

Pure-Error-Interpolated 입력을 활용한 딥러닝 기반 Surface Code 복호 성능 개선

이길한¹, 김지인¹, 오근영¹, 박호성^{1,*}, 박성준²

전남대학교¹, 삼성전자²

gilgilgil0214@naver.com, jlib24555@gmail.com, ohky0529@naver.com, hpark1@jnu.ac.kr,
joonpark2247@gmail.com

Improving Deep-Learning-Based Surface Code Decoding Performance with Pure-Error Interpolated Inputs

Gil Han Lee¹, Jiin Kim¹, Geun-yeong Oh¹, Hosung Park¹, Seong-Joon Park²

Chonnam National Univ.¹, Samsung Electronics²

요약

본 논문은 기존의 $(d+1) \times (d+1)$ 1채널 신드롬 격자 입력을 $(2d+1) \times (2d+1)$ 1채널 격자로 확장하고, 신드롬 위치 사이에 pure error 정보를 주입하는 방식인 Proposed-CNN을 제안한다. 동일한 데이터 생성 및 평가 조건에서 MWPM, FFNN, CNN, Proposed-CNN을 비교하여 논리 오류율(LER)과 pseudothreshold를 분석한다. 제안하는 방식은 단순한 모델 규모 확장이 아니라 측정된 신드롬에 대응하는 pure error의 구조적 단서를 반영함으로써, $d = 5$ 인 surface code의 물리 오류율 p 를 0.07부터 0.13까지 0.01 간격으로 변화시키며 논리 오류율을 측정된 결과 CNN 알고리즘 대비 평균 8.94%의 상대 성능 향상을 보였다.

I. 서론

큐비트는 양자 컴퓨터에서 정보를 저장하는 기본 단위이며, 양자 계산 과정이 멈춰 있거나 어떤 연산이 적용되지 않는 상태에서도 잡음에 취약하다. 따라서 양자 정보를 보호하기 위한 우수한 QEC 코드를 설계하는 것은 양자 컴퓨터를 구축하는 데 있어 중요한 과제이다[1][2].

Rotated surface code의 high-level decoding[3][4]에서는 임의의 오류를 $E = S \cdot T \cdot L$ 로 분해한다. High-level decoding은 이들 중 신드롬으로부터 결정 가능한 pure error인 T 를 분리하고 남은 논리 연산자 $\hat{L} \in \{\bar{I}, \bar{X}, \bar{Y}, \bar{Z}\}$ 를 예측하는 4-class 분류 문제로 환원한다. 여기서 \bar{I} 는 항등 연산자로서 논리적 오류가 발생하지 않았다는 것을 의미한다. \bar{X} 는 비트 반전, \bar{Z} 는 위상반전, \bar{Y} 는 \bar{X} 와 \bar{Z} 가 동시에 이루어짐을 의미하는 논리적 연산자이다. 노이즈 환경이 데이터 큐비트에 미친 영향이 이러한 논리적 연산자 적용과 같으면 논리적 오류가 발생했다는 것을 의미한다.

본 논문은 기존 디코더의 입력 표현을 개선하기 위해, 기존의 $(d+1) \times (d+1)$ 1채널 신드롬 격자 입력을 $(2d+1) \times (2d+1)$ 1채널 격자로 확장하고, 신드롬 위치 사이에 pure error에 해당하는 정보를 주입하는 Proposed-CNN 디코더를 제안한다. 채널 수를 증가시키지 않으면서도, 신드롬과 pure error 단서를 동일한 2차원 격자 상에 배치함으로써 CNN의 국소 컨볼루션 연산이 오류의 국소 연결 구조를 더 직접적으로 학습하도록 유도한다.

본 논문은 MWPM, FFNN, 기존 CNN, Proposed-CNN을 동일한 데이터 생성 및 평가 조건에서 비교한다. 주요 평가 지표는 물리 오류율 p 에 따른 논리 오류율 곡선과 논리 오류율이 물리 오류율과 같아지는 지점을 의미하는 pseudothreshold이다.

격자 확장 자체의 효과와 pure error 주입의 효과를 분리하기 위해, pure error를 제거한 scaled-CNN을 함께 제시하여 Proposed-CNN의 기여를 정량적으로 분석한다.

II. 본론

1. 문제 및 비교군 설정

본 논문은 rotated surface code의 게이트/측정/회로/상태 준비 노이즈를 고려하지 않고, 데이터 큐비트에 독립적인 depolarizing 오류만을 가정하므로 한 번의 신드롬 측정만 이루어지면 된다. 본 논문의 목표는 측정된 신드롬으로부터 논리 연산자 $\hat{L} \in \{\bar{I}, \bar{X}, \bar{Z}, \bar{Y}\}$ 를 예측하는 4-class 분류이다. High-level decoding 관점에서 임의의 오류는 $E = S \cdot T \cdot L$ 로 분해하며, 여기서 S 는 stabilizer, T 는 신드롬 s 로부터 결정적으로 얻을 수 있는 pure error, L 은 논리 연산자이다. 논문[4]의 pure error 산출은 최적 해를 보장하지는 않지만, 신드롬 길이에 선형인 계산량으로 데이터 큐비트에 대한 정보를 구성할 수 있어 제안 모델의 입력으로 사용했다.

비교하고자 하는 디코더들은 다음 네 가지다.

- (i) MWPM: 매칭 알고리즘 기반의 디코더.[5]
- (ii) FFNN: 신드롬을 벡터 형태로 입력받는 다층 퍼셉트론.[3]
- (iii) Baseline-CNN: 단일 채널 격자에 신드롬만 배치하는 CNN.[4]
- (iv) Proposed-CNN: 입력 격자를 확장하고, 신드롬 위치 사이에 pure error 정보를 주입하는 CNN.

2. Baseline-CNN

기존 MWPM과 FFNN 디코더는 격자 구조를 충분히 반영하지 못하며, CNN 모델에서 X/Z 신드롬을 분리해 depolarizing noise에 취약하거나 여러 채널 입력으로 데이터 큐비트 정보를 포함시켜 입력 복잡도를 증가시킨다.

논문[3]에서 제안한 디코딩 방식은 1채널의 입력을 가진다. 이는 데이터 큐비트 이외에 발생하는 오류 상황에서는 여러 번의 측정으로 다채널 입력이 필요하기 때문에 확장성을 고려한 형태이다. 또한 X/Z 신드롬을 분리

하지 않아 X와 Z 사이에 상관관계가 있는 노이즈 모델에서 유리하다.

신드롬의 격자 구조를 그대로 구현하면 불규칙한 형태를 가지게 되기 때문에, 사이에 incoherent value를 채워 넣어 정방구조를 만든다. 이때 ReLU 활성화와 입력 정규화를 위해 격자를 채우기 위한 값 m 은 다음으로 한다.

- (i) incoherent value 위치의 m 은 음수
- (ii) 신드롬 격자 위치에서 오류 감지 여부를 나타내기 위해 m 은 0과 1 사이의 두 값

또한 국소적인 패턴을 분석하기 위한 (2×2) 의 필터 크기와, 좀 더 전역적인 패턴을 분석하기 위한 (3×3) 의 필터 크기인 두 합성곱 층을 구성한다. 필터의 개수 N_f 는 충분한 패턴 식별 능력을 확보하면서 과도한 복잡도를 피하도록 실험을 통해 구한다.

3. Proposed-CNN: Pure-error Interpolated

본 논문은 데이터 큐비트에 대한 정보인 pure error를 포함하면서 복잡도를 줄이는 방법을 제안한다. 제안하는 모델은 1채널의 입력을 가지며, 입력 채널의 크기는 $(2d+1) \times (2d+1)$ 로 한다. (0, 0)부터 (짝수, 짝수) 위치의 격자에 신드롬 정보가 위치하며, (홀수, 홀수)인 d^2 개의 격자 위에 pure error 정보가 위치한다.

- (i) pure error 정보를 주기 위해 X/Z 중 하나의 오류만 발생한 데이터 큐비트 위치의 m 은 0.5로 X/Z 오류가 모두 발생한 경우 m 은 1로, 나머지의 m 은 0이 되도록 한다.

- (ii) 합성곱 층은 pure error 정보와 전역적 패턴 분석을 위한 (5×5) 크기의 필터와 국소적 패턴 분석을 위한 (3×3) 크기의 필터로 구성한다.

$(2d+1)$ 격자 확장은 feature map의 면적 증가로 연산량과 파라미터가 증가한다. 따라서 성능 향상이 단순 입력 해상도 증가 때문인지, 아니면 pure error 정보 삽입으로 인한 구조적 이점인지 분리하기 위해, 본 논문은 pure error 위치에 $m=0$ 을 적용한 scaled-CNN과 Proposed-CNN의 비교 결과를 제시한다.

4. 실험 및 평가

모든 학습 기반 디코더는 depolarizing noise 환경에서 물리 오류율 p 를 0.07부터 0.13 범위에서 선택하여 데이터 큐비트에 오류를 생성했고, 1000만 개(중복 포함)의 샘플로 학습했다.

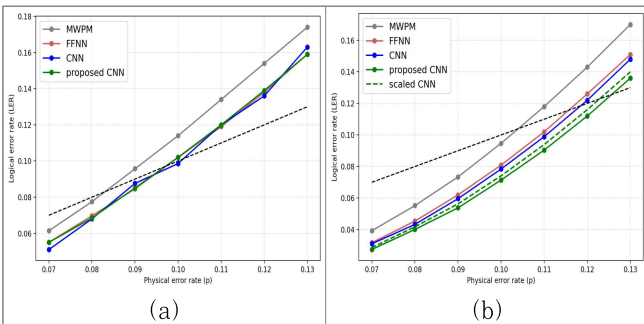


그림 1. Surface code의 physical error rate 대비 logical error rate 비교. ((a) $d=3$ (b) $d=5$)

표 1. 디코딩 알고리즘별 다른 code distance에서 pseudothreshold 비교

d	MWPM	FFNN	CNN	Proposed-CNN
3	0.0830	0.0973	0.1012	0.0971
5	0.1040	0.1157	0.1185	0.1257

그림 1(a)에서 $d=3$ 인 surface code를 대상으로 알고리즘별 성능을 비교한다. 전체 가능한 신드롬 패턴의 개수는 2^8 개로 대부분의 패턴이 학습 데이터에 포함돼 Proposed-CNN은 FFNN, CNN 알고리즘과 비슷한 성능을 보인다. 다만 MWPM은 X/Z 오류의 발생이 독립이라고 가정해 depolarizing noise 환경에서 상대적으로 낮은 성능을 보인다.

그림 1(b)에서 $d=5$ 인 경우에 대해 디코딩 알고리즘의 성능을 비교한다. 제안한 CNN 기반 디코딩 알고리즘이 가장 낮은 논리 오류율을 보이며, 그다음으로 CNN, FFNN, MWPM 순인 것을 알 수 있다.

III. 결론

본 논문은 surface code one-shot 디코딩에서 MWPM, FFNN, CNN, Proposed-CNN을 동일 조건으로 비교하고, 기존 CNN의 입력 표현을 $(d+1)$ 에서 $(2d+1)$ 로 확장하면서 pure error 정보를 interpolation 방식으로 주입하는 Proposed-CNN을 제안하였다. 제안 방법은 신드롬과 pure error 구조 단서를 동일한 2D 격자 상에서 처리하도록 하여 CNN 알고리즘과 같은 입력 채널 수를 가지면서 확장성을 유지한다. 또한 제안 방법은 $d=3$ 에서는 다른 머신러닝 디코더와 비슷한 성능을 보이지만 $d=5$ 에서 CNN 알고리즘 대비 상대 성능이 평균 8.94% 향상되었고, pseudothreshold 또한 증가했다. 향후 연구로는 제안 입력 표현을 유지하면서 초기 downsampling 또는 경량 conv 블록을 도입해 연산량 증가를 완화하는 방향을 고려할 수 있다.

ACKNOWLEDGMENT

본 연구는 2025년도 정부(교육부) 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(RS-2025-25398164) 및 첨단분야 혁신융합대학 사업(차세대통신)의 연구이자, 과학기술정보통신부 및 정보통신기획평가원의 학·석사연계ICT핵심인재양성(IITP-2026-RS-2024-00436837) 지원을 받아 수행된 연구임

참고 문헌

- [1] Steane, A. M., "Simple quantum error-correcting codes," Phys. Rev. A, vol. 54, no. 6, Art. no. 4741, 1996.
- [2] Knill, E., and R. Laflamme, "Theory of quantum error-correcting codes," Phys. Rev. A, vol. 55, no. 2, Art. no. 900, 1997.
- [3] Varsamopoulos, S., B. Criger, and K. Bertels, "Decoding small surface codes with feedforward neural networks," Quantum Science and Technology, vol. 3, no. 1, Art. no. 015004, 2017.
- [4] Jung, H., I. Ali, and J. Ha, "Convolutional neural decoder for surface codes," IEEE Transactions on Quantum Engineering, vol. 5, Art. no. 3102513, Jun. 2024.
- [5] Higgott, O. "Pymatching: A python package for decoding quantum codes with minimum-weight perfect matching," ACM Transactions on Quantum Computing, vol. 3, no. 3, pp. 1-16, 2022.