논문 리뷰

Factoring with n+2 clean qubits and n-1 dirty qubits (1)

발표자: 양유진

링크: https://youtu.be/zqcKiB_i7wg

한성대학교 HANSUNG UNIVERSITY

CryptoCraft LAB

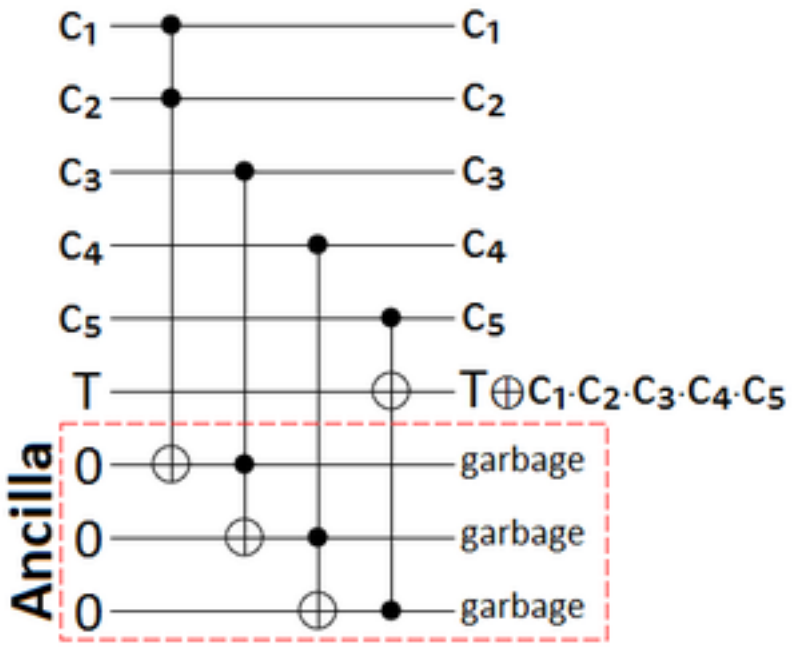
0. Objective

- dirty ancilla qubit 만 사용하여 다양한 산술 연산을 사용하는 방법과 다양한 기술 살펴보기
- 위 방법을 Shor 알고리즘에 적용하여 clan qubit 수를 1.5n + O(1)에서 n + 2으로 줄이기

1. Introduction

1) Ancilla qubit 이란?

- 임시작업공간 회로에서 사용할 수 있는 추가적인 qubit
- clean qubit, dirty qubit로 나뉨



[그림1] Ancilla qubit

(출처: https://en.wikipedia.org/wiki/Ancilla_bit)

1. Introduction

2) dirty qubit vs clean qubit

dirty qubit

- 회로가 끝나기 전에 복원되어야 하는 알려지지 않은 상태의 추가적인 qubit
- 회로의 다른 부분에서 사용된 dirty qubit를 빌릴 수 있음

→ 좁은 공간적 제약이 있거나, 다른 보조장치가 너무 멀리 있어서 qubit을 획득하기 쉽지 않은 회로 토폴로지에 적용하기 쉬움

- 임시적으로 사용하지 않는 qubit는 빌릴 수 있는 dirty qubit임.

→ dirty qubit ≻ clean qubit

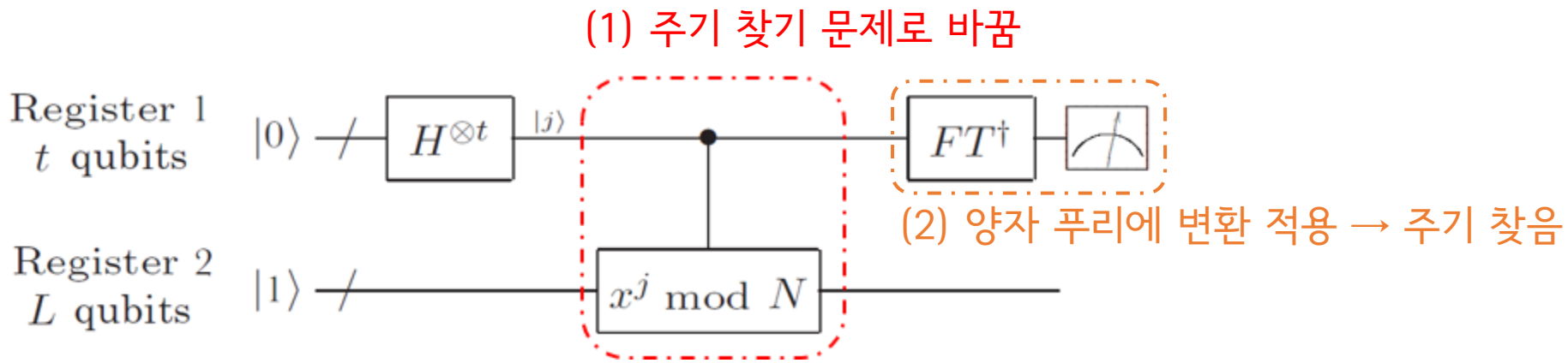
clean qubit

- 알려진 상태의 추가적인 qubit
- 회로를 압축적으로 만들어주기 때문에 일반적으로 **가치있게** 여겨짐

1. Introduction

3) Shor 알고리즘이란?

- 지수 시간이 걸리는 인수 분해 문제와 이산 대수 문제를 다항 시간 내에 풀 수 있게 도와주는 양자 알고리즘
- 두 난제는 공개키 암호 RSA, ECC가 기반하는 난제임 → 공개키 암호에 보안적 위협이 됨
- 인수분해 문제를 주기찾기 문제로 바꾼 후 푸리에 변환을 적용하여 답을 찾음



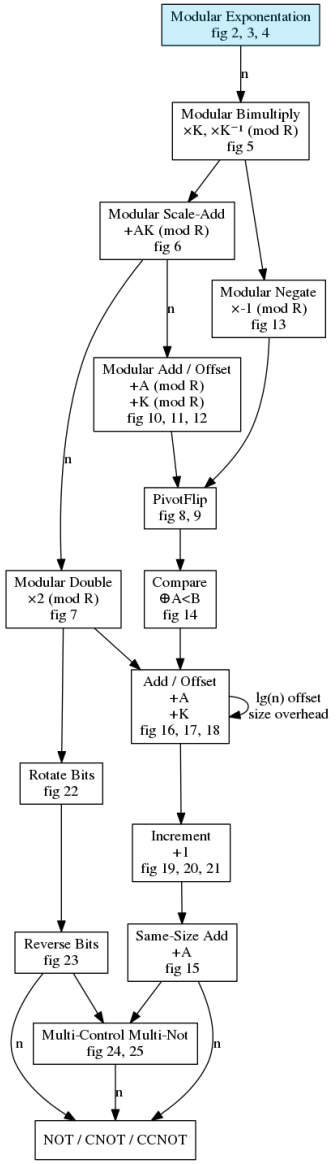
[그림2] Shor 알고리즘 양자 회로도

(출처: https://hiqsimulator.readthedocs.io/en/latest/examples/examples.ShorAlgorithm.html)

2. Constructions

- Shor 알고리즘의 주기 찾기 단계를 일정한 크기의 게이트로 축소하는 데는 많은 회로 구성이 사용됨.

[그림3] 주기 찾기 단계 크기 축소에 사용되는 다양한 작업의 개요와 수행 경로 →



2. Constructions

2.1 Modular Exponentiation을 이용한 주기 찾기

= 모듈러 거듭제곱법

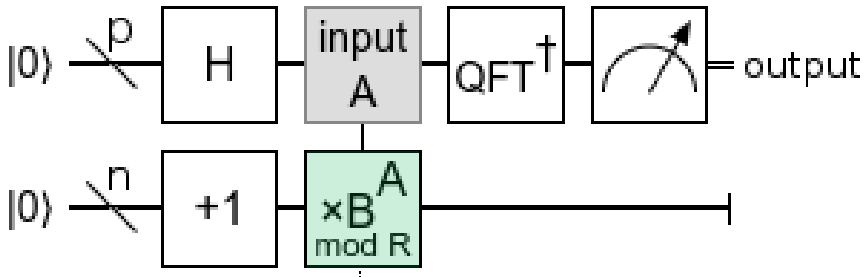
정수 b(밑수base)를 지수 e의 거듭 제곱으로 올리고 양의 정수 m(모듈러스modulus)로 나눈 나머지

$$c = b^e \bmod m \quad (0 \le c < m)$$

ex) $b = 5, e = 3, m = 13$ 인 경우 $5^3 \bmod 13 = 13 * 9 + 8 \rightarrow c = 8$

- 매우 큰 정수도 효율적으로 계산 가능 → 즉, 실행 시간 & 메모리를 절약할 수 있음
- 이산로그 계산(b, c, m으로 e찾기)은 어려운 것으로 간주됨.

2. Constructions 2.1 Modular Exponentiation을 이용한 주기 찾기

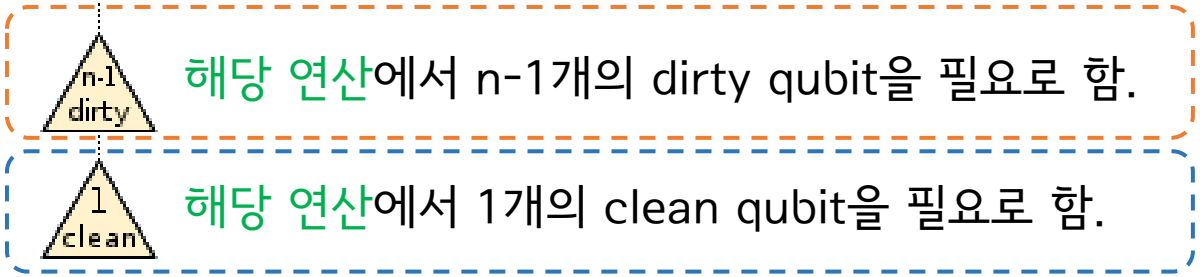


R : 인수분해 하고자 하는 자연수

B : R과 서로소 관계인 난수 $(0 < B < R)$

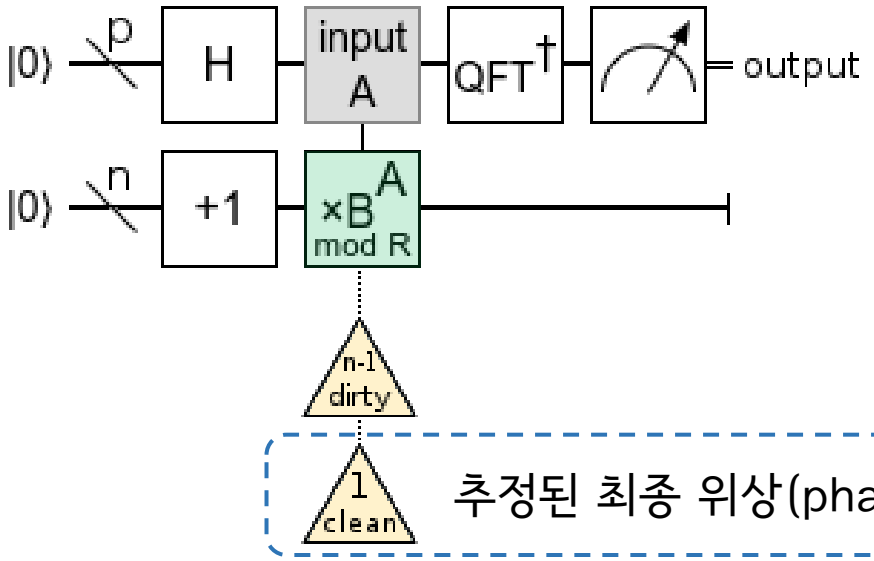
노란색_삼각형

: 개수가 충분치 않은 경우



[그림4] modular exponentiation이 적용된 Shor 알고리즘의 주기 찾기 양자 회로 (상위수준ver.)

2. Constructions 2.1 Modular Exponentiation을 이용한 주기 찾기



R : 인수분해 하고자 하는 자연수

B : R과 서로소 관계인 난수 $(0 < B < R)$

Unitary operator $\boldsymbol{U}$: $\boldsymbol{U}|y\rangle = |By \bmod R\rangle$

각 $\boldsymbol{U}$ 의 위상을 n번 shift 해가면서 추정 → 정확도 높임

[STEP1] Hadamard 게이트를 통해 중첩 상태 $|\psi_0\rangle$ 준비 $\left(|\psi_0\rangle = \sqrt{2^{-n}}\sum_{k=0}^{2^{-n}}|k\rangle\right)$

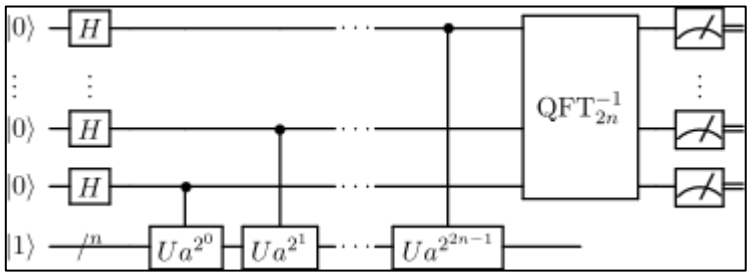
[STEP2] 지수 A를 입력으로 하여 $\times B^A \bmod R$ 계산 → 위상(phase) s/l 추정 $\left|\psi_{1,x}\right\rangle = \sqrt{l2^{-n}}\sum_{k=0}^{\approx(2^n/l)^{-1}}|lk+x\rangle$

[STEP3] $\mathrm{QFT}^\dagger$(역 양자푸리에 변환) 적용 → $\sqrt{l^{-1}}\sum_{s=0}^{l-1}|s/l\rangle|u_s\rangle$

[STEP4] 관측(measure) → s/l 구함

[STEP5] 연속분수알고리즘을 적용하여 주기 l 찾음

2. Constructions 2.1 Modular Exponentiation을 이용한 주기 찾기



Hadamard 게이트는 앞,뒤로 적용되어 있음

X축 회전

Figure 3: Period finding with a single phase-estimation qubit [2]. The small oplus' ($\oplus$) are "X-axis controls". An X-axis control is equivalent to a normal control, but with a Hadamard gate applied before and after. It conditions on the state $\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$ instead of on the state $|1\rangle$.

주기 찾기는 QFT^{-1}를 수행한 직후에 모든 큐비트를 측정하기때문에 양자푸리에 변환된 대부분의 큐비트는 표시된 것보다 일찍 측정이 가능함.

2. Constructions 2.1 Modular Exponentiation을 이용한 주기 찾기

bimultiplication (이중 곱셈)

$n = \lceil \lg_2(R) \rceil$

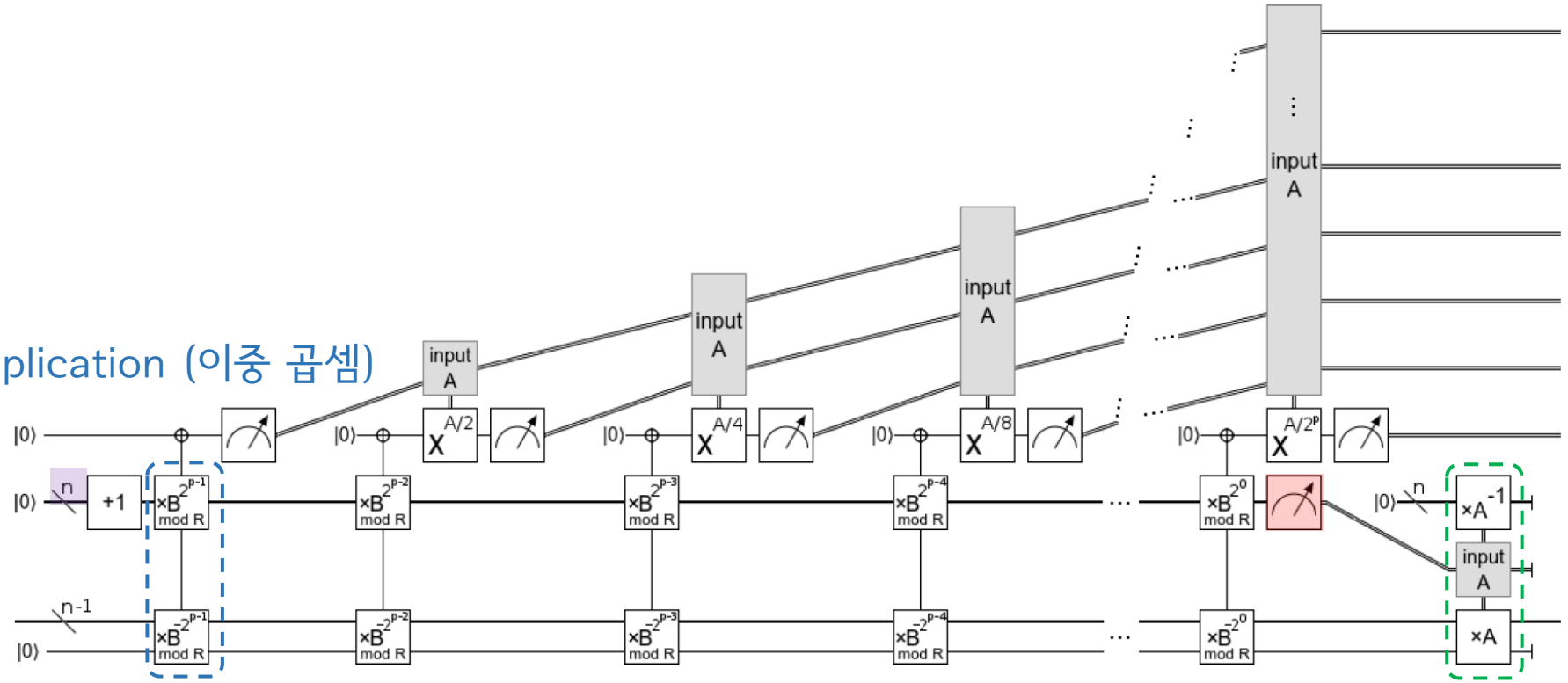


Figure 4: Period finding with a single phase-estimation qubit and paired inverse multiplications ("bimultiplications"). Uses $O(n^3 \lg n)$ gates, $O(n^3)$ depth, and no additional qubits beyond those shown.

dirty register의 원래 값을 복원하는 데 필요한 수정 요소를 복구하기 위하여 work register를 측정

work register를 폐기하는 대신 dirty ancilla register를 폐기하는 상수 A의 inverse 상수 A^{-1} 곱함

감사합니다