘 실험 결과, 0000은 53번, 0111은 67번, 1001은 73번, 1110은 47번이다. 노이즈가 있는 환경에서 Error mitigation을 적용한 Grover 알고리즘 실험 결과, 0000은 71번, 0111은 83번, 1001은 64번, 1110은 54번이다. 이에 따라 Error mitigation 적용을 통해 0000은 18번, 0111은 16번, 1110은 7번 증가하였고, 1001은 7번 감소하였음을 알 수 있다. 노이즈가 있는 환경에서 Error mitigation을 적용할 경우, 노이즈의 영향을 덜 받아 측정값이 정확하게 도출되는 결과를 볼 수 있다. 이에 따라 성능 향상을 위한 Error mitigation 적용 효과를 확인할 수 있으나, 오히려 감소한 측정값이 한 개 존재한다.

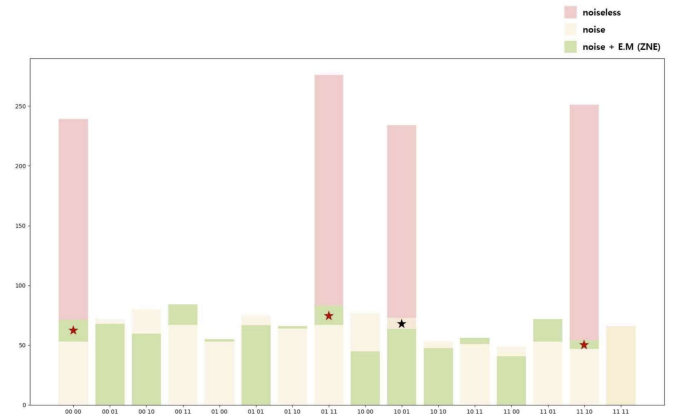


그림 2 Noise와 Error mitigation 여부에 따른 측정값 도출 횟수

V. 결론

양자 컴퓨터의 노이즈 영향을 최소화하기 위한 Error mitigation 연구를 위해 Grover 알고리즘에 노이즈와 Error mitigation 여부에 따른 세 가지 실험을 진행하였다. 노이즈가 있는 환경에서 Error mitigation인 ZNE를 적용할 경우, 올바른 측정값이 도출되는 횟수가 늘어나는 것을 알 수 있었지만 줄어드는 측정값도 존재하였다.

현재 toy example인 Grover 알고리즘에 맞는 Error mitigation을 적용하기 위해 회로에 효율적으로 동작할 수 있는 Error mitigation 기능 구현이 필요하다.

ACKNOWLEDGMENT

본 연구는 2024년도 한국과학기술정보연구원(KIST)의 기본사업으로 수행된 연구입니다. (K-24-L04-C02)

참고 문헌

- [1] LaRose, Ryan, et al. "Mitiq: A software package for error mitigation on noisy quantum computers." *Quantum* 6 (2022): 774.
- [2] Russo, Vincent, et al. "Testing platform-independent quantum error mitigation on noisy quantum computers." *IEEE Transactions on Quantum Engineering* (2023).