

논문 2024-61-3-8

양자 컴퓨팅 성능 평가 기술에 대한 비교 분석

(Comparative Analysis of Quantum Computing Performance Analysis Methods)

이 두 원*, 최 병 수**

(Duwon Lee and Byung-Soo Choi[©])

요 약

양자 컴퓨팅은 양자역학적 특성을 이용한 신개념 컴퓨팅을 의미한다. 양자 컴퓨팅과 관련한 연구개발은 크게 활용기술 개발과 구현 기술 개발로 구분되며, 다양한 활용기술과 구현 기술이 제안되고 있다. 현재까지 다양한 구현 기술이 제안되고 있으나, 실제로 비트 컴퓨팅의 한계를 뛰어넘기 위해서는 다양한 기술적 이슈들이 해결되어야 한다. 이 과정에서 양자 컴퓨팅의 성능에 영향을 끼치는 모든 구성 요소들에 대한 기능적, 성능적 요구 수준을 명확히 분석하는 것이 매우 중요하다. 이에 따라, 양자 컴퓨팅 성능 평가 기술 연구개발도 최근 급격한 관심을 받고 있다. 관련하여 이미 다수의 양자 컴퓨팅 성능 평가 기술이 개발되었으나, 양자 컴퓨팅 시스템의 모든 구성요소 중 일부분을 주로 다루고 있거나 혹은 분석 가능 수준이 제한되는 한계점이 지적되고 있다. 이에 양자 컴퓨팅 성능 평가 기술들을 체계적으로 비교 분석하는 것이 중요하다. 본 연구에서는 이를 달성하기 위하여 최초로 양자 컴퓨팅 풀 스택 구조 관점에서 분석하는 기술을 적용한다. 다음으로 양자 컴퓨팅 성능 평가 기술들을 양자 컴퓨팅 계산 성능 지표 범위(계산 시간 등)와 도출 능력(분석 시간 등) 측면에서 체계적으로 비교하였다. 결론적으로 본 연구를 통해 양자 컴퓨팅 성능 평가 기술은 아직까지는 풀 스택 구조를 모두 분석하지 못하고 있으며, 분석 기술 측면에서 해결해야 할 기술적 이슈가 많음을 확인하였다. 이를 통해 최종적으로 향후 지향해야 하는 연구개발 방향성을 제시한다.

Abstract

Quantum computing involves the use of quantum mechanical properties in computing. Research and development in this field are divided into utilization technology and implementation technology, with numerous proposals for both. Various implementation technologies have been proposed, but to overcome the limitations of bit computing, several technical issues must be resolved. Therefore, it is crucial to analyze the functional and performance requirements of all components that affect the performance of quantum computing clearly. As a result, there has been significant interest in the research of quantum computing performance evaluation technology. Several quantum computing performance evaluation technologies have been developed, but their limitations have been identified in that they primarily address only some components, analyze only few performance-metric of quantum computing, and restrict the level of analysis. Therefore, it is crucial to systematically compare and analyze quantum computing performance evaluation technologies. To resolve this issue, we propose a comparative method which is based on the quantum computing full stack structure as a new methodological perspective. We then use it to compare systematically various quantum computing performance evaluation technologies using quantitative indicators for computation time, analysis time, coverage of layers, convenience, etc. This study concludes that performance evaluation technology for quantum computing has not fully analyzed the overall quantum computing full-stack structure yet. We also list that there are various technical issues with respect to analysis technology. The study provides insight into the future direction of research and development that should be pursued.

Keywords : Quantum computing performance, Quantum computing analysis tools, Performance evaluation techniques, Full-stack quantum computing system

*비회원, 국립부경대학교 물리학과(Department of Physics, Pukyong National University)

**정회원, 국립부경대학교 과학컴퓨팅학과(Department of Scientific Computing, Pukyong National University)

[©] Corresponding Author(E-mail : bschoi@pknu.ac.kr)

※ 본 연구는 국립부경대학교 자율창의학술연구비 (2022년)의 지원으로 수행하였습니다.

Received ; November 13, 2023

Revised ; December 27, 2023

Accepted ; January 2, 2024

I. 서 론

양자 컴퓨터는 기존의 비트 컴퓨터와 다르게 중첩과 얽힘이라는 양자역학의 원리를 이용한다. 정보를 중첩하여 동시에 표현하기 때문에 계산 복잡도 측면에서 특정 분야의 문제들을 효율적으로 풀 수 있어서 주목받고 있다^[1]. 1994년 피터 쇼어(Peter Shor)는 양자 컴퓨터를 이용하여 다항 시간 안에 소인수 분해하는 기술을 개발하였다^[2]. 이는 RSA 암호 체계를 무력화하는 것으로, 이후 양자 컴퓨터에 대한 전 세계적 관심을 유도하였다. 1996년 그로버(Lov Grover)는 임의 데이터베이스 탐색 문제에 대하여 기존 비트 컴퓨팅 대비 다항적으로 감소시키는 양자 알고리즘을 제안하였다^[3]. 해당 알고리즘은 보편적으로 적용 가능한 알고리즘으로써 이후 양자 컴퓨터가 다양한 분야에 적용되는 시초가 되었다.

양자컴퓨터의 잠재적 가능성이 증명됨에 따라, 이후에는 양자 컴퓨터 시스템의 개발에 많은 발전이 진행되었다. 지난 20여년간의 꾸준한 발전을 통해서 양자 컴퓨터를 구성하는 큐비트의 개수는 지속적으로 증가하고 있으며, 양자 게이트의 정확도도 지속적으로 개선되고 있다. IBM은 127 큐비트를 갖는 “Eagle” 프로세서를 장착한 초전도 기반 양자 컴퓨팅 시스템을 개발하였다^[4]. Google은 양자 컴퓨터의 우수성을 증명하는 실험으로 비트 컴퓨팅에서 10,000년 걸리는 문제를 200초에 해결하는 양자 계산 실험을 자체 양자 컴퓨팅 시스템을 통해서 입증하였다^[5]. 하드웨어적 측면과 더불어 소프트웨어 측면에서도 많은 발전이 진행되었다. 현재 가장 많이 사용되는 QISKIT 양자 컴퓨팅 프로그래밍 환경이 IBM에 의해서 개발되었다^[6]. 현재 IBM은 이러한 양자 컴퓨팅 프로그래밍 환경을 기반으로 클라우드 환경에서 양자 컴퓨팅 사용자 경험을 제공하고 있다. 비슷한 방식으로 Google에서는 Cirq^[7] 환경을 제공하고 있다. 범용적인 양자 컴퓨팅 환경과 별개로 양자 컴퓨팅 구성 요소들에 대한 정교한 시뮬레이션을 제공하는 QuTiP^[8, 9] 등이 개발되었다. 최근에는 이러한 양자 컴퓨팅을 이용하여 양자 인공지능 분야에 적용하기 위한 프로그래밍 환경으로 Tensor Flow Quantum^[10]이 개발되었다. 이와 같이 양자 컴퓨팅 시스템 개발과 관련하여 다양한 활용 기술과 구현 기술들이 급속히 발전하고 있다. 이러한 활용 및 개발 기술과는 별도로 최근에는 양자 컴퓨팅 시스템 성능 평가 기술이 주목받고 있다. 양자 컴퓨팅 성능 평가 기술은 현재 만들어진 양자 컴퓨팅 시스템의 제공 가능한 양자 계산 능력을 분석하는데 사용

되거나, 주어진 작업을 수행하기에 요구되는 양자 컴퓨팅 시스템 내부 구조 및 기술 사양을 결정하는데 사용된다. 특히 이러한 양자 컴퓨팅 성능 평가 기술은 양자 컴퓨팅 시스템의 개발 크기가 커지거나 혹은 활용 요구 수준이 높아지면서 그 중요성이 증가하고 있다.

이러한 이유로 현재까지 다양한 양자 컴퓨팅 성능 평가 기술이 개발되었다. 앞서 언급하였던 QuTip의 경우에도 큐비트와 양자 게이트의 성능을 평가하는데 활용된다. 현재까지 다양한 양자 컴퓨팅 성능 평가 기술이 개발되었으나, 양자 컴퓨팅 시스템 사용자와 개발자 관점에서 이러한 기술들에 대한 체계적인 비교 분석은 거의 없는 상황이다. 결과적으로 현재 다양한 양자 컴퓨팅 성능 평가 기술이 제안되기는 하지만, 양자 컴퓨팅 시스템 사용자와 개발자는 적절한 평가 기술을 사용하지 못하는 한계가 있다. 이는 결과적으로 양자 컴퓨팅 시스템 도입과정 및 개발과정에서 잘못된 판단과 기술 개발이 될 수 있다. 이에 본 연구에서는 이러한 문제점을 해결하고자, 기존에 제안되었던 다양한 양자 컴퓨팅 기술을 체계적으로 분석하고, 이를 통해 각 기술들에 대한 동일 지표 관점에서의 비교 분석을 시도한다.

이러한 비교 과정에서 일관된 규칙을 적용하기 위하여 본 연구에서는 양자 컴퓨팅 시스템 풀 스택 구조를 고려한 비교 분석이 최초로 시도된다. 양자 컴퓨팅 시스템 풀 스택 구조는 현재 시점에서 매우 제한적 수준에서 고려되고 있으나, 비트 컴퓨팅 분야에서는 범용적으로 사용되는 고려 사항인 만큼, 양자 컴퓨팅 시스템에서도 적용되어야 한다. 그러나 현재까지 이러한 수준에서의 비교 분석은 없었다. 이에 본 연구에서는 양자 컴퓨팅 시스템이 추구하게 될 풀 스택 구조를 상세히 고찰하고, 이를 중심으로 하여 다양한 양자 컴퓨팅 성능 평가 기술을 일관된 측면에서 비교 평가하는 기술을 제시한다.

우선 양자 컴퓨팅 성능 지표를 중심으로 개별 성능 평가 기술들이 어떠한 성능 지표(계산 시간, 정확도, 자원 요구량, 시스템 크기 등)를 중심으로 개발되었는지를 분석한다. 현재까지는 표준화된 성능 지표가 없는 상황에서 다양한 성능 평가 기술들이 서로 다른 지표를 사용하기 때문에 양자 컴퓨팅 시스템 사용자 및 개발자 관점에서 하나의 단일화된 평가 기술 개발이 시급히 요구됨을 확인하였다.

다음으로는 성능 평가 기술을 실제 사용하는 과정에서 요구되는 성능 지표(분석 시간, 확장성, 편의성, 정교 수준 등)를 기준으로 분석하였다. 그 결과 양자 컴퓨

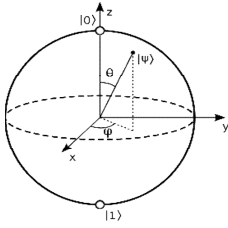


그림 1. 양자 정보의 기하학적 표현

Fig. 1. Geographical view of qubit (bloch sphere diagram).

팅 시스템의 풀 스택 구조를 최대한 고려한 성능 분석의 경우에는 분석 시간이 매우 크게 증가하는 현상이 대부분의 평가 기술에서 발견되었다. 이를 통해서, 향후에는 양자 컴퓨팅 성능 평가 기술의 구동 시간을 감소시키는 연구가 매우 중요함을 확인하였다. 본 논문은 다음과 같이 구성된다. II장은 양자 컴퓨팅의 개요를 설명한다. III장에서는 본 연구에서 고려되는 양자 컴퓨팅 시스템 풀 스택 구조를 자세히 기술한다. 현재까지 제안된 양자 컴퓨팅 성능 평가 기술들은 IV장에서 설명된다. 본 연구의 핵심인 양자 컴퓨팅 성능 평가 기술들에 대한 1) 도출 가능한 양자 컴퓨팅 성능 지표 중심 비교 평가와 2) 동작 과정에서의 성능 지표 중심 비교 평가를 각각 V장과 VI장에서 기술한다. 마지막으로 VII장은 본 연구를 결론지으며 향후 연구를 기술한다.

II. 양자 컴퓨팅 개요

본 연구에서는 양자 컴퓨팅과 관련한 양자 정보, 양자 게이트, 양자 회로, 양자 알고리즘을 간략한 수준에서 다음과 같이 요약 정리하여 설명한다. 자세한 내용은 문헌^[1]을 참고하기 바란다.

양자 정보(Qubit)은 양자 역학적 특성을 갖는 정보를 의미하며 다음과 같이 벡터로 기술된다.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

이러한 양자 정보는 0과 1의 중첩상태를 표현할 수 있으며, 위상 정보 값들 간에 간섭현상에 의해서 상태 확률의 증감이 발생한다. 단일 큐비트들을 여러 개 사용하여 지수적 상태공간을 다항적 큐비트로 표현 가능하기 때문에 매우 높은 수준의 정보 표현 능력을 갖는다. 이러한 큐비트는 초전도^[4], 이온 트랩^[11], 광자^[12], 양자점^[13] 등 다양한 물리계를 이용하여 구현할 수 있다. 그림 1은 이러한 큐비트를 3차원 공간에서 벡터 형태로 표현한 것이다. 이에 따라, 이론적으로는 하나의 큐비트에서도 매우 많은 상태공간을 표현할 수 있기 때

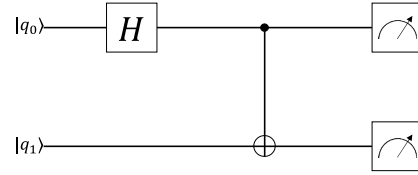


그림 2. 양자 회로 모식도 (벨 상태 생성용)

Fig. 2. Quantum circuit example (bell state generation circuit).

문에, 양자 연산 과정에서는 하나의 큐비트를 통해서 다수의 계산 결과를 동시에 표현할 수 있다. 그러나, 실제 관측 과정을 통하면 0과 1만 관측되기 때문에, 연산 후 관측시에는 1개의 비트 정보로만 관측된다.

양자 게이트는 이러한 양자 정보를 처리하는 것으로써 크게 I, X, Y, Z, H, T, CNOT 게이트 등이 있다. 해당 게이트들은 큐비트의 벡터 표현에 따라 다음과 같이 행렬로 정의된다.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (2)$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad (3)$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (4)$$

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (5)$$

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (6)$$

양자 정보에 대한 일련의 양자 게이트의 적용을 양자 회로로 하며, 그림 2와 같이 양자 회로는 기존 비트 회로와 유사하게 구성되어 있다.

이러한 양자 회로를 구동하여 양자 정보를 표현, 처리, 관측함으로써 양자 컴퓨터가 구동된다.

양자 알고리즘은 이러한 일련의 과정을 수행하는 논리 흐름으로써 주로 양자 정보를 표현하고, 양자 게이트를 적용하는 것으로 구성되어 있으나, 이러한 양자 회로를 순차적, 병렬적으로 구동하기 위한 고전적 알고리즘과 함께 구성된다.

양자 컴퓨팅 시스템은 이와 같이 양자 정보를 표현하고, 양자 게이트를 구동하여 양자 회로를 동작시키는 일련의 과정을 수행하고, 사용자로 하여금 문제 해결을 위해서 필요한 프로그래밍과 결과 확인이 가능한 인터페이스를 포함한다. 사용자 관점에서 살펴보면, 자신이 작성한 양자 알고리즘을 자동화된 분해 및 합성 과정^[14]을 거쳐서 양자 하드웨어를 제어하는 일련의 과정을 의미한다.

III. 양자 컴퓨팅 시스템 풀 스택 구조

앞서 개념적으로 설명되었던 양자 컴퓨팅의 사용자 수준부터 양자 하드웨어 수준까지의 일련의 과정을 모두 포함하는 경우, 이를 양자 컴퓨팅 시스템 풀 스택 구조라 한다. 그림 3은 이에 대한 개념을 도식화한 것이다. 양자 컴퓨팅의 연구개발과 성능 평가 과정에서는 이러한 양자 컴퓨팅 풀 스택 구조를 고려하는 것이 매우 중요하기 때문에, 본 연구에서는 이러한 풀 스택 구조를 다음과 같이 개별 Layer 단위로 분류하여 접근한다. 본 연구에서는 현재 가장 많이 사용되고 있는 초전도 기반 양자 컴퓨팅 시스템의 풀 스택 구조를 중심으로 설명한다. 그러나, 다른 양자 하드웨어의 경우에도 기본적으로는 계층 구조적 접근이 동일하기 때문에 설명된 방식이 그대로 적용 가능하다.

1. 양자 프로그래밍 계층(Layer 1)

- **주요 기능:** 사용자가 알고리즘을 프로그래밍 언어로 기술하는 과정. 양자 계산 결과를 확인해 주는 기능
- **사용자 입력 정보:** 양자 프로그램
- **주요 구성 요소:** 프로그래밍 언어, 인터페이스 화면, GUI 등
- **필요 내부 정보:** 양자 게이트 조합, 회로 라이브러리, 프로그래밍 언어 정보 (예: Python + Quantum Circuit)

2. 양자 회로 분해 계층(Layer 2)

- **주요 기능:** 양자 프로그램을 양자 하드웨어 기본 언어 수준으로 분해 (예: 양자 어셈블리 코드^[15, 16])
- **사용자 입력 정보:** 회로 분해 근사 수준
- **주요 구성 요소:** 다중 큐비트 분해기, 단일 큐비트 분해기, 비트 회로의 양자 회로 변환기, 고전 제어부 처리기
- **필요 내부 정보:** 하드웨어 수준 기본 게이트들, 회로 변환 알고리즘들

3. 양자 회로 합성 계층(Layer 3)

- **주요 기능:** 사용자가 알고리즘을 프로그래밍 언어로 기술하는 과정. 양자 계산 결과를 확인해 주는 기능
- **사용자 입력 정보:** 최대 허용 합성 시간
- **주요 구성 요소:** 시스템 합성기, 회로 합성기
- **필요 내부 정보:** 컴퓨터 구조, 큐비트 배치 구조, 큐비트 간 연결 구조

4. 결합 허용 양자 계산 계층(Layer 4)

- **주요 기능:** 고신뢰 대용량 양자 컴퓨팅을 위한 결합 허용 양자 컴퓨팅 프로토콜 적용 작업*
- **사용자 입력 정보:** 목표 신뢰도 수준
- **주요 구성 요소:** 결합 허용 회로 합성기
- **필요 내부 정보:** 양자 하드웨어 기본 성능 정보 (예: 양자 게이트별 동작시간, 에러율 등), 결합허용 양자 컴퓨팅 프로토콜

5. 제어 신호 생성 및 처리 계층(Layer 5)

- **주요 기능:** 양자 하드웨어 기본 단위 프로그램에 따라 양자 하드웨어 제어 신호 생성 및 처리
- **사용자 입력 정보:** 신호 생성 기술, 제어 기술
- **주요 구성 요소:** 운영체제, 양자 컴퓨팅 제어용 PC, 신호 생성기, 신호 분석기
- **필요 내부 정보:** 주요 전자 장비 내부 제어 명령어 및 프로그램

6. 양자 상태 유지 계층(Layer 6)

- **주요 기능:** 양자 상태 유지 기능
- **사용자 입력 정보:** 장치 정보
- **주요 구성 요소:** 냉각기, 진공챔버
- **필요 내부 정보:** 열적 유지 지원 수준, 진공 유지 시간

7. 큐비트 계층(Layer 7)

- **주요 기능:** 양자 상태 제어 및 측정 기능
- **사용자 입력 정보:** 양자칩의 설계/목표 수준 정보 (큐비트 수, 큐비트 연결 구조도, 게이트 유형, 게이트별 정확도 등)
- **주요 구성 요소:** 양자칩, 제어 신호선, 신호 전달선
- **필요 내부 정보:** 양자칩 기본 정보

* 양자 컴퓨팅은 노이즈에 취약하여 양자 계산에서 오류가 발생할 수 있다. 큐비트가 주변 환경과 상호 작용을 통하여 결맞음을 잃을 수 있고 큐비트끼리 상호 작용할 수 있으며 제어 과정에서도 노이즈로 인해 게이트의 충실도가 떨어질 수 있다^[17, 18]. 비트 컴퓨터에 Hamming code와 같은 오류 보정 코드가 있듯이 Steane code, Surface code 등의 양자 오류 보정 코드가 있다^[19~21]. 이러한 오류 보정 코드가 적용되어 잡음이 있어도 정확한 양자 계산을 수행할 수 있는 시스템을 결합허용 양자 컴퓨팅(Fault Tolerant Quantum Computing) 시스템이라고 한다^[22]. 하지만 양자 컴퓨팅 시스템의 하부 스택에 오류가 발생하면 오류 보정 코드에도 오류가 발생할 수 있다.

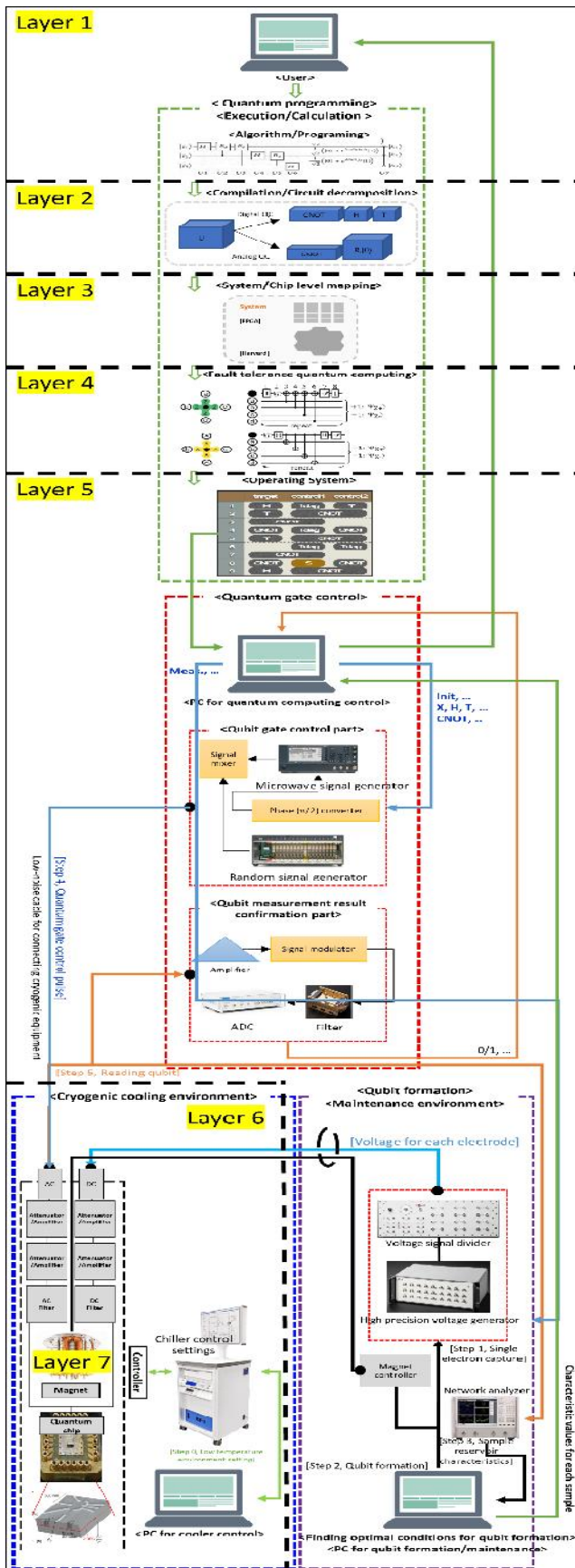


그림 3. 양자 컴퓨팅 시스템 풀 스택 구조

Fig. 3. Quantum computing system full stack structure.

IV. 성능 평가 기술 조사

1. Quantum Tomography

• **주요 분석 Layer 범위:** 5, 7

• **개요:** Quantum Tomography에는 크게 두 가지 기술로 나뉜다. Quantum State Tomography (QST)는 동일한 양자 상태의 앙상블에 대한 측정을 이용하여 양자 상태를 재구성하는 평가 기술이다^[23]. Quantum Process Tomography (QPT)는 시간에 따른 양자 시스템의 변화를 이용하여 양자 상태를 재구성하는 평가 기술이다^[24]. 양자 게이트가 양자 시스템에 어떤 영향을 미치는지 확인한다. 분석의 정확도는 높으나 분석 시간이 높은 편이다.

2. Randomized Benchmarking

• **주요 분석 Layer 범위:** 5, 7

• **개요:** Randomized Benchmarking는 양자 게이트를 무작위로 선택하여 결과를 측정하는 평가 기술이다^[25]. 특히 양자 게이트의 정확도를 평가하는데 많이 사용된다.

3. Quipper, Chisel-q, Scaffold and LIQUI

• **주요 분석 Layer 범위:** 1, 2, 3, (Scaffold, LIQUI), 4 (LIQUI)

• **개요:** Quipper, Chisel-q는 양자 컴퓨팅을 위한 함수형 프로그래밍 언어이다^[26-28]. Scaffold 양자 컴퓨팅 컴파일러이고, LIQUI는 양자 프로그래밍 언어, 컴파일러, 알고리즘 최적화 등을 지원하는 소프트웨어 아키텍처이다^[29, 30]. 이들은 컴파일러의 기능을 이용하여 성능 평가를 한다. LIQUI는 특정 큐비트 레이아웃에 독립적이며 오류 보정 코드를 지원한다. 오류 보정 코드를 적용할 수 있다.

4. DQC Architecture and QuDOS

• **주요 분석 Layer 범위:** 2, 3, 4

• **개요:** DQC(Distributed Quantum Computation) Architecture와 QuDOS(Quantum Dots with Optically controlled Spins)는 양자점 기반 양자 컴퓨터의 아키텍처를 제안하고 해당 아키텍처에서 소인수 분해 알고리즘의 계산 성능을 분석했다^[31, 32]. 양자 덧셈기(Adder)의 실행 시간과 양자 덧셈기가 이어진 개수인 깊이(Depth)를 곱하여 알고리즘의 계산 시간으로 평가했다.

5. Application Oriented Performance Benchmarking

- **주요 분석 Layer 범위:** 2, 3
- **개요:** Application Oriented Performance Benchmark는 양자 어플리케이션 실행을 중점으로 둔 성능 평가 기술이다^[33]. 다양한 알고리즘과 작은 어플리케이션에 대한 양자 컴퓨팅의 성능을 분석한다. 이는 회로의 깊이와 너비를 분석하는 Volumetric 평가 기술과 계산 시간 분석 기법을 사용한다.

6. MQT Bench

- **주요 분석 Layer 범위:** 1, 2, 3.
- **개요:** MQT Bench는 MQT(Munich Quantum Toolkit)의 벤치마크 제품군이다^[34]. 쇼어 알고리즘과 그로버 알고리즘뿐만 아니라 수많은 알고리즘을 하드웨어가 지원하는 게이트 세트, 큐비트 레이아웃에 맞춰 컴파일 하고 양자 자원 요구량을 분석하여 성능을 평가한다.

7. SupermarQ

- **주요 분석 Layer 범위:** 2, 3, 5, 7
- **개요:** SupermarQ는 성능 평가를 하기 위해 (Feature vector라는 점수 체계를 도입했다. NISQ를 가정하고 GHZ, Mermin-Bell, Phase Code 등의 알고리즘을 이용하는 평가 기법을 제시했다^[35].

8. QuRE

- **주요 분석 Layer 범위:** 2, 3, 4, 5
- **개요:** QuRE(Quantum Resource Estimator)는 양자 회로의 구현에 필요한 양자 자원 요구량과 계산 시간 등을 추정하는 도구이다^[36]. 6가지의 물리적 큐비트 레이아웃들을 지원하고 오류 보정을 위해 Surface code뿐만 아니라, Steane code, Bacon-shor code, Knill code 등 7가지의 오류 보정 코드를 지원한다. 하지만 양자 컴퓨팅의 모든 계층을 통합하지 않고 각 계층의 통계적인 수치를 계산한다.

9. Holistic Benchmarking

- **주요 분석 Layer 범위:** 1, 2, 3, 5, 7
- **개요:** Holistic Benchmark는 특정 분야에 특화된 세 가지의 회로를 제시하고 이들을 스택 구조, 응용 분야, 고전 시뮬레이션에 초점을 두는 세 가지의 벤치마크로 Heavy Output Probability, Cross-Entropy Difference, L1-Norm Distance라는 세 가지 항목을 계산하여 성능을 평가하는 기술이다^[37].

10. Azure Quantum Resource Estimation

- **주요 분석 Layer 범위:** 1, 2, 3, 4, 5, 7
- **개요:** Microsoft는 양자 컴퓨팅 시스템 스택 구조를 고려한 양자 자원 모델링 프레임워크를 제시했다^[38]. 해당 프레임워크는 상위 스택 구조는 위에서 아래로, 하위 스택 구조는 아래에서 위로 분석하는 특징이 있고 오류 보정 코드는 Surface code를 적용했다.

V. 양자 컴퓨팅 성능 분석 능력 비교

1. 양자 컴퓨팅의 성능 평가 항목

양자 컴퓨터 사용자 관점에서 제일 중요한 관심은 설정한 양자 컴퓨터의 계산 성능과 같은 지표이다. 따라서, 성능 평가 기술들은 양자 컴퓨팅의 실질적인 성능을 분석하는 기능이 요구된다. 이와 관련하여, 본 연구에서는 다음과 같은 양자 컴퓨팅 성능 평가 인자들을 고려한다.

가. 사용자 관점(User Perspective)

사용자 관점에서 특정 알고리즘을 구동하는데 필요한 양자 컴퓨팅 시스템 요구 수준 분석 요구 사항

(1) 양자 컴퓨팅 구동 시간(Time)

사용자가 설정한(혹은 제공되는) 양자컴퓨터 시스템에서 특정 알고리즘(및 입력 크기)을 1회 구동시 소요되는 시간

(2) 양자 컴퓨팅 계산 정확도(Accuracy)

사용자가 설정한(혹은 제공되는) 양자컴퓨터 시스템에서 특정 알고리즘(및 입력 크기)을 구동시 계산 결과 정확도

(3) 양자 컴퓨팅 구동시 자원 요구량(Resource)

사용자가 설정한(혹은 제공되는) 양자컴퓨터 시스템에서 특정 알고리즘(및 입력 크기)을 구동시 소요되는 물리적 자원 규모(예: 큐비트 수, 시스템 크기, 전력 소모량)

나. 시스템 관점(System Perspective)

양자 컴퓨터 시스템 관점에서 지원 가능한 양자 계산 능력

표 1. 양자 컴퓨팅 성능 분석 능력 비교

Table 1. Comparison of quantum computing performance analysis capabilities.

Methods	Parameter	User Perspective			System Perspective
		Time	Accuracy	Resource	Capacity
Quantum Tomography			◎		
Randomized Benchmarking			◎		
Quipper				○	
Chisel-q				○	
ScaffCC		○		○	
LIQUi		○		○	
DQC Architecture		△	△		△
QuDOS		△	△		△
Application Oriented Performance Benchmarking		○		○	○
MQT Bench				○	
SupermarQ		○	○	○	
QuRE		○			○
Holistic Benchmarking			○	○	
Azure Quantum Resource Estimation		○	○	○	○

(1) 양자 컴퓨팅 계산 용량(Capacity)

양자 컴퓨팅 시스템의 계산 용량 도출 능력

2. 평가 기술 비교

분석 대상 양자 컴퓨팅 성능 평가 기술들에 대한 도출 가능한 양자 컴퓨팅 성능 지표 기준 비교 결과는 표 1에 요약 설명되었다. 개별 양자 컴퓨팅 성능 평가 기술들의 주요 분석 결과는 다음과 같다.

가. Quantum Tomography, Randomized Benchmarking

- 큐비트의 양자 상태나 변화를 추정한다.
- Fidelity를 계산하여 정확도로 성능을 평가한다.

나. Quipper, Chisel-q, ScaffCC, LIQUi

- 컴파일러 기능을 이용하여 자원 요구량을 분석하여 성능을 평가한다.
- ScaffCC와 LIQUi는 알고리즘 계산 시간 추정 기능도 포함하고 있다.

다. DQC Architecture, QuDOS

- 계산 시간, 정확도, 양자 자원 요구량을 모두 분석하여 성능을 평가한다.
- 쇼어 알고리즘 분석에만 제한적으로 사용된다.

라. Application Oriented Performance Benchmarking

- 양자 자원 요구량을 분석하여 Quantum Volume을 계산한다.

- 알고리즘 계산 시간을 추정하여 성능을 평가한다.

마. MQT Bench

- 컴파일러 기능을 이용하여 양자 자원 요구량을 분석하여 성능을 평가한다.

바. SupermarQ

- 알고리즘 계산 시간 추정, 계산 정확도, 요구되는 양자 자원량을 분석한다.
- 이들이 제안한 Feature vector 수치를 계산하여 성능을 평가한다.

사. QuRE

- 입력된 회로의 계산에 소요되는 시간을 분석한다.
- 계산에 요구되는 큐비트와 게이트 개수를 추정하여 성능을 평가한다.

아. Holistic Benchmarking

- Heavy Output Probability, Cross-Entropy Difference, L1-Norm Distance라는 인자를 정의하여 성능을 평가한다.

자. Azure Quantum Resource Estimation

- 양자 컴퓨팅 시스템 각각의 스택에 대한 모델을 구축하였다.
- 양자 계산 시간, 양자 자원 요구량을 분석하여 성능을 평가한다.

3. 고찰

- **자원 요구량 분석 기능 포함:** 대부분의 평가 기술은 자원 요구량 분석 기능을 포함하고 있다. 이러한 측면은 양자 컴퓨팅 시스템에 대한 일반 사용자 혹은 도입 기관에서는 다양한 활용이 가능함을 의미한다.
- **구동 시간과 정확도 동시 평가 어려움:** 사용자 관점에서 양자 컴퓨팅 구동 시간과 계산 정확도를 동시에 도출하는 기술은 상대적으로 적었다. 실제로 양자 컴퓨팅의 연산 정확도가 높지 않다는 것을 고려하면, 이 두 가지 지표를 동시에 도출하는 능력이 중요한데, 현재까지는 많이 고려되지 않고 있음이 확인되었다.
- **양자 컴퓨팅 계산 용량 평가 어려움:** 시스템 관점에서 양자 컴퓨팅 계산 용량을 평가하는 기술의 개수가 적다. 이는 현재 대다수의 양자 컴퓨터 시스템 개발기관들이 개발하는 양자 컴퓨팅 시스템간 성능 비교가 어렵다는 것을 의미한다.

VI. 평가 기술의 동작 성능 비교

1. 평가 기술의 동작 성능 평가 항목

성능 평가 기술을 선정하는 과정에서는 다양한 인자들이 고려될 수 있다. 우선적으로는 양자 컴퓨팅의 성능을 주로 평가하는 기능을 고려할 것이다. 동시에 성능 평가 기술의 동작 성능 또한 매우 중요하다. 이에 따라 본 연구에서는 다음과 같은 평가 기술의 평가 인자들을 고려한다.

가. 분석 시간(Time)

평가 기술이 결과를 도출하는데 소요되는 총 시간으로 사용자 및 시스템 개발자 측면에서 분석시간이 적을수록 좋음

나. 확장성(Scalability)

양자 컴퓨팅 시스템 및 알고리즘에 대하여 특정 크기부터 임의의 크기까지 지원 가능 수준으로써 특히, 사용자 관점에서 다양한 확장성을 요구함

다. 편의성(Convenience)

사용자 관점에서 다양한 설정 조건에 대한 작업 편의성을 요구함

라. 분석 Layer 범위(Coverage)

개발자 관점에서 이상적 수준부터 실제 양자 하드웨

어 수준까지의 정교성을 요구함

2. 평가 기술 비교

분석대상 양자 컴퓨팅 성능 평가 기술들에 자체 성능에 대한 비교 결과는 표 2에 요약 설명되었다. 개별 양자 컴퓨팅 성능 평가 기술들의 주요 분석 결과는 다음과 같다.

가. Quantum Tomography, Randomized Benchmarking

- 폴 스택 하위 Layer를 집중적으로 분석한다.
- 하위 Layer를 분석하기 위해 많은 검증 회로를 계산하기 때문에 성능 평가에 오랜 시간이 소요된다.

나. Quipper, Chisel-q, LIQUI, Application Oriented Performance Benchmarking

- 폴 스택 상위 Layer를 분석한다.
- 상위 Layer 분석에는 짧은 시간이 소요된다.

다. Scaffold

- 폴 스택 상위 Layer를 분석한다.
- 상위 Layer 분석에는 짧은 시간이 소요된다.
- 대규모 회로에 대해 확장성을 갖기 위해 여러 기법을 사용한다.

라. DQC Architecture, QuDOS

- Surface code라는 양자 오류 정정 코드를 적용한 쇼어 알고리즘으로 성능을 분석한다.
- 특정 하드웨어에서만 분석하였기 때문에 분해 과정이 필요 없다.

마. MQT Bench

- 컴파일러가 양자 게이트 세트와 양자 하드웨어를 고려하기 때문에 Layer 1~3을 평가한다고 볼 수 있다.
- 사용자 인터페이스를 제공하여 편의성에 용이하다.

바. SupermarQ

- 다양한 양자 어플리케이션을 선택하여 평가 가능하다.
- 상위 Layer 일부분과 하부 Layer를 평가한다.
- NISQ 시스템을 가정하여 성능을 평가한다.

사. QuRE

- 모듈식 설계를 사용해서 확장성을 갖는다.
- 양자 오류 정정 코드로 Steane code, Surface code를 채택하였다.

표 2. 평가 기술의 동작 성능 비교

Table 2. Comparison of behavioral performance of assessment techniques.

Methods	Parameter	Time	Scalability	Convenience	Coverage (Layers)						
					1	2	3	4	5	6	7
Quantum Tomography									○		◎
Randomized Benchmarking									○		◎
Quipper		○			◎	○	○				
Chisel-q		○			◎	○	○				
ScaffCC		○	○		○	◎	◎				
LIQUi		○			○	◎	◎	Steane Code etc.			
DQC Architecture						△		Surface Code			
QuDOS						△		Surface Code			
Application Oriented Performance Benchmarking		○				○	○				
MQT Bench			○	○	○	○	○				
SupermarQ				○		○	○	NISQ			○
QuRE			○			○	○	Steane Code, Surface Code	○		○
Holistic Benchmarking			○		○	○	○		○		○
Azure Quantum Resource Estimation			○	○	○	○	○	Surface Code	○		○

아. Holistic Benchmarking

- 대부분의 스택 계층을 고려한 평가 기법이다.
- 3가지 벤치마크를 사용하여 확장성이 좋다.

자. Azure Quantum Resource Estimation

- 대부분의 스택 계층을 고려한 평가 기법이다.
- 대규모 양자 컴퓨팅 환경에서도 적절한 평가를 가능하게 하여 확장성이 좋다.
- 사용자의 편의성을 고려하였다.

3. 고찰

- **이상적 수준 분석 대부분:** 대부분의 평가 기술이 상부 Layer를 포함하여 성능을 평가한다. 이는 양자 알고리즘을 연구하는 측면에서는 매우 효과적이나, 그러한 알고리즘이 실제 양자 컴퓨팅 시스템에서 어느 정도의 양자 계산 시간을 요구하는지를 명확하게 하지는 못하는 한계를 의미한다.
- **실제 수준 분석시 분석 요구 시간 증가:** 양자 컴퓨팅 시스템의 하위 Layer를 포함하는 경우, 분석 시간이 길어지는 것을 확인하였다. 이는 하위 Layer (Layer 5, 7)가 대부분 양자 현상을 다루고 있기 때문에 발생하는 것으로써, 근본적으로는 해결하기

어려운 문제이다. 그러나, 상위 Layer에서 도출하고자 하는 양자 컴퓨팅 계산 시간의 정교함을 고려하면, 이러한 이슈는 해결되어야 한다.

- **폴 스택 구조 중 누락 계층 존재:** 특정 Layer의 경우에는 평가 기술이 없음이 확인되었다. 본 연구를 통해서 양자 컴퓨팅 시스템 폴 스택 구조상 Layer 6을 평가하는 기술이 없음을 확인하였다. 향후 대용량 양자 컴퓨팅 시스템을 개발하는 과정에서는 이러한 Layer에서의 성능이 큰 영향을 끼치는 만큼, 향후 양자 컴퓨팅 성능 평가 기술은 이러한 부분을 고려할 필요가 있음을 확인하였다.

VII. 결 론

본 연구에서는 점진적으로 중요성이 높아지고 있는 양자 컴퓨팅 성능 평가 기술들에 대해서 비교 분석하였다. 특히, 기존에는 고려되지 않았던 양자 컴퓨팅 시스템 폴 스택 구조를 대상으로 한 새로운 측면에서의 비교 분석이 시도되었다.

이를 통해서, 대부분의 양자 컴퓨팅 성능 평가 기술이 양자 자원 요구량을 분석하는 기능을 갖고 있음을 확인하였다. 그러나, 양자 컴퓨팅 시스템 폴 스택 구조

상 특정 Layer의 영향을 분석하는 기능은 포함되지 않은 것으로 확인되었다. 또한, 상위 수준 Layer에 비해서 하위 수준 Layer의 분석 시간이 매우 긴 것을 확인하였는데, 이로 인해서 전체적인 양자 컴퓨팅 시스템의 구동 성능을 정교하고 빠르게 분석하는 것이 쉽지 않음을 예상할 수 있었다.

이러한 분석을 토대로, 향후 양자 컴퓨팅 성능 평가 기술은 양자 컴퓨팅 시스템 풀 스택 구조를 모두 포함하면서, 더 빠르게 계산하는 기술이 시급함을 확인하였다. 또한, 각 Layer별로 다양한 기술적 접근법들에 대한 다양한 분석 기능이 포함되는 것이 요구됨을 확인하였다. 최종적으로, 양자 컴퓨팅 시스템을 사용하는 측면과 개발하는 측면에서 볼 때 이러한 양자 컴퓨팅 성능 평가 기술을 고도화하는 것이 매우 중요함을 확인하였다.

REFERENCES

- [1] Michael A Nielsen, Isaac L Chuang, "Quantum computation and quantum information", Cambridge University Press, 2000.
- [2] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring", Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 124-134, 1994.
- [3] Lov K. Grover, "A fast quantum mechanical algorithm for database search", Association for Computing Machinery, pp. 212-219, 1996.
- [4] Dial, Oliver E, "Moving the Needle on Scale", APS March Meeting 2022, Volume 67, Number 3, 2022.
- [5] Arute, F., Arya, K., Babbush, R. et al., "Quantum supremacy using a programmable superconducting processor", Nature, 574, pp. 505-510, 2019.
- [6] Qiskit contributors, "Qiskit: An Open-source Framework for Quantum Computing", Zenodo, 2023.
- [7] Cirq Developers, "Cirq", Zenodo, 2023.
- [8] J.R. Johansson and P.D. Nation and F. Nori, "QuTiP: An open-source Python framework for the dynamics of open quantum systems", Computer Physics Communications, Volume 183, Number 8, pp. 1760-1772, 2012.
- [9] J.R. Johansson and P.D. Nation and F. Nori, "QuTiP 2: A Python framework for the dynamics of open quantum systems", Computer Physics Communications, Volume 184, Number 4, pp. 1234-1240, 2013.
- [10] Michael Broughton, Guillaume Verdon. Et al., "TensorFlow Quantum: A Software Framework for Quantum Machine Learning", arXiv, 2021.
- [11] Zhu, D., Kahanamoku-Meyer, G.D., Lewis, L. et al., "Interactive cryptographic proofs of quantumness using mid-circuit measurements", Nat. Phys., Volume 19, pp. 1725-1731, 2022.
- [12] Y. Kwon, B. S. Choi, "A Software Platform for Programmable Linear Optical Quantum Computer", IEEE Access, Volume 11, pp. 112682-112692, 2023.
- [13] Tarucha, S., Yamamoto, M., Oiwa, A., Choi, B.S., Tokura, Y., "Spin Qubits with Semiconductor Quantum Dots", Springer, 2016.
- [14] Yongsoo Hwang, Taewan Kim, Chungheon Baek, Byung-Soo Choi, "Integrated Analysis of Performance and Resources in Large-Scale Quantum Computing", Phys. Rev. Appl., Volume 13, Number 27, 2020.
- [15] Andrew W. Cross, Lev S. Bishop, John A. Smolin, Jay M. Gambetta, "Open Quantum Assembly Language", arXiv, 2017.
- [16] N. Khammassi, G.G. Guerreschi, I. Ashraf, J. W. Hogaboam, C. G. Almudever, K. Bertels, "cQASM v1.0: Towards a Common Quantum Assembly Language", arXiv, 2018.
- [17] Yuval R Sanders, Joel J Wallman, Barry C Sanders, "Bounding quantum gate error rate based on reported average fidelity", arXiv, 2015.
- [18] Sheldon, Sarah and Magesan, Easwar and Chow, Jerry M. and Gambetta, Jay M., "Procedure for systematically tuning up cross-talk in the cross-resonance gate", Phys. Rev. A, Volume 93, Number 5, 2016.
- [19] R. W. Hamming, "Error detecting and error correcting codes", The Bell System Technical Journal, Volume 29, Number 2, pp. 147-160, 1950.
- [20] Andrew Steane, "Simple Quantum Error Correcting Codes", arXiv, 1996.
- [21] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, Andrew N. Cleland, "Surface codes: Towards practical large-scale quantum computation", Phys. Rev. A, Volume 86, Number 48, 2012.
- [22] Peter W. Shor, "Fault-tolerant quantum computation", arXiv, 1997
- [23] Altepeter, J.B., James, D.F., Kwiat, P.G., "4

- Qubit Quantum State Tomography”, Springer, 2004.
- [24] M. Mohseni, A. T. Rezakhani, and D. A. Lidar, “Quantum-process tomography: Resource analysis of different strategies”, *Phys. Rev. A*, Volume 77, Number 15, 2007.
- [25] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin, D. J. Wineland, “Randomized Benchmarking of Quantum Gates”, arXiv, 2007.
- [26] Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron, “Quipper: a scalable quantum programming language”, Association for Computing Machinery, 2013.
- [27] Jonathan M. Smith, Neil J. Ross, Peter Selinger, Benoît Valiron, “Quipper: Concrete Resource Estimation in Quantum Algorithms”, arXiv, 2014.
- [28] X. Liu and J. Kubitowicz, “Chisel-Q: Designing quantum circuits with a scala embedded language,” 2013 IEEE 31st International Conference on Computer Design (ICCD), pp. 427-434, 2013.
- [29] Ali JavadiAbhari, Shruti Patil, Daniel Kudrow, Jeff Heckey, Alexey Lvov, Frederic T. Chong, Margaret Martonosi, “ScaffCC: Scalable compilation and analysis of quantum programs”, *Parallel Computing*, Volume 45, pp. 2-17, 2015.
- [30] Dave Wecker, Krysta M. Svore, “LIQ|>: A Software Design Architecture and Domain-Specific Language for Quantum Computing”, arXiv, 2014.
- [31] N. Cody Jones, Meter Rodney Van, Austin G. Fowler, Peter L. McMahon, Jungsang Kim, Thaddeus D. Ladd, Yoshihisa Yamamoto, “Layered Architecture for Quantum Computing”, *Phys. Rev. X*, Volume 2, Number 27, 2012.
- [32] Rodney Van Meter, Thaddeus D. Ladd, Austin G. Fowler, Yoshihisa Yamamoto, “Distributed Quantum Computation Architecture Using Semiconductor NanoPhotonics”, arXiv, 2009.
- [33] Thomas Lubinski, Sonika Johri, Paul Varosy, Jeremiah Coleman, Luning Zhao, Jason Necaie, Charles H. Baldwin, Karl Mayer, Timothy Proctor, “Application-Oriented Performance Benchmarks for Quantum Computing”, arXiv, 2023.
- [34] Nils Quetschlich, Lukas Burgholzer, Robert Wille, “MQT Bench: Benchmarking Software and Design Automation Tools for Quantum Computing”, *Quantum*, Volume 7, pp. 1062, 2023.
- [35] Teague Tomesh, Pranav Gokhale, Victory Omole, Gokul Subramanian Ravi, Kaitlin N. Smith, Joshua Viszlai, Xin-Chuan Wu, Nikos Hardavellas, Margaret R. Martonosi, Frederic T. Chong, “SupermarQ: A Scalable Quantum Benchmark Suite”, arXiv, 2022.
- [36] M. Suchara, J. Kubitowicz, A. Faruque, F. T. Chong, C. -Y. Lai and G. Paz, “QuRE: The Quantum Resource Estimator toolbox”, 2013 IEEE 31st International Conference on Computer Design (ICCD), pp. 419-426, 2013.
- [37] Daniel Mills, Seyon Sivarajah, Travis L. Scholten, Ross Duncan, “Application-Motivated, Holistic Benchmarking of a Full Quantum Computing Stack”, *Quantum*, Volume 5, pp. 415, 2021.
- [38] Michael E. Beverland, Prakash Murali, Matthias Troyer, Krysta M. Svore, Torsten Hoefler, Vadym Kliuchnikov, Guang Hao Low, Mathias Soeken, Aarthi Sundaram, Alexander Vaschillo, “Assessing requirements to scale to practical quantum advantage”, arXiv, 2022.

— 저 자 소 개 —



이 두 원(비회원)
2019년~국립부경대학교
물리학과 학부과정
재학중.

<주관심분야: 양자컴퓨터, 양자물리>



최 병 수(정회원)
1996년 충남대학교 컴퓨터공학과
학사 졸업.
1998년 광주과학기술원
정보통신공학 석사 졸업.
2004년 광주과학기술원
정보통신공학 박사 졸업.

2004년~2006년 영국 요크대학교 박사후연구원.
2006년~2007년 성균관대학교 전자전기공학부
박사후연구원/연구교수.
2007년~2010년 이화여자대학교 전자전기공학부
연구교수.
2010년~2011년 서울시립대학교 양자정보기술
연구단 연구원.
2011년~2013년 미국 듀크대학교
전기컴퓨터공학과 연구원.
2013년~2015년 일본 동경대학교 응용물리학과
연구원/이화학연구소 방문연구원.
2015년~2022년 한국전자통신연구원 양자컴퓨팅
연구실장.
2022년~국립부경대학교 과학컴퓨팅학과 조교수.
<주관심분야: 양자컴퓨팅 전반>