https://doi.org/10.33778/kcsa.2019.19.3.061

동형암호적 양자계산이 가능한 양자오류정정부호 기법★

손 일 권*, 이 종 현**, 이 원 혁*, 석 우 진*, 허 준**

요 약

최근 엄청난 계산 능력을 보여주는 양자 컴퓨터와 정보 접근성이 높고 비용이 낮은 클라우드 컴퓨팅에 대한 개발이 활발하게 이루어지고 있다. 이러한 양자 컴퓨터의 경우 양자오류정정부호가 필수적이며, 클라우드 컴퓨팅의 경우 보안성 및 계산성을 확보하기 위해 동형암호가 사용될 수 있다. 각각 다른 목적을 위해 사용되는 이 두 기법은 서로 비슷한 가정을 바탕으로 하고 있어, 양자오류정정부호를 기반으로 동형암호를 구성하는 연구들이 진행되어왔다. 따라서 본 논문에서는 일반적인 양자오류정정부호를 변형하여 동형암호적 양자정보처리가 가능한 기법을 제시한다. 기존의 양자오류정정부호를 이용한 동형암호 기법의 경우 부호를 사용하였지만 오류정정 능력이 전혀 없는데 반해, 제시한 양자오류정정부호 기법을 사용하면 동형암호적 양자정보처리가 가능하면서도, 동시에 양자오류정정부호 본연의 기능인 양자정보의 연산, 저장 중의 오류를 정정할 수 있는 장점이 존재한다.

Quantum Error Correction Code Scheme used for Homomorphic Encryption like Quantum Computation

Sohn Il Kwon*, Lee Jonghyun**, Lee Wonhyuk*, Seok Woojin*, Heo Jun**

ABSTRACT

Recently, developments on quantum computers and cloud computing have been actively conducted. Quantum computers have been known to show tremendous computing power and Cloud computing has high accessibility for information and low cost. For quantum computers, quantum error correcting codes are essential. Similarly, cloud computing requires homomorphic encryption to ensure security. These two techniques, which are used for different purposes, are based on similar assumptions. Then, there have been studies to construct quantum homomorphic encryption based on quantum error correction code. Therefore, in this paper, we propose a scheme which can process the homomorphic encryption like quantum computation by modifying the QECCs. Conventional quantum homomorphic encryption schemes based on quantum error correcting codes does not have error correction capability. However, using the proposed scheme, it is possible to process the homomorphic encryption like quantum computation and correct the errors during computation and storage of quantum information unlike the homogeneous encryption scheme with quantum error correction code.

Key words : Quantum Error Correction Code, Homomorphic Encryption, Universal Quantum Computation, Quantum Information Processing

접수일(2019년 8월 13일), 수정일(1차: 2019년 9월 19일), 게재확정일(2019년 9월 26일)

★ 본 연구는 2019년도 한국과학기술정보연구원(KISTI) 주요사업 과제로 수행한 것입니다. 본 연구는 과학기술정보통신부 ITRC 사업 (IITP-2019-2018-0-01402)으로 수행 되었음

* 한국과학기술정보연구원/과학기술연구망센터
** 고려대학교/전기전자공학부

1. 서 론

양자정보처리를 통한 양자 컴퓨팅[1]은 고속 병렬 연산을 통해 고전 컴퓨터가 처리하기 어려운 문제를 해결할 수 있을 것으로 기대된다. 그러나 연산에 사용되는 큐비트는 스스로 붕괴하는 성질 및 외부 간섭에 취약하며, 현재 양자 소자들의 성능이 매우 낮아 바로 양자 컴퓨터를 제작하는데 사용되는 것은 어려운 상황이다. 따라서 양자오류정정부호를 사용하여 큐비트 및 게이트에서 발생하는 오류를 정정[2]하고 연접을 통해 수명, 성능을 향상시켜 사용한다. 1995년 Peter shor가 양자오류정정부호를 통해 양자정보처리 과정에서 발생하는 오류를 정정하는 것이 가능하다는 사실[2]을 발표함으로써 각종 양자정보처리 기술이 주목받게 되었다. 그러나 디지털 시스템과는 다르게 양자정보는 복사가 불가능[3]하다. 따라서 기존의 오류정정부호처럼 간단하게 정보를 복사하여 패리티 비트에 저장 및 연산하는 방식으로 부호를 생성할 수 없다. 이러한 어려움 때문에 초기에는 기존의 오류정정부호들을 양자오류정정부호에 이용하는 연구[4,5]가 주를 이루었고, 이후 Topological 부호와 같이 양자 고유의 성질을 이용한 양자오류정정부호들에 대한 연구가 진행되었다[6-8].

현재 여러 대형 IT 기업들이 이러한 양자오류정정부호를 사용하여 양자 컴퓨터를 개발하고 있다. 그중에서 IBM은 올해 CES 2019에서 최초의 상용 20-큐비트 양자 컴퓨터 IBM Q System One을 공개하였으며, 2016년부터 일반 사용자에게 클라우드 양자 컴퓨팅 플랫폼 IBM Q Experience를 통해 5-큐비트 양자 컴퓨터를 공개하여 여러 연구가 이루어지고 있다[9-11].

여기서 클라우드 컴퓨팅이란 인터넷을 통해 가상화된 컴퓨터의 연산 자원을 사용할 수 있는 것을 말한다[12]. 즉 이용자의 정보를 자신의 컴퓨터가 아닌 클라우드에 연결된 다른 컴퓨터로 처리하는 기술이다. 최근 데스크톱, 태블릿, 랩톱, 스마트폰 등 다양한 기기들이 인터넷에 연결이 가능해지면서 클라우드 컴퓨팅이 보편화 되어가고 있다. 클라우드 컴퓨팅은 초기 구축 비용이 적고, 다양한 기기가 단말로 사용될 수 있으며 일관된 사용자 환경을 구축 가능하다는 장점이 있다.

그러나 클라우드 컴퓨팅은 서비스 제공자가 클라우드에 저장된 데이터에 항상 접근할 수 있어 개인정보 및 보안 데이터 유출 가능성이 있다는 큰 단점이 존재한다. 유출 문제를 해결하기 위해 데이터를 일반적인 암호체계를 사용하여 암호화 할 경우, 모든 데이터 처리가 서버에서 이루어지는 클라우드 컴퓨팅 특성상 연산 전에 복호화가 필요하여 해당 문제를 전혀 해결할 수 없다.

클라우드 컴퓨팅의 이러한 보안 문제를 동형암호[13-15]를 통해 해결할 수 있다. 동형암호란 간단히 말해 암호화된 데이터를 복호화를 하지 않고도 연산이 가능한 암호화 기법을 말한다. 즉, 암호화된 상태에서 연산한 결과값을 복호화했을 때, 평문 상태에서 연산한 결과와 동일한 값을 얻을 수 있다. 따라서 동형암호를 사용하면 클라우드 컴퓨터에 암호키를 제공하지 않고도 연산이 가능하며, 클라우드 컴퓨팅 환경에서 발생하는 보안 문제를 해결할 수 있다. 이러한 동형암호는 양자정보처리 분야에서도 활발히 연구되고 있다[16-20]. 이 중에서 일부 논문들은 양자오류정정부호의 논리적 게이트 구성 특성을 이용하여 양자 동형암호를 구현하는 연구를 진행하였지만[17-19], 양자오류정정부호의 논리적 게이트와 물리적 게이트의 상관관계를 이용하였을뿐 부호의 오류정정 능력에는 주목하지는 않은 연구였다.

본 논문에서는 현재 개발이 활발하게 이루어지고 있는 양자 컴퓨터 및 클라우드 컴퓨팅 환경에서 사용이 가능한 동형암호적 양자 계산이 가능한 양자오류정정부호 기법을 제시한다. 기존의 양자오류정정부호의 특성을 이용한 양자동형암호 기법들과는 다르게 양자오류정정부호의 오류정정 능력을 그대로 가지고 있는 기법이다. 2장에서는 양자오류정정부호 중 하나인 안정부호와 범용 양자 계산 게이트 세트를 설명하고, 3장에서 동형암호에 대해 간략히 설명한다. 마지막으로 4장에서는 양자오류정정부호를 사용하여 양자동형암호를 구성한 기존연구에 대해 설명한다. 마지막으로 5장에서는 양자오류정정부호와 동형암호의 기본 가정의 유사

성을 설명하고 이를 통해 동형암호적 양자 계산이 가능한 양자오류정정부호 기법을 제안한다.

2. 양자오류정정부호

2.1 안정 부호(stabilizer code)

본 논문에서는 양자오류정정부호들 중 가장 기본적인 부호인 안정 부호(stabilizer code)[21]를 소개한다. 안정 부호는 가장 많은 연구가 이루어졌으며, 디지털 시스템에서 쓰이고 있는 선형 오류정정부호와 유사한 특성들을 보인다.

양자오류정정부호는 양자 정보에 발생하는 오류를 정정할 수 있는 부호이다. 이러한 양자오류는 단일 큐비트 Pauli 연산자들로 표현이 가능하며, Pauli 연산자는 다음과 같다.

$$I=\begin{bmatrix}1&0\\0&1\end{bmatrix}, X=\begin{bmatrix}0&1\\1&0\end{bmatrix}, \quad (1)$$
$$Y=\begin{bmatrix}0&-i\\i&0\end{bmatrix}, Z=\begin{bmatrix}1&0\\0&-1\end{bmatrix},$$

안정 연산자(stabilizer) 그룹 S는 안정 부호를 구성하는데 있어 가장 중요한 그룹이다. 안정 연산자는 안정 부호의 코드워드에 대해 '+1'의 고윳값을 갖는 연산자이다. 따라서 안정 부호의 코드워드는 안정 연산자의 고유 벡터임을 알 수 있다.

$$S_i|\psi\rangle = |\psi\rangle \quad (2)$$

안정 연산자 그룹 S는 안정 연산자 생성자(stabilizer generator)를 통해 표현 가능하다.

$$S=\langle S_1, S_2, \ldots S_m\rangle \quad (3)$$

X, Z 등의 연산자가 물리적 큐비트에 작용하는 역할을 코드워드 수준에서 동일하게 동작하는 연산자를 논리적 연산자라고 한다. 이 중 X, Z 연산자의 역할을 하는 논리적 연산자를 $\overline{X}$, $\overline{Z}$로 나타내며, 두 논리적 연산자는 안정 연산자 모두와 commute한 관계를 가진다. 보호하려는 정보가 k 큐비트인 경우 $\overline{X}$, $\overline{Z}$는 $\overline{X_1}\cdots\overline{X_k}$, $\overline{Z_1}\cdots\overline{Z_k}$,로 각각 k개가 존재한다. 이 중 $\overline{X}$는 논리 $|0\rangle$ 벡터인 $|\overline{0}\rangle$와 함께 코드워드들을 생성하는데 사용되기 때문에 $\overline{X_1}\cdots\overline{X_k}$를 통해 2^k개의 코드워드를 구성할 수 있다.

안정 부호는 다음과 같은 단계를 통해 구성 가능하다. 앞서 설명한대로 양자 정보는 복제할 수 없기 때문에, n 큐비트를 사용하여 k 큐비트의 정보를 보호하는 $[[n,k,d]]$ 부호의 경우 먼저 n 큐비트의 $|0\rangle^{\otimes n}$ 기저를 준비해야 한다. 이를 안정 연산자와의 연산을 통해 $|\overline{0}\rangle$로 변환한다.

$$|\overline{0}\rangle = \frac{1}{(\sqrt{2})^{n-k}}\sum_{S_i\in S} S_i|0\rangle^{\otimes n} \quad (4)$$

안정 연산자를 통해 구성한 $|\overline{0}\rangle$에 $\overline{X_1}\cdots\overline{X_k}$들을 연산해줌으로써 코드워드를 구성할 수 있다.

$$|\overline{c_1c_2\ldots c_k}\rangle = \overline{X_1}^{c_1}\overline{X_1}^{c_2}\cdots\overline{X_1}^{c_k}|\overline{0\cdots0}\rangle \quad (5)$$

큐비트는 이진 정보의 형태를 갖기 때문에 c_i는 $\{0,1\}$중 하나의 값을 가진다. c_i가 1일 경우에는 $|\overline{0}\rangle$에 $\overline{X_1}\cdots\overline{X_k}$가 작용되어 $|\overline{0\cdots c_i\cdots0}\rangle$가 된다.

예를 들어 $[[5,1,3]]$ 안정 부호[10]의 구성 과정은 다음과 같다. $[[5,1,3]]$ 부호의 안정 연산자 생성자는 다음과 같다.

$$XZZXI,\ IXZZX,\ XIXZZ,\ ZXIXZ \quad (6)$$

위의 $[[5,1,3]]$ 부호 안정 연산자들을 통해 $|\overline{0}\rangle$을 구성하면 다음과 같다.

$$|\overline{0}\rangle = \frac{1}{(\sqrt{2})^4}\sum_{S_i\in S} S_i|0\rangle^{\otimes n} = \quad (7)$$
$$\frac{1}{4}(|00000\rangle+|10010\rangle+|01001\rangle+|10100\rangle$$
$$+|01010\rangle-|11011\rangle-|00110\rangle-|11000\rangle$$
$$-|11101\rangle-|00011\rangle-|11110\rangle-|01111\rangle$$
$$-|10001\rangle-|01100\rangle-|10111\rangle+|00101\rangle)$$

안정 연산자와의 commute 관계를 가지는 연산자들로부터 구한 5-큐비트 안정 부호의 $\overline{X}$는 $XXXXX$이다. 이를 $|\overline{0}\rangle$에 적용하면 $|\overline{1}\rangle$을 구할 수 있다.

$$|\overline{1}\rangle = \overline{X}|\overline{0}\rangle = XXXXX|\overline{0}\rangle = \quad (8)$$
$$\frac{1}{4}(|11111\rangle+|01101\rangle+|10110\rangle+|01011\rangle$$
$$+|10101\rangle-|00100\rangle-|11001\rangle-|00111\rangle$$
$$-|00010\rangle-|11100\rangle-|00001\rangle-|10000\rangle$$
$$-|01110\rangle-|10011\rangle-|01000\rangle+|11010\rangle)$$

이를 통해 5 큐비트를 사용하여 1 큐비트의 정보에 발생하는 1개의 오류를 정정할 수 있는

$[[5,1,3]]$ 부호를 구성할 수 있다.

2.2 범용 양자 컴퓨팅 게이트 세트

디지털 시스템에서는 하나의 게이트로 모든 연산들을 표현할 수 있고 이를 범용 게이트(universal gate)라 한다. 하지만 양자 컴퓨팅 시스템에서는 하나의 게이트로 모든 연산들을 표현 할 수가 없으며, 단일 큐비트 게이트와 2 큐비트 이상의 게이트들로 이루어진 범용 양자 컴퓨팅 게이트 세트가 필요하다[22]. 이러한 세트들의 기본적인 예로는 $\{H, T, CNOT\}$, $\{H, Toffoli\}$, $\{H, CCZ\}$ (CCZ : Controlled-Controlled-Z 게이트) 등이 있다. 이러한 세트들의 선택은 양자정보처리에서 어떤 양자오류정정부호를 사용하는지에 따라 결정된다. 주로 논리적 게이트의 형태가 각 큐비트에 동일한 물리적 게이트들을 해주는 형태인 transversal 게이트들이 많은 세트를 고르는 것이 전체 연산 복잡도 측면에서 유리하다. Non-transversal인 게이트의 경우 연산을 위하여 Magic 상태라 불리는 특별한 상태가 필요하며 이러한 non-transversal 게이트를 연산하는데 드는 소요자원은 양자 컴퓨터를 구현하는데 굉장히 큰 부담 중 하나이다.

3. 동형암호

개인정보 보호법 상 모든 개인정보는 암호화 및 접근 통제를 통해 보호하도록 규정되어 있다. 그러나 최근까지도 빈번하게 발생하는 기업 고객 정보 유출, 공공기관 데이터베이스 유출 등을 보면 현재도 데이터베이스 보안이 충분하지 않음을 알 수 있다. 데이터를 암호화하여 저장할 경우, 데이터가 유출이 된다 하더라도 암호키가 유출되지 않는다면 해당 데이터들이 복호화 되어 유출될 가능성은 없다. 그러나 저장중인 데이터를 통계적으로 처리, 키워드 검색 등을 수행하기 위해서는 해당 데이터들을 다시 복호화 해야하고 복호화 작업은 처리 시간이 큰 과정 중 하나이다. 또한 복호화 및 연산 과정에서 데이터가 유출될 가능성이 존재한다. 따라서 데이터를 암호화 한 상태에서 연산 가능한 암호체계의 필요성이 대두되었고, 해당 암호체계가 바로 동형암호이다. 동형암호는 암호화된 상태에서 연산할 수 있으므로 복호화 된 데이터의 유출을 막을 수 있고, 암·복호화에 필요한 시간을 단축하여 검색 속도를 향상시킬 수 있다. 또한, 클라우드 컴퓨팅에서 개인정보가 외부 저장 공간에 전송 및 저장 시 발생할 가능한 보안 문제를 해결할 수 있다. 동형암호 중에서도 암호화된 데이터를 복호화 하지 않고 원하는 횟수만큼 검색 및 연산이 가능한 동형암호가 완전동형암호(fully homomorphic encryption)이다[9].

동형암호에서 동형이라는 용어는 수학에서 사용하는 준동형 사상(homomorphism)에서 온 것으로, 준동형 사상이란 연산이 정의된 두 집합 사이에 연산이 보존되는 맵핑을 뜻한다. 따라서 동형암호는 평문·암호문 공간 사이에 덧셈, 곱셈을 보존하는 암호체계라 할 수 있다. 즉, 평문 상태에서 연산을 한 결과와 암호문 상태에서 연산 후 복호화한 결과가 동일한 암호체계이다. 더 나아가 XOR, AND 등 모든 논리 연산이 보존되는 암호체계가 완전 동형암호이다. 이해를 돕기 위해 RSA 암호시스템을 변형한 형태의 동형암호 기법의 예시는 다음과 같다.

1) 충분히 큰 소수 p와 q를 선택

2) p와 q의 곱을 공개키 n으로 사용

3) 데이터 a를 암호화 :

$$Enc(a) = (a(\mathrm{mod}\ p) + r_1 \times p,\ a(\mathrm{mod}\ q) + r_2 \times q) = (c_1, c_2)\ \ (a \in Z_n)$$

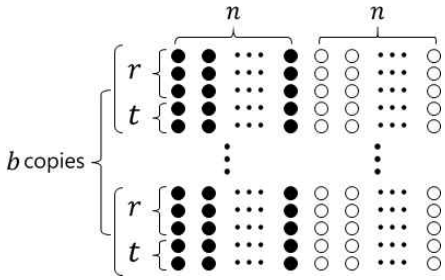
4) 클라우드 컴퓨터쪽에서의 복호화 : p와 q를 알기 때문에 난수 r_1과 r_2 제거 가능

$$Dec(c_1, c_2) = (c_1(\mathrm{mod}\ p), c_2(\mathrm{mod}\ q)) = (a(\mathrm{mod}\ p), a(\mathrm{mod}\ q))$$

임의의 난수 r_1과 r_2를 사용하여 암호화를 하며, 중국인의 나머지 정리를 사용하여 복호화를 할 수 있다.

4. 관련 연구

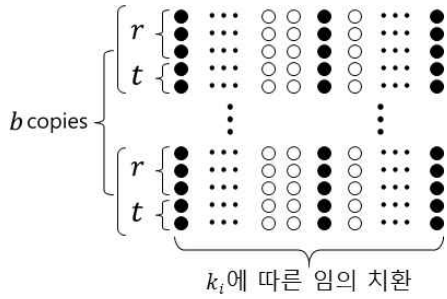
이 장에서는 양자오류정정부호의 논리적 게이트와 물리적 게이트 간의 상관관계를 이용하여 양자동형암호를 구현한 기법 중 한 가지를 설명한다[19]. 해당 기법에서는 입력 상태, 앞서 설명한 범용 양자 계산을 위한 Magic 상태($|T\rangle\langle T| = \frac{I}{2} + \frac{X+Y}{2\sqrt{2}}$), 그리고 보조 큐비트들을 사용한다. 일단 r-큐비트 입력상태와 t-큐비트 개의 Magic 상태를 준비한다. 또한, $(r+t)\times(2n-1)$ 최대혼합상태의 보조 큐비트($\frac{I}{2}$)를 준비한다. 최대혼합상태는 상태로부터 아무런 정보를 뽑아낼 수 없기 때문에 도청자나 클라우드 컴퓨터가 데이터의 위치를 알지 못해 최대혼합상태를 고를 경우 데이터와 관련된 정보를 전혀 얻을 수 없다.



(그림 1) 암호화 이전의 데이터 준비상태

이후 보조 큐비트 중 $(r+t)\times(n-1)$개를 반복 양자오류정정부호 부호화 연산자를 통해 데이터로 부호화한다. 이러한 데이터와 최대혼합상태 보조 큐비트의 복사본을 b개 준비하면 모든 준비가 끝난다. 준비 상태를 그림으로 나타내면 (그림 1)과 같다.

암호화를 위해서 1과 $2n$ 사이에서 n개의 수를 뽑아 임의로 섞는 암호화 키로 사용한다. 즉 암호화 키 $k_i(1 \leqq i \leqq n)$는 $1 \leqq k_1 < \cdots < k_n \leqq 2n$을 만족하며 각 k_i에 따라 최대혼합상태 열을 다른 열들과 섞어줌으로써 보안성을 추가로 확보할 수 있다. 부호화/암호화가 완료된 최종적인 상태는 (그림 2)와 같다.



(그림 2) 부호화/암호화가 완료된 데이터

참고문헌 [19]에서는 이후 해당 기법을 사용하여 부호화/암호화한 서로 다른 임의의 데이터 간의 trace distance를 측정함으로써 안정성을 증명하였다.

참고문헌 [19]에서는 반복 양자오류정정부호를 이용하여 양자동형암호를 구성하였다. 반복 오류정정부호는 디지털 오류정정부호에서도 가장 간단한 부호로 0은 '00…00'으로 1은 '11…11'로 동일한 값을 반복한 것을 코드워드로 사용하는 부호이며 반복 양자오류정정부호는 이를 양자상태에 그대로 적용한 부호이다. 그러나 앞서 2장에서 설명하였듯이 양자오류정정부호의 안정 연산자는 $|0\rangle$ 상태에 대해서만 제대로 동작한다. 보안성을 위해 보조 큐비트로 최대혼합상태인 $\frac{I}{2}$ 상태를 사용하였기 때문에 사용한 부호의 안정 연산자가 제대로 된 역할을 할 수 없다. 따라서 해당 기법은 부호화 연산자는 그대로 사용하였으나 오류정정 능력은 가질 수 없다.

5. 동형암호적 양자 계산이 가능한 양자오류정정부호 기법

5.1 양자오류정정부호와 동형암호의 기본가정

양자오류정정부호와 동형암호 사이에는 유사한 가정이 존재한다. 양자오류정정부호는 논리적 큐비트 수준에서의 연산이 물리적 큐비트 수준에서의 연산과 동일하게 적용 가능해야한다. 동형암호는 암호문 상태에서의 연산이 평문 상태에서의 연산과 동일한 결과를 나와야한다. 이 두 가정이 양

자 계산에 있어서는 매우 유사한 형태로 표현될 수 있다. 두 가정을 조금 변형하면 다음과 같이 설명이 가능하다.

$$U_E^{\dagger}A_l U_E(|\psi\rangle|0\rangle^{\otimes k-1}|0\rangle^{\otimes n-k}) \quad (9)$$
$$=(A_p\otimes I^{\otimes k-1})U_E^{\dagger}U_E(|\psi\rangle|0\rangle^{\otimes n-1})$$

이 때 U_E는 양자오류정정부호의 부호와 연산자이며 A_l, A_p는 논리적/물리적 큐비트 수준에서의 수행하고자 하는 연산, $|\psi\rangle$는 연산에 사용할 데이터, $|0\rangle$는 보조 큐비트이다. 또한, U_E는 유니터리 연산자이므로 $U_E^{\dagger}U_E=I$이다.

수식 (9)의 좌변은 양자오류정정부호를 사용하여 부호화 된 데이터 $U_E(|\psi\rangle|0\rangle^{\otimes k-1}|0\rangle^{\otimes n-k})$에 논리적 연산 A_l을 수행하고 다시 복호화 $U_E^{\dagger}$를 해준 결과를 나타내며, 우변은 부호화하지 않은 데이터 $|\psi\rangle$에 물리적 연산 A_p를 수행한 결과를 나타낸다. 이때 우변에서 $I^{\otimes k-1}$, $|0\rangle^{\otimes n-1}$은 연산의 차원을 좌변과 맞춰주기 위해 존재한다. 즉, 수식 (9)는 부호화 후에 논리적 연산 A_l을 수행한 결과와 부호화하지 않은 데이터에 수행한 물리적 연산 A_p를 수행한 것과 등가여야 함을 나타낸다.

(9)번 수식에서 우변이 기본적으로 수행하고자하는 연산이라고 할 때, 동형암호 측면에서의 수식은 다음과 같이 표현할 수 있다.

$$(A_p\otimes I^{\otimes k-1})(|\psi\rangle|0\rangle^{\otimes n-1}) \quad (10)$$
$$=U_R^{\dagger}A_H U_R(|\psi\rangle|0\rangle^{\otimes n-1})$$

이 때 U_R은 동형암호적 연산을 위한 암호화 연산자이다. 수식 (10)은 암호화하지 않은 상태에서의 연산 A_p를 수행한 결과와 암호화 한 상태에서의 연산 A_H을 수행하고 복호화한 결과가 동일해야함을 나타낸다. 따라서 수식 (9)와 (10)은 적절한 변형을 통해 하나로 합쳐질 수 있으며 U_R을 잘 구성할 경우 양자오류정정부호를 사용함으로써 동형암호적 양자 계산이 가능함을 알 수 있다.

5.2 제안하는 변형된 양자오류정정부호 기법

앞서 2.1에서 설명한 것처럼 양자 정보는 복제가 불가능하기 때문에 $[[n,k,d]]$ 양자오류정정부호를 사용하기 위해서는 $n-k$ 큐비트의 기저 상태 $|0\rangle^{\otimes n-k}$를 준비해야한다. 이 때, 동형암호적 양자 계산이 가능하도록 변형이 필요하기 때문에 k 큐비트의 정보에서 $k-1$ 큐비트는 더미 데이터로 사용하며, 실제 연산에 사용할 데이터 $|\psi\rangle$는 그 사이에 임의의 위치 k_i에 삽입한다. 즉 양자오류정정부호를 사용하기 전 초기 상태는 다음과 같다.

$$|D_1\rangle=|\underbrace{0\ldots\psi\ldots 0}_{k}\rangle|0\rangle^{\otimes n-k} \quad (11)$$

앞서 설명하였듯이 양자오류정정부호는 보안성은 없기 때문에, 이를 위해 더미 데이터들을 랜덤 인코딩 유니터리 연산자들로 랜덤 인코딩을 한다.

$$|D_2\rangle=U_R\otimes I^{n-k}|\underbrace{0\ldots\psi\ldots 0}_{k}\rangle|0\rangle^{\otimes n-k} \quad (12)$$

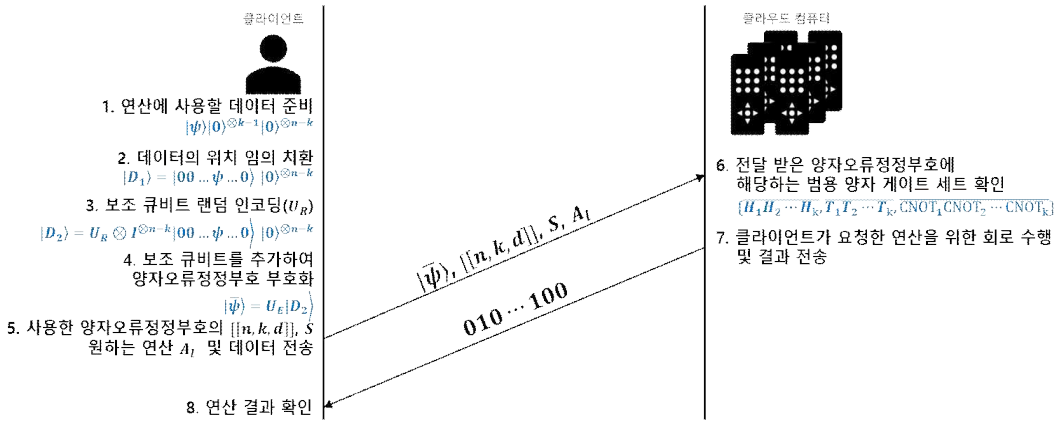
이때 랜덤 인코딩 연산자 U_R은 $U_R=U_{R_1}\otimes U_{R_2}\otimes\cdots I\cdots\otimes U_{R_k}$로 이루어져있다. 이를 통해 공격자는 어느 위치에 데이터 큐비트가 존재하는지를 알 수 없으며 추가적으로 더미 데이터들이 랜덤한 값으로 인코딩 되었기 때문에 부호화를 통해 보조 큐비트로 분산된 정보의 양을 줄일 수 있다.

$|D_2\rangle$를 데이터로 사용하여 $[[n,k,d]]$ 부호로 부호화한다. 사용할 수 있는 $[[n,k,d]]$ 부호는 어느 부호이든 관계없으나 본 논문에서는 범용 게이트 세트 중 T 게이트가 non-transversal한 부호를 고려하겠다.

$$|\overline{\psi}\rangle=U_E|D_2\rangle \quad (13)$$

이때 U_E는 사용하고자하는 $[[n,k,d]]$ 부호의 부호화 연산자이다.

$[[n,k,d]]$ 부호로 부호화하였기 때문에 범용 양자 계산을 위해서는 해당 부호의 범용 게이트 세트를 사용하면 된다. transversal 게이트의 경우 문제가 없으나, non-transversal 게이트의 경우 게이트 teleportation을 통해 연산하도록 한다. 제안하는 기법의 경우 복호화 과정의 복잡도에 수행하려는 연산이 영향을 미치지 않기 때문에 compactness[23]는 일부 확보하였으나, 양자오류정정부호 자체가 non-transversal 게이트 연산의 소요 자원에 있어 굉장히 큰 문제를 보이기 때문에 관련 문제에 있어 자유롭지는 못하다.



(그림 3) 동형암호적 양자 계산이 가능한 양자오류정정부호 기법의 동작 과정 및 단계

k 큐비트 전체가 데이터 큐비트가 아니기 때문에 각 k개씩 논리적 연산에서 모든 큐비트에 해당 연산을 해주는 경우를 골라서 사용한다. 예를 들어 비트 플립을 해주려는 경우 해당 부호에서 $\overline{X_1X_2\cdots X_k}$에 해당하는 연산을 취해주면 된다.

즉, 수식 (11), (12)에서 데이터의 임의 치환 위치 k_i와 더미 데이터 랜덤 인코딩 연산자 U_R이 암호화 키라고 할 수 있다. 해당 정보들은 클라우드 컴퓨팅 과정에서 밖으로 유출되지 않기 때문에 공격자나 클라우드 컴퓨터는 암호화된 데이터 $|\overline{\psi}\rangle$로부터 실제 데이터를 얻을 수 없다. 클라우드 컴퓨터는 어떤 데이터가 어디에 있는지 전혀 알지 못하는 상태로 통보받은 부호의 스펙에 따라 범용 게이트 세트의 연산자들을 사용하여 연산을 수행할 수 있으며 이에 대한 data privacy[23] 안정성은 이후에 서술하겠다. 클라이언트 또한 연산 결과인 비트열 외의 정보는 클라우드 컴퓨터로부터 제공받지 못한다. 클라우드 컴퓨터쪽에서 어떤 회로를 사용하여 결과를 도출했는지 전혀 알 수 없이 결과만 돌려받는다. 따라서 제안하는 기법은 circuit privacy[24]를 만족할 수 있다.

랜덤 유니터리 연산자를 통한 랜덤 더미 데이터와 임의의 위치에 데이터를 삽입함으로써 얻어지는 동형암호적 계산의 data privacy는 서로 다른 코드워드 간의 fidelity를 통해 알아볼 수 있다.

$$F(\rho,\sigma) = \left|\langle\overline{\psi_\rho}|\overline{\psi_\sigma}\rangle\right|^2 \tag{14}$$

<표 1> 제안하는 기법을 통한 클라우드 양자 컴퓨팅에 사용되는 정보의 공개 범위

소유자	정보
클라이언트	k_i, U_R
클라우드 컴퓨터	연산 회로
공개	양자오류정정부호(U_E, $[[n,k,d]]$, S), 해당 부호의 범용 게이트 세트

일반적으로 density operator의 형태를 사용하기 때문에 안정성은 참고문헌 [19]에서와 같이 trace distance를 사용지만 제안하는 기법의 경우 양자계산에 사용하는 코드워드를 가정하였기 때문에 입력 데이터의 형태를 $|0\rangle$, $|1\rangle$, $|+\rangle$, $|-\rangle$로 볼 수 있다. 따라서 양자오류정정부호 부호화 이후에 각각의 코드워드는 pure state이기 때문에 수식 (14)와 같은 간략화 된 형태로 정리할 수 있다. 또한, 서로 동일한 부호를 사용하였기 때문에 각 코드워드의 부호화 연산자 U_E는 동일하므로 수식 (14)는 다음과 같이 정리된다.

$$\left|\langle\overline{\psi_\rho}|\overline{\psi_\sigma}\rangle\right|^2 = \left|\langle D_{2,\rho}|U_E^\dagger U_E|D_{2,\sigma}\rangle\right|^2 = \left|\langle D_{2,\rho}|D_{2,\sigma}\rangle\right|^2 \tag{15}$$

각 $|D_2\rangle$는 서로 다른 랜덤 유니터리 연산자를 통해 랜덤 인코딩되었으므로 수식 (15)는 다음과 같다.

$$\left|\langle D_{2,\rho}|D_{2,\sigma}\rangle\right|^2 = \langle 0|^{\otimes n}U_{R,\rho}^\dagger U_{R,\sigma}|0\rangle^{\otimes n} \tag{16}$$

이때 U_R의 경우 앞서 설명에서 사용된 U_R과는 다르게 데이터 큐비트 $|\psi\rangle$를 인코딩하는데 사용

된 연산자까지를 포함한다. 각각의 랜덤 유니터리 연산자는 $U_{R=}\overline{n}\cdot\overline{\sigma}$로 나타낼 수 있다. 이때 $\overline{n}=(n_1,n_2,n_3)$ 형태의 실수 벡터이며 $\overline{\sigma}=(\sigma_x,\sigma_y,\sigma_z)$로 $\sigma_x=X$, $\sigma_y=Y$, $\sigma_z=Z$이다. 이를 수식 (16)에 대입하면 다음과 같이 정리될 수 있다.

$$\langle 0|^{\otimes n}U_{R,\rho}^{\dagger}U_{R,\sigma}|0\rangle^{\otimes n}=(\overline{n_\rho}\cdot\overline{n_\sigma})^2 \quad (17)$$
$$+(n_{\rho,1}n_{\sigma,2}-n_{\rho,2}n_{\sigma,1})^2$$

따라서 해당 fidelity 값이 최대한 큰 랜덤 유니터리 연산자들을 사용하면 서로 다른 데이터를 부호화 및 암호화한 코드워드이지만 외부에서는 구분하기 어려운 보안성을 확보할 수 있다.

5. 결 론

본 논문에서는 양자오류정정부호만을 이용하여 동형암호적 양자 계산이 가능한 기법을 제시하였다. 제안한 기법은 $[[n,k,d]]$ 양자오류정정부호에 있어 k 큐비트의 데이터 큐비트 중 $k-1$ 큐비트를 랜덤 인코딩 된 더미 데이터로 사용하고 1 큐비트의 실제 데이터를 임의의 위치에 삽입하여 클라우드 컴퓨터쪽에서 데이터에 대한 정보를 얻을 수 없도록 변형하였다. 양자오류정정부호가 가지는 범용 양자 게이트 세트를 통해 transversal한 게이트의 경우 간단하게 논리적 연산이 가능하며 non-tranversal 게이트의 경우 게이트 teleportation을 통해 범용 양자 계산이 가능하다.

최대혼합상태를 보조 큐비트를 사용하여 오류정정 능력은 상실했던 기존 기법과는 다르게, 데이터 큐비트를 랜덤 인코딩하여 사용하였기 때문에 양자오류정정부호의 부호화만을 사용하여 부호 고유의 특성인 오류정정 능력을 그대로 가지면서 동시에 동형암호의 보안성을 확보할 수 있도록 하였다. 향후에는 보안성 분석을 강화하여 정보이론적 보안을 확보할 수 있는지에 대한 검토 및 보안성 확보를 위한 k 값의 하한 등에 대한 분석이 필요하다.

참고문헌

[1] Richard P. Feynman, “Simulating Physics with Computers,” International Journal of Theoretical Physics, VoL 21, Nos. 6/7, 1982.

[2] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," Phys. Rev. A 52, R2493. 1995.

[3] Wootters, William; Zurek, Wojciech , “A Single Quantum Cannot be Cloned,” Nature vol. 299 pp. 802-803, Oct. 1982.

[4] A. R. Calderbank and P. W. Shor., “Good quantum error-correcting codes exist,” Phys. Rev. A, 54:1098, 1996.

[5] A. M. Steane., “Multiple particle interference and quantum error correction,” Proc. R. Soc. London A, 452:2551-2577, 1996.

[6] Vuillot, Christophe and Asasi, Hamed and Wang, Yang and Pryadko, Leonid P. and Terhal, Barbara M., “Quantum error correction with the toric Gottesman-Kitaev-Preskill code,” Phys. Rev. A, 99, 3, 032344, 2019.

[7] Layden, David and Zhou, Sisi and Cappellaro, Paola and Jiang, Liang, “Ancilla-Free Quantum Error Correction Codes for Quantum Metrology,” Phys. Rev. Lett., 122, 4, 040502, 2019

[8] Viyuela, Oscar and Vijay, Sagar and Fu, Liang, “Scalable fermionic error correction in Majorana surface codes,” Phys. Rev. B, 99, 20 ,205114, 2019

[9] IBM Quantum Experience, http://www.research.ibm.com/quantum.

[10] Harper, Robin and Flammia, Steven T., “Fault-Tolerant Logical Gates in the IBM Quantum Experience,” Phys. Rev. Lett., 122, 8, 080504, 2019

[11] Behera, B.K., Reza, T., Gupta, A. et al, “Designing quantum router in IBM quantum computer,” Quantum Inf. Process., 18, 328, 2019

[12] M. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali, “Cloud computing: Distributed in-

ternet computing for it and scientific research," Internet Computing, IEEE, 13(5):10-13, 2009.

[13] Ron Rivest, Leonard Adleman, and Michael L. Dertouzos., "On data banks and privacy homomorphisms," In Foundations of Secure Computation, pages 169.180, 1978.

[14] Craig Gentry. "Fully homomorphic encryption using ideal lattices," In Michael Mitzenmacher, editor, STOC, pages 169.178. ACM, 2009.

[15] A. Chatterjee and I. Sengupta, "Translating Algorithms to Handle Fully Homomorphic Encrypted Data on the Cloud," in IEEE Transactions on Cloud Computing, vol. 6, no. 1, pp. 287-300, 1, 2018.

[16] M. Liang, "Symmetric quantum fully homomorphic encryption with perfect security," Quantum Inf. Process., vol. 12, no. 12, pp. 3675-3687, 2013.

[17] C.-Y. Lai and K.-M. Chung, "On statistically-secure quantum homomorphic encryption," Quantum Inf. Comput., 18, 785-794, 2018

[18] M. Newman and Y. Shi, "Limitations on transversal computation through quantum homomorphic encryption," Quantum Inf. Comput., Vol. 18, No. 11&12, pp. 0927-0948, 2018

[19] Ouyang, Yingkai and Tan, Si-Hui and Fitzsimons, Joseph F., "Quantum homomorphic encryption from quantum codes", Phys. Rev. A, 98, 4, 042334, 2018.

[20] Chen, X.-B., Sun, Y.-R., Xu, G., Yang, Y.-X.: Quantum homomorphic encryption scheme with flexible number of evaluator based on (k, n)-threshold quantum state sharing. Inf. Sci. 501(10), 172-181, 2019.

[21] D. Gottesman, "Stabilizer codes and quantum error correction," Ph.D. dissertation, California Institute of Technology, 1997.

[22] Jean-Luc Brylinski, Ranee Brylinski, "Universal Quantum Gates", arXiv:quant-ph/0108062v1, 2001.

[23] Craig Gentry, "A FULLY HOMOMORPHIC ENCRYPTION SCHEME", Ph.D. dissertation, STANFORD UNIVERSITY, 2009.

[24] EL-YAHYAOUI, A.; ECH-CHERIF EL KETTANI, M.D. A Verifiable Fully Homomorphic Encryption Scheme for Cloud Computing Security. Technologies 7, 21, 2019.

〔저자소개〕

손 일 권 (IlKwon Sohn)
2018년 고려대학교 공과대학 전기전자전파공학부 박사
2019년 ~ 현재 한국과학기술정보연구원 박사후연구원
email : d2estiny@kisti.re.kr

이 종 현 (Jonghyun Lee)
2015년 ~ 현재 고려대학교 공과대학 전기전자공학부 석박사통합과정
email : ljh0523@korea.ac.kr

이 원 혁 (Wonhyuk Lee)
2003년 성균관대학교 공과대학 컴퓨터공학과 석사
2010년 성균관대학교 공과대학 전자전기컴퓨터공학과 박사
2003년 ~ 현재 한국과학기술정보연구원 선임연구원
email :livezone@kisti.re.kr

석 우 진 (Woojin Seok)
2002년 University of North Carolina at Chapel Hill, Computer Science MS
2008년 충남대학교 공과대학 컴퓨터공학과 박사
2003년 ~ 현재 한국과학기술정보연구원 책임연구원

email : wjseok@kisti.re.kr

허 준 (Jun Heo)
1991년 서울대학교 공과대학 전자공학과 석사
2002년 University of Southern California Electrical Engineering PhD
2007년 ~ 현재 고려대학교 전기전자공학부 교수
email : junheo@korea.ac.kr