정 보 보 호 학 회 지
제 33 권 제1호, 2023. 2

격자 기반 양자내성암호 양자 분석 기술

전 찬 호*, 허 동 회*, 이 명 훈*, 이 창 원*, 최 영 록*, 천 병 호*, 김 수 리**

요 약

Peter Shor의 양자 컴퓨터를 이용하여 다항식 시간 안에 인수분해 문제를 해결할 수 있는 알고리즘이 제안됨에 따라, 다양한 수학적 난제를 기반으로 한 새로운 양자내성암호에 대한 연구가 진행되었다. 현재 가장 많이 연구되고 있으며 유망한 양자내성암호는 격자 기반 암호이고, 격자 기반 암호의 안전성을 입증하기 위해 격자 암호 분석 알고리즘들 또한 함께 연구되고 있다. 본 논문에서는 격자 암호 분석 알고리즘 및 양자 탐색 알고리즘을 적용한 격자 암호 양자 분석 기술들에 대해서 설명한다.

Ⅰ. 서 론

RSA, DSA, ECC(타원곡선암호)와 같은 공개키 암호 알고리즘은 서명, 키 교환 등과 같은 역할을 하는 알고리즘으로서 현재 통신계열, 금융계열 등에서 가장 많이 활용되는 중요 핵심 기술이다. 각 암호의 안전성은 각각 소인수분해 문제, 이산대수 문제, 타원곡선 이산대수 문제 등의 어려움에 기반하고 있으며, 문제를 해결하는 데에는 하지수 시간이 필요한 것으로 알려져 있었다.

그러나 1994년에 Peter Shor에 의하여[1], 양자 컴퓨터를 이용하였을 때 다항식 시간 안에 이산대수/인수분해 문제를 해결할 수 있는 알고리즘이 제안되었다. 이를 Shor's Algorithm이라 하며, 양자 컴퓨팅 환경에서는 Shor's Algorithm에 의해 위의 RSA, DSA, ECC 등의 고전 암호 알고리즘들이 안전하지 않다는 사실이 밝혀졌다.

그 이후 아직 양자 컴퓨터로 해결할 수 없는 문제에 기반하여 공개키 암호들이 제안되기 시작하였으며, 이러한 암호들을 양자내성암호(post-quantum cryptography, PQC)라고 한다. 양자내성암호는 다양한 수학적 난제들을 기반으로 하고 있으며, 2017년 NIST는 양자내성암호 표준화 공모전을 시작하였다. 표준화 공모전 round 3에서 Kyber[2], Dilithium[3], FALCON[4], SPHINCS+[5] 가 표준화 대상 암호로 선정되었으며, 이 중 3개가 격자 문제를 기반으로 하는 암호로써 현재 가장 각광을 받고 있는 암호는 격자 기반 암호라고 할 수 있다.

격자 기반 암호가 연구됨에 따라 격자 기반 암호의 안전성을 입증하기 위한 연구 또한 함께 진행되고 있다. 격자 기반 암호의 안전 강도를 올바르게 분석하기 위해서는 격자 암호의 비밀키를 공격할 수 있는 알고리즘에 대한 연구가 수반되어야 하며, 일반 컴퓨터뿐만 아니라 양자 컴퓨터를 활용한 양자 알고리즘에 대해서도 어느 정도의 계산 복잡도로 격자 암호 분석을 수행할 수 있는지에 대해 고려가 되어야 한다.

본 논문에서는 일반 컴퓨팅 환경에서 격자 암호를 분석하는 알고리즘들에 대해서 설명하고, 양자 탐색 알고리즘을 적용할 수 있는 양자 컴퓨팅 환경에서 격자 기반 암호를 분석하는 양자 분석 기술들에 대하여 설명한다.

Ⅱ. 배경 지식

2.1. 격자 기반 암호

$\mathbb{R}^m$위의 벡터들에 대해서 $\{b_1, b_2, ..., b_n\}, n \leq m$을 만족하는 독립 벡터들의 집합 B를 가정한다. 이때, B에 의해 만들어지는 격자 공간은

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2022R1F1A1063611)
* 고려대학교 정보보호학과 암호알고리즘 연구실 (대학원생, cksgh419@korea.ac.kr, dong5641@korea.ac.kr, ope2527@korea.ac.kr, lchwon@korea.ac.kr, dudfhrchl@korea.ac.kr, cheonbh0819@korea.ac.kr)
** 성신여자대학교 수리통계데이터사이언스학부 (조교수, suhrikim@gmail.com)

$$L = \left\{ \sum_i c_i b_i \mid c_i \in \mathbb{Z} \right\}$$

로 표현하고, B를 L의 기저라고 하며, 동일한 격자 공간 L을 표현하는 기저는 다양하게 존재할 수 있다.

2.2. 격자 기반 암호 문제

격자 기반 암호는 다양한 기반 문제를 가지고 있지만, 가장 기본이 되는 기반 문제는 Shortest Vector Problem(SVP)이다. SVP는 주어진 격자 공간 L 위에서 가장 짧은 길이를 가지는 벡터를 찾는 문제이다. 이 외에도 많은 격자 기반 문제들이 존재하지만, 모두 SVP를 푸는 방향으로 변환될 수 있다. 따라서 현재 SVP를 푸는 알고리즘에 대해서 많은 연구가 진행되고 있으며, 해당 알고리즘들에 양자 탐색 알고리즘을 적용하는 방법 또한 함께 연구되고 있다.

2.3. Grover 탐색 알고리즘

Grover 탐색 알고리즘은 1996년 Lov Grover가 제안한 탐색 알고리즘으로써[6], 양자 컴퓨터 상에서 중첩상태와 얽힘 현상을 이용하여 주어진 데이터 집합 $\{d_1, d_2, \ldots, d_n\}$내에 원하는 특성을 가진 데이터 d_i들을 $O(\sqrt{n})$의 계산 복잡도로 찾아내는 알고리즘이다. Grover 알고리즘은 탐색 대상인 데이터 집합의 모든 원소가 동일한 확률로 중첩된 상태로 진행되며, 원소들이 중첩되도록 변환시킬 때는 Hadamard 게이트가 사용된다. 그 후 특정 원소의 상태에 대해서만 부호를 반전시키는 게이트와 특정 원소의 상태들의 진폭을 증폭시키는 게이트를 하나의 반복 단위로 설정하여, $\frac{\pi}{4}\sqrt{\frac{n}{t}}$ (t : 탐색하려는 원소의 개수) 번 만큼 반복 수행한다. 이때 찾고자 하는 원소들이 확률 합은 1에 가까우므로 측정(measurement)을 하여 원하는 원소들을 찾을 수 있다.

2.4. 양자 무작위 행보 알고리즘

양자 무작위 행보는 무작위 행보(random walk)를 양자 컴퓨터상에서 수행할 수 있도록 구현한 알고리즘이다. 무작위 행보는 다음과 같이

1. 처음 시작 지점인 v를 설정(Setup)
2. 임의의 인접한 다른 지점으로 이동(Update)
3. 도착 지점이 목표 지점인지 확인(Check)

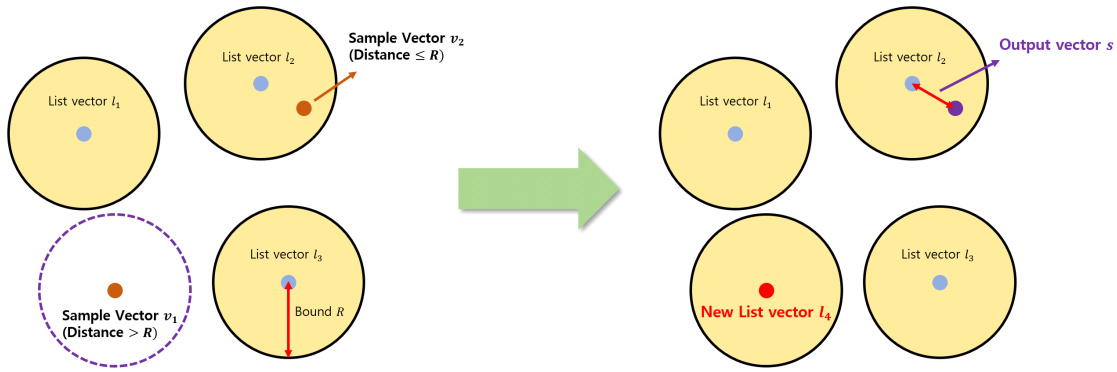
총 3단계로 구성되어 있다. 양자 무작위 행보에서는 update에 해당하는 부분을 unitary operation, check에 해당하는 부분을 phase estimation으로 대체하여 수행한다[7]. Phase estimation을 수행하는 양자 게이트는 unitary operation을 수행하는 게이트를 이용하여 설계한다. 임의의 양자 상태 $|r\rangle$에 대하여 반복적인 phase estimation을 수행하여 eigenvalue를 측정하였을 때 1이 나온 횟수가 전체 phase estimation 수행 횟수의 $\frac{3}{8}$을 넘어가게 된다면, $|r\rangle$부터 시작한 무작위 행보에서 목표 지점이 존재하고 찾을 수 있다는 것을 알 수 있다. 반대로 $\frac{3}{8}$을 넘지 못한다면 $|r\rangle$부터 시작하는 무작위 행보에서는 목표 지점이 존재하지 않고 찾을 수 없다. 탐색해야 하는 지점의 개수 또는 그 상한을 T, 알고리즘의 수행이 실패할 확률이 δ, unitary operation을 수행해야 하는 횟수를 n이라 할 때, 알고리즘의 계산복잡도는 $O(\sqrt{Tn}log(1/\delta))$로 표현된다.

Ⅲ. 격자 암호 분석 알고리즘

격자 기반 양자내성암호의 양자 분석 기술들은 기존의 격자 암호 분석 알고리즘의 탐색 부분을 양자 탐색 부분으로 전환하여 계산 복잡도를 줄이는 방식을 사용하고 있다. 따라서 격자 암호 양자 분석 기술의 이해를 위해서는, 기존의 컴퓨팅 환경에서 사용되던 알고리즘을 이해하여야 한다. 이 장에서는 격자 암호 분석 알고리즘에 관해서 서술한다.

3.1. Sieving 알고리즘

Sieving 알고리즘은 여러 벡터들을 샘플링한 후에, 벡터간에 거리를 확인하면서 뺄셈을 반복하는 과정을 통해 최단 벡터를 찾는 알고리즘이다. Sieving 알고리즘에는 여러 알고리즘들이 존재하지만[8, 9, 10, 11, 12], 기본적으로 리스트 내의 벡터들을 이용하여 샘플링된 벡터들을 새로운 리스트 벡터로 변환하거나 짧은 길이의 새로운 벡터를 만드는 작업을 수행한다. 효율적인 알고리즘의 수행을 위해서 최근접 이웃 탐색 기법을 도입한 방식도 존재하며, 해당 기법을



〔그림 1〕 Sieving 알고리즘 도식도

도입하기 위해 locality-sensitive filtering(LSF)라는 필터링 함수를 사용한 Sieving 알고리즘도 제안되었다[13]. 샘플링 벡터 및 리스트 벡터들을 저장하기 위해 지수 공간만큼의 공간복잡도가 필요하다는 단점이 존재한다.

3.2. Enumeration 알고리즘

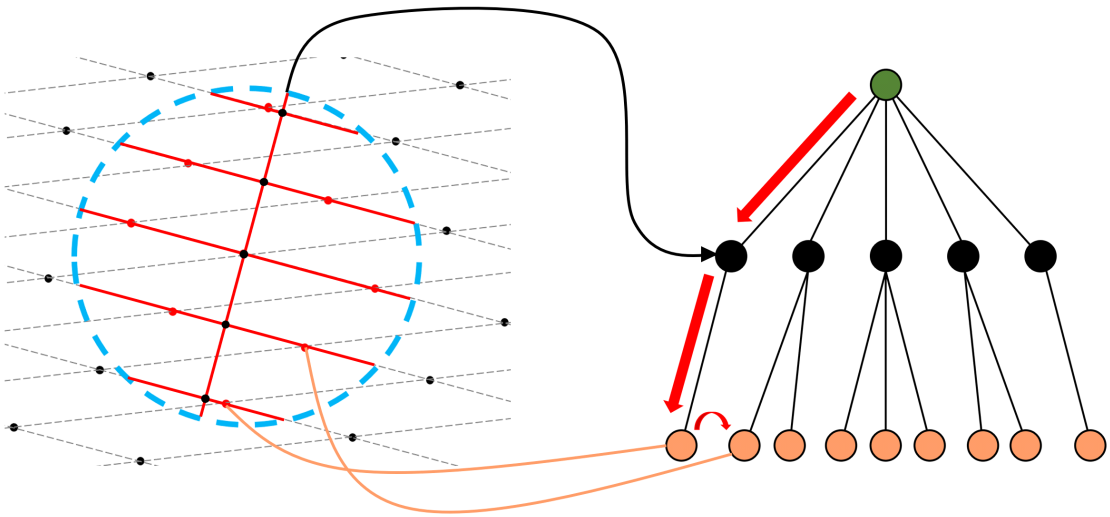
Enumeration 알고리즘은 최단 벡터를 찾는 알고리즘으로써, 주어진 범위 내에 있는 모든 격자 벡터를 탐색하는 방식을 사용하는 알고리즘이다[14]. 일반적인 격자 벡터 v는 격자 기저 벡터 집합 $B=\{b_1, b_2, \ldots, b_n\}$를 이용하여 다음과 같이 표현할 수 있다.

$$v = c_1 b_1 + c_2 b_2 + \ldots + c_n b_n \, (c_1, c_2, \ldots, c_n \in \mathbb{Z})$$

그람-슈미트 직교화 과정을 거쳐서 만들어진 새로운 직교 기저 벡터 집합 $B' = \{b_1^*, b_2^*, \ldots, b_n^*\}$에 의해서 다음과 같이 새로 벡터를 표현할 수 있다.

$$v = \sum_{j=1}^{n} \left(c_j + \sum_{i=j+1}^{n} \mu_{i,j} c_i \right) b_j^*$$

모든 b_i^*들끼리는 각각 수직하므로, 벡터 v의 크기



〔그림 2〕 Enumeration 알고리즘 도식도

(norm)은 $\| v \| = \sqrt{\sum_{j=1}^{n}\left(c_j + \sum_{i=j+1}^{n} \mu_{i,j} c_i\right)^2 \| b_j^* \|^2}$ 와 같이 식으로 나타낼 수 있으며, 이 크기가 주어진 범위 R 보다 작아야 한다.

$\sum_{j=1}^{n}\left(c_j + \sum_{i=j+1}^{n} \mu_{i,j} c_i\right)^2 \| b_j^* \|^2 \leq R^2$ 을 만족하도록 c_i들을 정하기 위해서는 b_n^* 방향에서 시작하여 b_1^* 방향까지 역순으로 계수의 범위를 계산하여 가능한 c_i의 값들을 계산하는 방식을 사용한다.

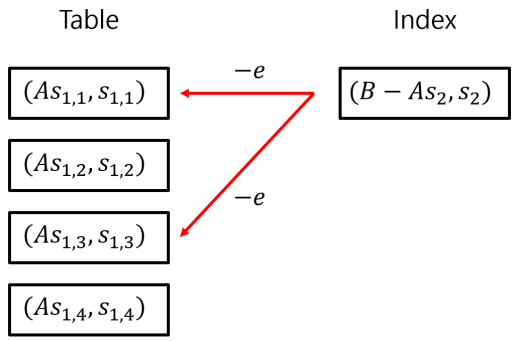
$$\left| c_{n+1-k} + \sum_{i=n+2-k}^{n} \mu_{i,j} c_i \right|$$

$$\leq \frac{\sqrt{R^2 - \sum_{j=n+2-k}^{n}\left(c_j + \sum_{i=j+1}^{n} \mu_{i,j} c_i\right)^2 \| b_j^* \|^2}}{\| b_{n+1-k}^* \|}$$

위 식과 같은 방식으로 b_{n+1-k}^* 방향 값의 범위를 계산할 수 있으며, 앞서 정해진 $c_{n+2-k}, \ldots, c_n$의 값들을 이용하여 c_{n+1-k}의 범위를 결정할 수 있다. 이후 $c_{n+1-k}, \ldots, c_n$를 이용하여 c_{n-k}의 값의 범위를 결정하게 된다. c_{n+1-k}의 가능한 범위 내에서 어떤 값을 선택하냐에 따라 다음 계수의 범위가 달라질 수 있으므로, 가능한 범위 내의 모든 경우에 대해서 탐색을 진행해야 하며, 이 형태는 나무 형태에서 되추적을 진행하는 것과 같다. 되추적을 진행하여 가능한 모든 $(c_1, c_2, \ldots, c_n)$의 길이를 계산하여 가장 짧은 길이의 벡터 v를 찾는다.

3.3. 중간일치 공격(Meet-In-The-Middle) 알고리즘

격자 암호 알고리즘에서의 중간일치 공격 알고리즘은 비밀키에 해당하는 값을 2개 값의 합으로 표현되도록 나누고, 1개 부분에 대한 값을 추측하여 올바른 비밀키를 탐색하는 방식의 알고리즘이다. NTRU 계열의 암호에서는 공개키 다항식 h와 비밀키 다항식 f, g간에 $h = \frac{g}{f}$라는 식이 성립하므로, $h = \frac{g}{f} = \frac{g}{f_1 + f_2}$로 나누어 $hf_1 = g - hf_2$의 식을 유도한다[15]. LWE 계열에서는 공개키 A, B와 비밀키 s, 에러 e에 대해서



〔그림 3〕 중간일치 공격 알고리즘 도식도

$B = As + e$의 식이 성립하므로, $As_1 = B - As_2 - e$와 같이 나누는 식을 도출할 수 있다[16]. 이후 f_1이나 s_1의 가능한 모든 경우에 대해서 hf_1 또는 As_1의 값을 계산하여 표의 형태로 저장한다. 이후 나머지 비밀키 부분에 대하여 $-hf_2$ 또는 $B - As_2$값의 형태로 계산을 진행한 후에 기존의 표와 비교하여 올바른 비밀키 쌍을 찾아내는 것이 중간일치 공격 알고리즘의 방식이다. 중간일치 공격은 표를 저장하기 위해 공간복잡도를 많이 사용한다는 단점이 존재한다.

Ⅳ. 양자 탐색 알고리즘을 적용한 격자 암호 분석 기술

양자 탐색 알고리즘은 앞서 배경지식에서 설명하였듯이, Grover 탐색 알고리즘과 양자 무작위 행보 알고리즘을 이용한 방식 2가지가 알려져 있다. 각각의 알고리즘은 그 특징에 따라 적용할 수 있는 상황이 다르므로, 이 장에서는 각각의 탐색 알고리즘이 격자 암호 분석 알고리즘에 어떻게 적용되는지를 설명한다.

4.1. Grover 알고리즘을 이용한 격자 암호 분석 기술

4.1.1. 양자 Sieving 알고리즘

Sieving에서 Grover 탐색 알고리즘을 적용하는 곳은 입력 벡터들에 뺄셈 또는 덧셈을 수행할 리스트 내의 벡터들을 찾는 과정을 수행하는 과정이다. 리스트 L 내에서 다음 조건을 만족하는 n차원 벡터 w들은 입력 벡터 v와 뺄셈 또는 덧셈을 수행한다.

$$w \in L : \| v \pm w \| < \left(1 - \frac{1}{n}\right) \cdot \| v \|$$

해당 조건을 만족하는 리스트 내의 벡터를 찾는 부분의 계산 복잡도는 리스트 내의 벡터들의 개수에 영향을 받으므로 고전 컴퓨팅 환경에서는 $O(|L|)$로 표현할 수 있다. 그러나 Grover 탐색 알고리즘을 적용하였을 때 리스트 내의 벡터를 찾는 계산 복잡도는 $O(|\sqrt{L}|)$로 감소한다. 양자 컴퓨터 위에서 Grover 탐색 알고리즘을 적용하였을 때 변화되는 Sieving 알고리즘의 계산 복잡도는 [표 1][17]과 같이 표현된다.

4.1.2. 양자 중간일치 공격 알고리즘

NTRU 계열과 같이 고정된 해밍 웨이트를 사용하는 격자 암호에서는 중간일치 공격에 Grover 탐색 알고리즘을 적용하여 공격에 필요한 계산 복잡도를 감소시킬 수 있다[18]. 고정된 해밍 웨이트의 f_2 또는 s_2의 값을 추측하여 테이블과 비교하며 탐색하는 부분에서, 입력 다항식을 대응되는 수로 변화시킨 후 Grover 알고리즘을 이용하여 원하는 수를 탐색한다. 전체의 비밀키가 n차 다항식이고 고정 해밍 웨이트가 d라고 가정한다면, f_2 또는 s_2의 경우에는 $\frac{n}{2}$차 다항식이며, 고정 해밍 웨이트는 $\frac{d}{2}$의 값으로 결정된다. 다항식에서 계수가 1인 위치를 각각 $i_1, i_2, \ldots, i_{d/2}$번째로 정의할 때, 다항식 벡터 v에 대응되는 수 I_v는 $I_v = 1 + \binom{i_1}{1} + \binom{i_2}{2} + \ldots + \binom{i_{d/2}}{d/2}$와 같이 계산하는 것으로 정의한다. 이때 I_v가 가질 수 있는 최댓값은

$$1 + \binom{1+(n-d)/2}{1} + \binom{2+(n-d)/2}{2} + \ldots + \binom{n/2}{d/2} = \binom{n/2+1}{d/2}$$

의 식을 통하여 $\binom{n/2+1}{d/2}$임을 알 수 있고 $t = \lceil \log_2 \binom{n/2+1}{d/2} \rceil$ 큐빗만큼 할당하여 I_v의 값들이 중첩될 수 있도록 한다. Grover 탐색 알고리즘에서의 오라클 함수는 특정 조건을 만족하는 다항식 벡터 v에 대응하는 I_v 값이 입력될 때만 1의 값을 출력하고, 그 외에는 0을 출력하는 함수를 사용한다. 위와 같이 설정하여 Grover 알고리즘을 수행하였을 때 계산 복잡도는 $O(\sqrt{2^t}) = O\left(\sqrt{\binom{n/2+1}{d/2}}\right)$이다.

4.2. 양자 무작위 행보를 이용한 격자 암호 분석 기술

4.2.1. 양자 Sieving 알고리즘

최근접 이웃 탐색을 이용하는 Sieving 알고리즘은 locality-sensitive filtering(LSF)라는 필터링 함수를 사용하여 전체 과정을 수행한다. 필터링 함수는 임의의 벡터가 필터로 선택된 필터 벡터와 이루는 각도를 확인하여 근접 여부를 결정하는 함수이다. Sieving 알고리즘에서는 샘플 벡터와 가까운 필터 벡터를 찾고, 해당 필터에 샘플 벡터를 저장하는 방식으로 필터링 함수를 이용한다. 같은 필터에 저장된 샘플 벡터들끼리는 이루는 각도가 작으므로 가깝다고 판단할 수 있다. 따라서 같은 필터에 저장된 샘플 벡터들끼리의 뺄셈을 통해 짧은 길이의 벡터를 구하는 방법이 최근접 이웃 탐색을 적용한 Sieving 알고리즘의 방식이다. 양자 무

〔표 1〕 Sieving 알고리즘 계산 복잡도 비교 표

Algorithm		Classical Computer $\log_2(Time)$	Quantum Computer $\log_2(Time)$
Provable	AKS Sieve〔8〕	$3.398n$	$2.672n$
	List Sieve〔9〕	$3.199n$	$2.527n$
	AKS Sieve-Birthday〔10〕	$2.648n$	$1.986n$
	List Sieve-Birthday〔11〕	$2.465n$	$1.799n$
Heuristic	NV Sieve〔12〕	$0.415n$	$0.312n$
	Gauss Sieve〔9〕	$0.415n$	$0.312n$

작위 행보는 샘플 벡터와 가까운 필터 벡터를 탐색할 때 사용된다[19]. 처음에 임의의 필터 벡터들을 선정하고, 필터 벡터들의 인접한 다른 필터 벡터들로 unitary operation을 통한 이동을 하면서 최종적으로 phase estimation을 통해 샘플 벡터와 근접한 필터 벡터들을 찾는다.

4.2.2. 양자 Enumeration 알고리즘

Enumeration에서는 계수의 범위를 결정하여 탐색하는 데에 순서가 필요하므로, 모든 원소를 중첩시켜서 탐색하는 Grover 탐색 알고리즘의 적용이 불가능하다. 그러나 enumeration에서 사용되는 tree는 양자 무작위 행보를 이용하여 원하는 원소를 찾을 수 있는 구조이다[20]. Enumeration 알고리즘은 tree내에서 탐색해야 지점의 개수의 상한이 정해져 있고 탐색해야 하는 깊이의 상한 역시 정해져 있다. 따라서 특정 길이(bound R) 이하의 벡터를 목표 지점으로 하여, 목표하는 벡터의 존재 여부를 확인하는 데에 양자 무작위 행보를 적용할 수 있다. $N \leftarrow R$, $N' \leftarrow 0$으로 설정 후에, 목표 벡터가 존재 시 $N \leftarrow \left\lfloor \frac{N+N'}{2} \right\rfloor$, 목표 벡터가 존재하지 않을 시 $N' \leftarrow \left\lfloor \frac{N+N'}{2} \right\rfloor$ 와 같이 범위를 변화시키면서 실제 가장 짧은 길이의 벡터가 가지는 길이의 범위를 특정시킬 수 있다. 이후 가장 짧은 길이를 가지는 벡터의 길이를 찾은 후에는, 해당 길이를 가지는 벡터를 찾는 알고리즘을 이용한다. 따라서 양자 enumeration 알고리즘은 양자 무작위 행보를 적용하여 tree 탐색을 하는 방향으로 수행할 수 있다.

V. 결 론

본 논문에서는 격자 암호 분석 알고리즘 및 양자 탐색 알고리즘을 적용한 양자 분석 기술의 동향에 대하여 서술하였다. 격자 기반 암호 분석 알고리즘들의 안전 강도는 양자 컴퓨팅 환경 위에서의 공격 알고리즘을 가정하여 계산되므로, 격자 암호 분석 기술의 연구 및 양자 탐색 알고리즘의 적용에 관한 연구의 중요성이 강조된다. 격자 암호 분석 기술뿐만 아니라, 양자 컴퓨터 위에서의 격자 기저 축소 알고리즘의 연구 또한 함께 수반되어야 한다. 향후엔 LLL 격자 기저 축소 알고리즘 및 다른 격자 암호 분석 기술들에 대한 연구를 수행하여, 이를 통해 BKZ 격자 기저 축소 알고리즘의[21] 양자 회로 구현을 진행할 예정이며, 양자 BKZ 알고리즘을 이용하여 격자 기반 암호의 안전 강도를 새로 분석하는 연구를 수행할 예정이다.

참 고 문 헌

[1] Shor, Peter W. "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer." SIAM review 41.2 (1999): 303-332.

[2] R. Avanzi, et al., “CRYSTALS-Kyber: Algorithm Specifications And Supporting Documentation,” NIST PQC round 3 submission, Oct. 1, 2020.

[3] S. Bai, et al., “CRYSTALS-Dilithium: Algorithm Specifications and Supporting Documentation,” NIST PQC round 3 submission, Oct. 1, 2020.

[4] P.-A. Fouque, et al., “Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU: Specification v1.2,” NIST PQC round 3 submission, Oct. 1, 2020.

[5] J.-P. Aumasson, et al., “Sphnics+: Submission to the NIST post-quantum project, v.3” NIST PQC round 3 submission, Oct. 1, 2020.

[6] Grover, Lov K. "A fast quantum mechanical algorithm for database search." Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. 1996.

[7] Montanaro, Ashley. "Quantum walk speedup of backtracking algorithms." arXiv preprint arXiv:1509.02374 (2015).

[8] Ajtai, Miklós, Ravi Kumar, and Dandapani Sivakumar. "A sieve algorithm for the shortest lattice vector problem." Proceedings of the thirty-third annual ACM symposium on Theory of computing. 2001.

[9] Micciancio, Daniele, and Panagiotis Voulgaris. "Faster exponential time algorithms for the shortest vector problem." Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics, 2010.

[10] Hanrot, Guillaume, Xavier Pujol, and Damien Stehlé. "Algorithms for the Shortest and Closest Lattice Vector Problems." IWCC 6639 (2011): 159-190.

[11] Pujol, Xavier, and Damien Stehlé. "Solving the shortest lattice vector problem in time 2^ 2.465 n." Cryptology ePrint Archive (2009).

[12] Nguyen, Phong Q., and Thomas Vidick. "Sieve algorithms for the shortest vector problem are practical." Journal of Mathematical Cryptology 2.2 (2008): 181-207.

[13] Laarhoven, Thijs. "Search problems in cryptography: from fingerprinting to lattice sieving." (2016).

[14] Gama, Nicolas, Phong Q. Nguyen, and Oded Regev. "Lattice enumeration using extreme pruning." Advances in Cryptology – EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 – June 3, 2010. Proceedings 29. Springer Berlin Heidelberg, 2010.

[15] Howgrave-Graham, Nick. "A hybrid lattice-reduction and meet-in-the-middle attack against NTRU." Advances in Cryptology-CRYPTO 2007: 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007. Proceedings 27. Springer Berlin Heidelberg, 2007.

[16] Cheon, Jung Hee, et al. "A hybrid of dual and meet-in-the-middle attack on sparse and ternary secret LWE." IEEE Access 7 (2019): 89497-89506.

[17] Laarhoven, Thijs, Michele Mosca, and Joop Van De Pol. "Finding shortest lattice vectors faster using quantum search." Designs, Codes and Cryptography 77 (2015): 375-400.

[18] Xiong, Zhijian, et al. "An improved MITM attack against NTRU." International Journal of Security and Its Applications 6.2 (2012): 269-274.

[19] Chailloux, André, and Johanna Loyer. "Lattice sieving via quantum random walks." Advances in Cryptology – ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6 – 10, 2021, Proceedings, Part IV 27. Springer International Publishing, 2021.

[20] Aono, Yoshinori, Phong Q. Nguyen, and Yixin Shen. "Quantum lattice enumeration and tweaking discrete pruning." Advances in Cryptology – ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2 – 6, 2018, Proceedings, Part I 24. Springer International Publishing, 2018.

[21] Chen, Yuanmi, and Phong Q. Nguyen. "BKZ 2.0: Better lattice security estimates." Advances in Cryptology – ASIACRYPT 2011: 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings 17. Springer Berlin Heidelberg, 2011.

〈저자소개〉

전 찬 호 (Chanho Jeon)

2019년 2월 : 고려대학교 수학과 졸업
2019년 3월~현재 : 고려대학교 정보보호대학원 석박사 통합과정
<관심분야> 후양자암호, 양자알고리즘

허 동 회 (Donghoe Heo)

2019년 2월 : 한양대학교 수학과 졸업
2019년 3월~현재 : 고려대학교 정보보호대학원 석박사 통합과정
<관심분야> 후양자암호, 양자알고리즘

최 영 록 (Younglok Choi)

2021년 2월 : 고려대학교 수학과 졸업
2021년 3월~현재 : 고려대학교 정보보호대학원 석사과정
<관심분야> 후양자암호, 양자알고리즘

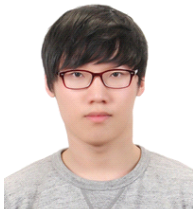
이 명 훈 (Myeonghoon Lee)

2020년 2월 : 고려대학교 수학과 졸업
2020년 3월~현재 : 고려대학교 정보보호대학원 석박사 통합과정
<관심분야> 후양자암호, 대칭키암호

천 병 호 (Byeongho Cheon)

학생회원

2021년 8월 : 호서대학교 컴퓨터정보공학부 졸업
2021년 9월~현재 : 고려대학교 정보보호대학원 석사과정
<관심분야> 후양자암호, 양자알고리즘



이 창 원 (Changwon Lee)

학생회원

2021년 2월 : 서울시립대학교 수학과 졸업
2021년 3월~현재 : 고려대학교 정보보호대학원 석사과정
<관심분야> 후양자암호, 양자알고리즘

김 수 리 (Suhri Kim)

정회원

2014년 2월 : 고려대학교 수학과 졸업
2016년 8월 : 고려대학교 정보보호대학원 공학석사
2020년 2월 : 고려대학교 정보보호대학원 공학박사
2020년 3월~2021년 2월 : 고려대학교 정보보호대학원 박사후연구원
2020년 3월~2021년 2월 : K.U. Leuven ESAT/SCD-COSIC 박사후연구원
2022년 3월~현재 : 성신여자대학교 수리통계데이터사이언스학부 조교수
<관심분야> 공개키 암호시스템, 후양자암호