Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.34, NO.2, Apr. 2024 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2024.34.2.167

SeaSign에 대한 효율적인 서명 방법 및 최적 파라미터 제안 연구*

김 수 리†‡
성신여자대학교 (교수)

A Study on Efficient Signing Methods and Optimal Parameters Proposal for SeaSign Implementation*

Suhri Kim†‡
Sungshin Women's University (Assistant Professor)

요 약

본 논문은 isogeny 기반 전자 서명 알고리즘인 SeaSign의 최적화 방안을 제안한다. SeaSign은 CSIDH의 class group action에 Fiat-Shamir with abort를 결합한 전자서명 알고리즘이다. CSIDH 기반 암호는 SIDH 기반 암호가 다항시간안에 공격됨에 따라 다시 주목받고 있지만, 이를 기반한 전자서명인 SeaSign은 비효율적인 속도로 많은 최적화가 진행되지 않았다. 본 논문에서는 SeaSign에 대한 효율적인 서명 방법을 제안한다. 제안하는 서명 방법은 간단하지만 강력하며, 알고리즘 내에서 rejection sampling의 위치 변경을 통해 이루어진다. 추가로, 본 논문에서는 제안하는 알고리즘이 최적 성능을 제공할 수 있는 파라미터를 제시한다. 제시한 결과, 기존 SeaSign의 파라미터를 사용할 경우, 본 논문에서 제안한 서명방법은 기존 SeaSign 대비 3배 빠른 성능을 보인다. 추가로 신규 제시된 파라미터와 본 논문의 서명 방법을 결합한 경우, 기존 SeaSign 대비 290배 빠른 성능과 Decru 등이 제안한 방법 대비 7.47배 빠른 성능을 제공한다.

ABSTRACT

This paper proposes optimization techniques for SeaSign, an isogeny-based digital signature algorithm. SeaSign combines class group actions of CSIDH with the Fiat-Shamir with abort. While CSIDH-based algorithms have regained attention due to polynomial time attacks for SIDH-based algorithms, SeaSiogn has not undergone significat optimization because of its inefficiency. In this paper, an efficient signing method for SeaSign is proposed. The proposed signing method is simple yet powerful, achived by repositioning the rejection sampling within the algorithm. Additionally, this paper presnts parameters that can provide optimal performance for the proposed algorithm. As a result, by using the original parameters of SeaSign, the proposed method is three times faster than the original SeaSign. Additonally, combining the newly suggested parameters with the signing method proposed in this paper yields a performance that is 290 times faster than the original SeaSign and 7.47 times faster than the method proposed by Decru et al.

Keywords: post-quantum cryptography, isogeny-based cryptography, SeaSign, rejection samploing, CSIDH

Received(01. 10. 2024), Modified(02. 21. 2024), Accepted(02. 21. 2024)

* 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2022R1F1A1063611).

† 주저자, suhrikim@sungshin.ac.kr
‡ 교신저자, suhrikim@sungshin.ac.kr(Corresponding author)

I. 서 론

Isogeny 기반 암호의 시작은 Couveignes의 제안으로 부터라고 볼 수 있다[1]. 2006년 Couveignes는 유한체 위에서 정의된 두 타원곡선 사이의 isogeny를 찾는 어려움을 기반으로 키 교환 방식과 전자 서명 방법을 제안하였으며, 이후 이 아이디어가 Rostovtsev와 Stolbunov에 의해 발전되어 오늘날 CRS 암호라고 불리게 된다. CRS는 ordinary curve를 사용하여 암호를 제안하였는데, ordinary curve의 경우 endomorphism ring이 가환적이라는 특징이 있어, 후에 2014년 Childs 등에 의해 이를 활용한 양자 하지수시간 알고리즘에 의해 공격당하게 된다[2]. 하지만 CRS 암호의 더 큰 문제점은 하지수시간의 공격 보다도 느린 속도에 문제가 있었으며, 지수시간의 복잡도를 가진 래티스 기반 암호와 같은 다른 PQC (Post-quantum cryptography) 암호에 비해 몇 배나 느린 속도로 주목받지 못하게 되었다. Isogeny 기반 암호가 다시 활발하게 연구되기 시작한 것은 2011년 De Feo와 Jao에 의해 제안된 Supersingular Isogeny Diffie-Hellman (SIDH) 부터이다[3]. SIDH는 supersingular 곡선을 사용하여 endomorphism ring이 가환적이지 않아 기존 Childs 등의 공격에 대응하여 양자 지수시간의 복잡도를 가진다. 또한, 최적 알고리즘 개발로 인해 CRS보다 훨씬 효율적인 성능을 자랑한다. SIDH 에 기반한Supersingular Isogeny Key Encapsulation (SIKE)는 NIST PQC 표준화 공모전에 제출되어 Round 4의 대체 후보로 선정되었다. 이후, SIKE를 비롯한 SIDH 기반 암호는 제안된 지 10년 동안 Meet-in-the-Middle (MitM)보다 효율적인 공격 방법이 제안되지 않았으며, 심지어 양자 컴퓨터를 활용해서 공격하는 것보다 고전 컴퓨터를 활용한 공격이 더 효율적일 수 있다는 연구결과가 나오면서 표준화될 PQC 암호의 후보로 자리매김하였다. 하지만, 2023 Castryck-Decru의 키 복구 공격에 의해 모든 보안강도에서 SIDH 기반 암호의 개인키가 다항시간안에 분석되면서 더 이상 사용할 수 없게 된다.[4]에서 제안된 Castryck-Decru의 공격은 Alice와 Bob이 키를 교환하는 과정에서 주고받는 torsion point 정보를 활용한다. SIDH 기반 암호의 경우 비가환적인 성질로 인해 Diffie-Hellman 키 교환처럼 단순히 값을 연산해서 전달해주는 것으로 같은 공유키를 유도할 수 있는 것이 아니라, 자신의 개인키로 상대방의 공개키를 연산한 결과를 전달해 주어야 같은 공유키를 유도할 수 있다. 이 정보를 torsion point information이라 하며, 이 정보와 Kani의 알고리즘을 활용하면 SIDH 기반 암호의 개인키가 복구된다. 이렇듯 SIDH 기반 암호가 더 이상 사용할 수 없게 되면서 CSIDH 기반 암호가 isogeny 기반 암호의 대체로 떠오르게 된다.

Castryck 등에 의해 제안된 CSIDH (Commutative SIDH)는 기존 CRS 기반 암호를 supersingular curve를 활용하는 것을 통해 파라미터 선택을 모던화 한 키 교환 알고리즘이다[5]. CRS 기반 암호의 장점은 CCA 안전한 암호화 설계가 가능하며, 가환적인 성질로 non-interactive 키 교환 알고리즘 설계가 가능하다. 따라서, 이 장점을 살리고자 De Feo, Kieffer 등과 Castryck 등이 독립적으로 CRS 기반 암호의 최적화 방법을 제안하였다. 이 중 CSIDH는 F_{p^2} 에서 정의된 supersingular curve를 F_p로 제안하여, 가환성을 유지하면서 supersingular curve의 효율적인 파라미터 선택을 가능하게 하였다. 현재, 고전 128-비트 보안강도에서 CSIDH 기반 암호의 키 교환을 진행하는데 대략 80ms 정도의 시간이 걸린다. 처음 제안당시에는 동일한 양자 보안강도에서는 SIDH 기반 암호보다 더 느린 속도와 더 큰 키 사이즈로 주목받지 못하다가, SIDH 기반 암호가 Castryck-Decru 공격으로 인해 다항시간 안에 해결되면서 CSIDH 기반 암호가 다시 주목받기 시작하였다.

Isogeny 기반 암호는 느린 속도가 큰 단점으로 여겨지고 있지만, 사실 이 못지 않게 단점으로는 효율적인 전자 서명 알고리즘의 부재라고 볼 수 있다. 처음 제안된 isogeny 기반 전자서명으로는 Yoo 등이 제안하였다[6]. Yoo 등은 양자 컴퓨팅 환경에서도 안전한 Fiat-Shamir transform 인 Unruh의 방법을 사용하여 isogeny 기반 전자서명을 제안하였다. 하지만, 제안한 전자서명은 속도가 느릴 뿐만 아니라, 다른 PQC 전자 서명에 비해 훨씬 큰 서명과 키 사이즈로 isogeny 기반 암호가 가지고 있던 장점마저도 사라진 형태의 전자서명이였다. 또한, 마찬가지로 SIDH 기반으로 설계가 되었기 때문에 기존 Castryck-Decru 공격에 대응할 경우 성능이 더 좋지 않은 결과를 가져온다. 이와 독립적으로 비슷한 시기에 Galbraith, Petit, Silva는 KLPT

알고리즘을 사용하여 isogeny 기반 전자 서명 알고리즘을 제안하였다[7]. 해당 알고리즘은 수학적으로 완성도가 높은 알고리즘이나, 마찬가지로 비효율적인 성능이였는데, 이후 이 KLPT 알고리즘을 최적화하는 파라미터를 활용한 SQISign은 현재 isogeny 기반 암호중에서 가장 효율적인 전자서명으로 주목받고 있다. SQISign은 NIST PQC 전자서명 표준화 공모전 Round 1의 후보로 등록되어있다.

한편, 2019년에 CSIDH 기반 전자서명인 SeaSign이 De Feo와 Galbraith에 의해 제안되었다[8]. 해당 방법은 기존 Stolbunov가 제안한 전자서명 알고리즘에서 개인키가 노출되는 부분을 rejection sampling을 적용하여 대응하였으며, Lyubashevsky의 'Fiat-Shamir with abort'을 활용하여 안전한 전자 서명을 제안하였다. 초기 제안된 Stolbunov의 버전에서는 같은 균등 분포에서 선택된 개인키 벡터와 랜덤한 벡터의 차를 서명값에 포함한다. 균등 분포에서 선택된 벡터의 차는 균등 분포를 이루지 않기 때문에 다량의 서명값을 획득할 경우 개인키의 분포가 노출되는 위험이 있다. 따라서 De Feo와 Galbraith는 rejection sampling 방법을 활용하여, 랜덤한 벡터는 더 큰 범위의 균등분포에서 선택하고, 개인키와 랜덤한 벡터의 차의 성분이 특정 범위 안에 존재할 때만 서명값으로하여 개인키 분포의 노출을 막았다. 하지만, 랜덤한 벡터를 선택하는 범위가 isogeny 연산 횟수에 영향을 미치는 만큼 몇 시간 이상이 되는 서명 속도로 비효율적이다. 한편, SeaSign 논문에는 만약 class group의 구조를 파악할 수 있을 경우, class group의 랜덤한 ideal을 기존 벡터형태가 아니라 이산대수 문제와 유사하게 generator에 대한 지수 형태로 선택할 수 있고, 이 경우에는 랜덤 ideal을 선택하기 위해 더 큰 범위를 사용할 필요가 없어 효율적이라고 제시하였다. 해당 방법을 실제로 구현하여 제시한 결과가 CSI-FiSh 이다[9]. CSI-FiSh는 기존 CSIDH-512 파라미터에 사용된 512비트 소수 위에 정의된 타원곡선에 대해 class group 연산을 실제로 수행하여 해당 class group의 generator 하나로 생성된 cyclic group이라는 결론을 통해 효율적인 서명 속도를 제시하였다. 제안하는 방법을 활용하면 고전 128비트 보안강도에서 서명하는데 대략 390ms 정도의 속도로, 이는 isogeny 기반 암호에서는 획기적인 결과로 CSIDH 기반 전자서명이 주목받게 되었다. 그러나 최근 CSIDH 기반 암호에 대한 엄밀한 양자 분석 결과 128비트 양자 보안강도를 위해서는 1024비트 이상의 큰 유한체가 필요하고, class group 연산은 현재 컴퓨팅 환경에서 하지수시간 복잡도를 가지고 있기 때문에, CSI-FiSh는 확장성이 부족하다. 이후, CSI-FiSh의 확장성을 위해 KLPT 알고리즘처럼 두 타원곡선 사이의 ideal을 직접 찾는 방법을 활용한 전자서명인 SCALLOP이 제안되고, Kani 방법과 결합해서 이를 고차원에서 확장한 SCALLOP-HD가 제안되었으나, 공격 방법인 Clapoti(s) 알고리즘이 제안되면서 더 이상 사용할 수 없게 된다[10].

본 논문은 SeaSign의 성능 향상을 목적으로 한다. 결론적으로 SeaSign에서 성능 측면에서의 주된 병목 현상은 rejection sampling 과정에 있으므로, 본 논문에서는 이를 개선하는 방안에 대해 제안한다. 본 논문의 기여는 다음과 같이 정리할 수 있다.

- 본 논문에서는 먼저 rejection sampling이 사용되지 않은 환경에서 SeaSign의 안전성을 확인한다. 이를 위해 rejection sampling을 사용하지 않을 경우 SeaSign 서명을 수집한 뒤 개인키를 복원하는 공격 도구를 개발하였다. 또한, 해당 공격 도구를 사용하여 rejection sampling이 포함된 SeaSign의 안전성을 분석하였다.
- 본 논문에서는 SeaSign의 효율적인 성능을 위한 새로운 파라미터를 제안한다. 기존 rejection sampling을 사용한 경우에는 높은 rejection율일 경우에 성능 저하가 심하므로, 큰 범위에서 벡터를 선택하게 된다. 하지만 본 논문에서는 추가로 SeaSign의 서명 알고리즘을 변경하여 작은 범위에서도 안전성을 만족하면서 효율적인 전자서명이 가능한 방법을 제안한다.
- 제안한 서명방법은 간단하지만 강력한 방법으로, rejection sampling의 위치 조정을 통해 효율적인 서명 생성 방법을 제안한다. 이를 활용하면 rejection 이 일어나지 않으면서도 기존 SeaSign의 아이디어와 유사하게 개인키의 노출이 일어나지 않는다. 제안하는 서명 방법에 대한 자세한 내용은 3장에서 소개한다.
- 본 논문에서는 SeaSign과[11]에서 제안된 최적화 방법, 그리고 본 논문에서 제안한 방법의 파라미터를 활용하여 서명 속도 비교를 진행한다. 실험 결과, SeaSign의 기본 파라미터를 사용할 경우, 본 논문의 서명 방법은 기존 SeaSign 대비

3배 빠른 성능을 보인다. 추가로, 본 논문에서 제시한 파라미터를 사용할 경우 기존 SeaSign보다 약 290배 빠르며, Decru 등이 제안한 방법보다 7.47 배 빠르다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 필요한 CSIDH, SeaSign을 소개한다. 3장에서는 rejection sampling을 사용하지 않은 SeaSign에 대한 공격 결과를 제시하고, 이를 활용해서 rejection sampling을 사용할 경우 적정 파라미터를 신규 서명 알고리즘과 함께 소개한다. 4장에서는 기존 SeaSign의 최적화와 비교하여 본 논문의 구현 결과를 제시하고, 5장의 결론으로 마무리 한다.

II. 배경지식

본 장에서는 논문에 필요한 배경 지식을 소개한다. 먼저 CSIDH에 대해 소개를 한 다음 여기에 기반을 둔 전자서명인 SeaSign을 소개한다.

2.1 CSIDH

CSIDH는 기존 CRS 기반의 non-interactive key exchange가 가능하다는 장점을 살리기 위해 supersingular curve를 사용하는 아이소제니 기반 키 교환 프로토콜이다[5]. 기존 CRS 기반 암호의 경우 ordinary curve를 사용하였는데 ordinary curve의 경우 특정 조건을 만족하는 위수를 선택하는 부분이 어렵다. 반면에 CSIDH는 F_p 위에 정의된 supersingular curve를 사용하여, 가환성을 가지는 group action을 활용할 수 있을 뿐만 아니라 효율적인 파라미터 선택이 가능해져 기존 CRS 기반 키 교환 알고리즘보다 속도를 향상시켰다.

E를 F_p위에서 정의된 supersingular 타원곡선이라 하고, 이 경우 타원곡선 E의 F_p위에서 정의된 endomorphism ring $End_p(E)$은 $End_p(E)=Z[\pi]$를 만족한다. 한편, $End(E)$는 quaternion order로 $End_p(E)$는 $End(E)$의 가환적인 subring이 되며, supersingularity로 인해 trace는 0이 되고, $E(F_p)=p+1$을 만족한다. 만약 유한체 F_p의 소수 p를 $p=f\prod \ell_i-1$ 의 형태라고 가정하자. 여기에서 ℓ_i는 서로 다른 홀수 소수를 의미하고 f는 p가 소수가 되게하는 cofactor를 의미한다. 이 경우 $E(F_p)=p+1=f\prod \ell_i$로 가 되어 모든 ℓ_i의 torsion point를 가질 뿐만 아니라, 각 소수 ℓ_i에 대해 $\pi^2-1\equiv 0 \bmod \ell_i$를 만족하게 된다. 따라서 ideal $\ell_i O$는 $\ell_i O=\iota_i\bar{\iota_i}$의 형태로 분해가 가능한데, 여기에서 $\iota_i=(\ell_i,\pi-1)$, $\bar{\iota}_i=(\ell_i,\pi+1)$을 의미한다. 이는 group action $[\iota_i]E$는 F_p위의 아이소제니 ϕ_{ι_i}로 연산할 수 있음을 의미한다.

CSIDH를 활용한 키 교환 방법은 다음과 같다. Alice와 Bob가 서로 키를 교환한다고 하자. Alice는 랜덤한 ideal $[a]=[\iota_1^{e_1}\cdots\iota_n^{e_n}]$를 선택하는데, 여기에서 ι_i는 고정이 되기 때문에 $e_1,\cdots,e_n$만 선택하면 된다고 볼 수 있으며, 이는 벡터 $(e_1,\cdots,e_n)\in Z^n$를 선택한다고 볼 수 있다. 따라서 CSIDH 기반 암호에서 개인키 ideal $[a]$과 동치인 것은 Z^n의 랜덤한 벡터 $(e_1,\cdots,e_n)$이다. 이때 각각의 e_i는 양의 정수 m에 대해서 $e_i\in[-m,m]$의 범위를 가지게 되어 전체 가능한 벡터의 가지수 $(2m+1)^n$가 class group의 원소의 개수를 포함하도록 한다.

Alice는 Velu 공식을 활용하여 공개키 $E_A=[a]E$를 연산하고 E_A를 Bob에게 전달한다. Bob도 마찬가지로 개인 ideal $[b]$에 해당하는 벡터를 선택한 뒤, 공개키 $E_B=[b]E$를 Alice에게 전달한다. Alice는 Bob으로 받은 E_B에 대해 자신의 개인 ideal로 $[a]E_B$를 연산하고, Bob도 Alice로부터 받은 E_A에 대해 자신의 개인 ideal로 $[b]E_A$를 연산한다. CSIDH는 가환적인 성질을 가지기 때문에 Diffie-Hellman 키 교환과 유사하게 연산된 결과가 $[a]E_B=[b]E_A$를 만족한다.

2.2 SeaSign

De Feo와 Galbraith에 의해 제안된 SeaSign은 CSIDH의 class group action에 Fiat-Shamir with abort를 적용하였다[8]. SeaSign은 기존 제안된 Stolbunov의 방법에서 개인키의 분포가 노출되는 것을 rejection sampling을 통해 해결하였다. CSIDH 기반 알고리즘에서는 공개키는 유한체 F_p위에 정의된 supersingular curve E와 $E_A=[a]E$ 이

다. 여기에서 $[a]$는 개인키로, class group에서 ideal을 나타낸다. 소수 p가 $p=4\ell_1\ell_2\cdots\ell_n-1$의 형태를 가지므로, $[a]=\prod_{i=1}^{n}\iota_i^{e_i}$ 형태라고 볼 수 있으며, 따라서 $[a]=(e_1,\cdots,e_n)$의 벡터 형태로 표현된다. 일반적으로 e_i는 양의 정수 B에 대해서 $e_i\in[-B,B]$ 범위에서 균등하게 선택된다.

Stolbunov의 identification protocol에서는 검증자는 랜덤한 ideal $[b]=(f_1,\cdots,f_n)$ 에서 선택한다. 여기에서 $f_i\in[-B,B]$를 따른다. 만약에 challenge bit가 0일 경우에 검증자는 $[b]$를 제시하고, challenge bit가 1일 경우에 $[b][a]^{-1}$을 제시한다. 여기에서 $[b][a]^{-1}$는 $(f_1-e_1,\cdots,f_n-e_n)$으로 표현될 수 있으며, f_i-e_i의 분포는 더 이상 균등하지 않기 때문에 e_i의 분포가 드러나게 된다.

이러한 현상을 제거하기 위해, SeaSign에서는 f_i를 $[-B,B]$에서 선택하지 않고, 양의 정수 δ에 대해 $[-(\delta+1)B,(\delta+1)B]$ 범위에서 선택한다. SeaSign에서 $\delta=nt+1$로, n은 유한체에 사용된 홀수 소수의 개수, t는 전자서명에 사용되는 해시 함수의 출력 비트의 크기이다. $[b][a]^{-1}$를 진행하는 과정에서 $f-e$를 연산한 뒤, 만약 모든 f_i-e_i가 $[-\delta B,\delta B]$ 범위에 있을 경우 accept하고 그렇지 않으면 reject 한다. 해당 과정을 알고리즘으로 나타내면 다음과 같다.

512비트 유한체 상에서는 실제 class group 연산을 통해서 SeaSign을 최적화 할 수 있다. [] 에서는 CSIDH-512 파라미터를 타겟으로 class group을 연산하고, 그 결과 $Cl(O)$는 ideal $\iota=\langle 3,\pi-1\rangle$ 로 생성되는 cyclic group이라는 것을 확인하였다. 따라서 기존에 class group을 모르기 때문에 랜덤한 벡터 e를 선택하여 ideal $[a]=\prod_{i=1}^{n}\iota_i^{e_i}$ 을 $Cl(O)$에서 랜덤하게 ideal을 선택했다 여기는 것과 다르게, $\#Cl(O)=n$일 경우, 이산대수 형태처럼 Z_n에서 랜덤한 수 r을 선택하면 ι^r은 $Cl(O)$의 랜덤한 ideal이 된다. 따라서 이를 활용할 경우, $[b][a]^{-1}$의 연산이 $\iota^{r-r'}$ 형태로 변경될 수 있어, 더 이상 개인키에 대한 노출 없이 연산이 가능하다. 이 결과 512비트 유한체에서 서명하는데 390ms 가 걸리게 되었으며, 하루 이상이 걸리던 기존 SeaSign보다 빠를 뿐만 아니라, 그때 제안되었던 다른 isogeny 기반 전자서명에 비해서도 효율적인 속도를 제공하였다. 하지만, CSIDH에 대한 면밀한 양자 안전성 분석 결과, 128비트 양자 안전성을 이루기 위해서 유한체 크기는 1024비트 보다 커져야 하고, class group 연산 자체도 현재 컴퓨팅 환경에서 하지수시간 복잡도가 걸리는 만큼, CSI-FiSh의 다른 파라미터로의 확장은 불가능한 상황이다.

Algorithm 1.
SeaSign Signature Generation [8]

Require: A message m, (E,E_A), a private key e
Ensure: $\sigma=(z_1,\cdots,z_t,b_1,\cdots,b_t)$

for k from 0to $t-1$ do
  $f_k\leftarrow[-(\delta+1)B,(\delta+1)B]^n$
  $E_k=\left(\prod_{i=1}^{n}\iota_i^{f_{k,i}}\right)E$
end for
$b_1\parallel\cdots\parallel b_t=H(j(E_1),\cdots,j(E_t),m)$
for k from 0to $t-1$ do
  if $b_k=0$ then
    $z_k\leftarrow f_k$
  else
    $z_k\leftarrow f_k-e$
  end if
  if $z_k\notin[-\delta B,\delta B]^n$ then reject
  end if
end for

Fig. 1. SeaSign Signature Generation in [8]

III. SeaSign 안전성 분석 및 최적화

본 장에서는 기존 SeaSign의 최적화 방법에 대해 소개한 뒤, SeaSign 안전성 분석을 통한 새로운 최적화 방안에 대해 제시한다.

3.1 기존 최적화 방안

[11]에서는 향상된 rejection sampling 방법을 활용하여 SeaSign의 최적화 방안을 제안하였다. 해당 논문에는 SeaSign 최적화를 위한 2가지 방안이 제시되었지만, 가장 핵심적인 아이디어는

'Fiat-Shamir with aborts'에 대한 변형이라고 볼 수 있다. 해당 방법은 서명자가 서명시에 몇 개의 challenge를 reject 할 수 있도록 하는 방안이다. 이를 위해 새로운 파라미터 $u \geq 0$ 이 사용되는데,

Algorithm 2.
Optimization in [11]

Require: A message m, (E, E_A), a private key e

Ensure: $\sigma = ([f_1]E, \cdots [f]_{t+u}E, z_1, \cdots, z_{t+u})$

1 for k from 1 to $t+u$ do
2 $f_k \leftarrow [-(\delta+1)B, (\delta+1)B]^n$
3 $E_k = \left(\prod_{i=1}^{n} \iota_i^{f_{k,i}}\right) E$
4 end for
5 $b_1 \| \cdots \| b_{t+u} = H(j(E_1), \cdots, j(E_{t+u}), m)$
6 cnt$\leftarrow$0
7 iter$\leftarrow$1
8 $L = []$
9 while iter $\leqq t+u$ do
10 select k randomly from 1 to $t+u$
11 if k not in L then
12 $k \in L$
13 iter$\leftarrow$iter+1
14 else
15 go to step 10
16 end if
17 if $b_k = 0$ then
18 $z_k \leftarrow f_k$
19 cnt $\leftarrow$cnt+1
20 else
21 $z_k \leftarrow f_k - e$
22 if $z_k \notin [-\delta B, \delta B]^n$ then
23 cnt $\leftarrow$cnt+1
24 end if
25 end if
26 end while
27 if cnt $< t$ then
28 go to step 1
29 end if

Fig. 2. Proposed optimizatino of SeaSign Signature Generation in [11]

이는 서명자가 응답을 거부할 수 있는 challenge의 개수를 의미한다.[11]에 제안하는 서명 방법은 다음 알고리즘으로 요약할 수 있다.

Algorithm 2를 활용하여 서명을 진행하면, 서명값에 정확히 u개의 reject 범위 안에 있는 z_k들이 존재한다. $u=0$일 경우에는 기존 SeaSign과 동일하다는 것을 의미하며, $u>0$인 경우에는 reject 범위 안에 있는 z_k를 포함한다는 의미이며, 이는 기존 SeaSign일 경우라면은 바로 reject하고 다시 서명 생성을 시작한다면, Algorithm 2의 경우에는 u번까지만 허용하고 $u+1$번째 서명 생성을 다시 신작한다는 의미이다. 한편, t개 미만으로 accept 범위 안에 있는 z_k가 생성될 때에도 서명을 다시 생성한다. 이를 통해 rejection 확률이 기존 SeaSign 보다 줄어들기 때문에 더 효율적인 속도를 기대할 수 있다.

3.2 제안하는 최적화 방안

본 논문에서는 서명 알고리즘의 변경을 통해 서명 속도를 향상시키는 방안에 대해 제안한다. 제안하는 방법은 기존처럼 벡터 z_k를 생성한 뒤, hash output이 1일 경우 랜덤한 벡터와 개인키 e의 차를 계산한 결과인 $z_k - e$ 의 성분의 범위가 특정 조건을 만족하지 않으면 바로 reject 하는 것이 아니라, 벡터 z_k를 생성할 때, 먼저 $z_k - e$를 계산해서 특정 범위 안에 있는 벡터들을 필요한 개수대로 모은 다음에, hash output 이 0일 경우 원래의 z_k를, 1일 경우 $z_k - e$를 서명값에 포함하는 방법에 대해 제안한다. 이 경우 z_k는 $z_k - e$ 가 $[-\delta B, \delta B]$인 범위가 되도록 하게 z_k가 선택된다. 하지만, 해당 z_k는 개인키 정보를 포함하지 않기 때문에 개인키 정보가 드러나지 않는다. 최적의 δ를 선택하기 위해 먼저 알고리즘을 제안한 후, SeaSign의 안전성을 분석한다.

3.2.1 서명 알고리즘의 변형

제안하는 알고리즘은 간단하지만 강력한 방법으로, 알고리즘 내의 rejection sampling에 대한 변경이다. 방법은 랜덤한 ideal (vector)를 먼저 샘플한 다음에, 개인키 벡터와의 차를 구한 다음, 특정 범위 내에 있으면 isogeny 연산을 수행하고 그렇지 않으면 다시 선택한다. 좀 더 명확하게 설명하자면,

랜덤한 ideal $[b_k]=(f_{k,1},\cdots,f_{k,n})$에 대해 b_k-e 의 각 성분 $f_{k,i}-e_i$가 $[-\delta B,\delta B]$내에 있을 경우 저장한다. 이러한 t개의 $[b_k]$에 대해, $E_k=[b_k]E$ 를 연산한다. 그 후 서명 과정에서 메시지 m에 대해 $H(j(E_1),\cdots j(E_t),m)$을 연산한다. 여기에서 $j(\bullet)$는 타원곡선의 j-invariant를 의미한다. 만약 해시의 output 비트가 0일 경우 $[b_k]$를 출력하고, 1일 경우 $[b_k][a]^{-1}=f_k-e$를 출력한다. 이미 f_k를 선택하는 과정에서 f_k-e가 개인키의 정보를 노출하지 않기 때문에 안전하다.

기존 SeaSign의 경우에는, t개의 랜덤한 ideal을 샘플한 뒤, $E_k=[b_k]E$를 연산하고, 해시값을 마찬가지로 계산하여, 해시 output이 1일 경우 f_k-e 서명값에 포함하는 과정에서 f_k-e의 범위가 특정 조건을 만족하지 않으면 다시 t개의 랜덤한 ideal을 선택하는 과정으로 돌아간다. $E_k=[b_k]E$ 연산과정이 가장 성능이 저하되는 부분이기 때문에, 해당 과정을 반복함으로써 SeaSign의 성능이 저하되었다. 하지만 제안하는 방법의 경우 먼저 ideal이 특정 범위를 만족하는지 확인하고, 범위을 만족하는 ideal에 대해서만 isogeny 연산과정이 들어가기 때문에 rejection이 일어나지 않으며 더 효율적으로 동작한다. 제안하는 방법을 알고리즘으로 나타내면 다음과 같다.

Algorithm 3.
Proposed Signature Generation

Require: A message m, (E,E_A), a private key e

Ensure: $\sigma=(z_1,\cdots,z_t,b_1,\cdots,b_t)$

cnt$\leftarrow$1
while cnt$\leqq t$ do
  $f_{cnt}\leftarrow[-(\delta+1)B,(\delta+1)B]^n$
  $b\leftarrow f_{cnt}-e$
  if $b\in[-\delta B,\delta B]^n$ then
    $z_{cnt}\leftarrow f_{cnt}$
    $E_{cnt}=\left(\prod_{i=1}^{n}\iota_i^{f_{cnt,i}}\right)E$
    cnt $\leftarrow$cnt+1
  end if
end while
$b_1\parallel\cdots\parallel b_t=H(j(E_1),\cdots,j(E_t),m)$
for k from 0to $t-1$ do
  if $b_k=0$ then
    $z_k\leftarrow f_k$
  else
    $z_k\leftarrow f_k-e$
  end if
end for

Fig. 3. Proposed Signature Generation

정리 1. 벡터 z_k의 분포는 개인키 e에 대해 독립적이다.

proof. Algorithm 3은 $f-e$의 각 성분의 분포가 $[-\delta B,\delta B]$에 있을 경우에만 f를 sample 한다. $f-e$가 $[-\delta B,\delta B]$ 범위에 존재한다는 의미는, f가 $[-(\delta-1)B,(\delta+1)B]$에 존재한다는 의미이다.

만약 $b_i=0$이면, f가 결과적으로는 $[-(\delta-1)B,(\delta+1)B]$ 범위에 존재하지만, 해당 부분에 개인키 정보가 포함되지 않고, f 자체는 $[-(\delta+1)B,(\delta+1)B]$에서 sampling 한 결과이다. 따라서 f는 개인키 정보를 포함하지 않는다. $b_i=1$일 경우에는 Algorithm 3은 $f-e$를 공개하고, 이 결과는 $[-\delta B,\delta B]$ 범위 내에 존재하며, 마찬가지로 개인키 정보를 노출하지 않는다. 다시 말해, 원래 SeaSign도 $b_i=1$ 경우에는 f자체가 $[-(\delta-1)B,(\delta+1)B]$ 범위를 가져 $f-e$의 결과가 $[-\delta B,\delta B]$ 가 된다는 경우를 의미하고, 본 논문에서 제안된 결과는 $b_i=0$인 경우에도 $[-(\delta-1)B,(\delta+1)B]$ 범위에 있는 f를 선택한다는 것을 의미한다. □

3.2.2 분석을 통해 변형된 서명 알고리즘에 대한 최적 파라미터 제안

본 장에서는 서명값에서 얻은 $f-e$ 를 통해 개인키를 분석하는 도구에 대해 설명한다. 해당 도구를 활용하여 기존 CRS 기반 전자서명의 안전성과 SeaSign을 분석한 후, 제안하는 새로운 서명 알고리즘의 최적 파라미터를 제시한다.

3.2.2.1 서명알고리즘에 대한 안전성 분석

먼저, 목표는 개인키에 해당하는 ideal $[a]$ 혹은 이에 대응하는 벡터 $e=(e_1, \cdots, e_n)$의 복원이다. 해당 공격은 SeaSign-512 파라미터를 대상으로 제시하였으며, 해당 파라미터는 〔Table 1〕과 같이 주어진다. 여기에서 사용하는 512비트 소수 p는

$$p=4\prod_{i=1}^{n-1}\ell_i \cdot \ell_f - 1$$

형태로 주어지며, 여기에서 $\ell_1, \cdots, \ell_{n-1}$은 처음부터 작은 $n-1$개의 서로 다른 홀수 소수를 의미하며, ℓ_f는 $p+1$의 가장 큰 소수 인수에 해당한다. B는 개인키 선택의 범위를 의미하는데, 개인키를 $[a]=(e_1, \cdots, e_n)$이라 할 때, $e_i \in [-B, B]$를 의미한다. t는 해시함수의 출력 비트의 크기를 의미하는데, 본 논문에서는 SHA-256을 고려한다. 마지막으로 랜덤한 벡터 f의 성분이 선택되는 범위인 $[-(\delta+1)B, (\delta+1)B]$에서 $\delta=nt$에 해당된다.

공격은 다음과 같이 진행된다. 먼저 개인키 $e=(e_1, \cdots, e_{74})$ 가 선택된다. 그 뒤 랜덤한 벡터 $f=(f_1, \cdots, f_{74})$가 선택된 다음에 $f-e$가 연산된다. $c_i=f_i-e_i$라 하자. 각 c_i에 대해서 가능한 e_i의 값들이 list의 i 번째에 저장된다. 예를 들어 $c_1=9$인 경우, $f_1=5, e_1=-4$ 이거나 $f_1=4, e_1=-5$가 가능하다. 따라서 index 1번째에 -4와 -5를 저장한다. 74개의 성분에 대해서 저장한 뒤, 랜덤한 f를 다시 선택하고 $f-e$를 연산한다. 마찬가지로 $c_i=f_i-e_i$라 하자. 이번$f_1=4, e_1=-4$에 $c_1=8$이라 하자. 이 경우 $f_1=5, e_1=-3$, , $f_1=3, e_1=-5$가 가능하다. 따라서, 가능한 e_1의 값으로는 -3, -4, -5 이지만, 기존 index 1번째에 저장된 값이 -4, -5이기 때문에, -3은 불가능하다는 것을 의미한다. 따라서 기존 list와 충돌이 일어나는 -4, -5만 저장한다. 해당 방법은 index에 해당하는 개인키 성분의 후보가 전수조사가 가능해지는 범위 (2^{60} 이하) 가 될 때까지 수행한다. 해당 공격 방법을 알고리즘화 하면 다음과 같이 정리할 수 있다.

Table 1. Parameter for SeaSign-512

$\log_2 p$	n	ℓ_f	B	t
511	74	373	5	256

Algorithm 4.
Recovering the private key through $f-e$

Require: empty list L of length n
Ensure: A private key $e=(e_1, \cdots, e_n)$

```
target_complexity ← t
While ttarget_complexity ≥ t do
  f ← [−d, d]^n
  b ← f − e
for i from 0 to n − 1 do
   find possible cases (f_i, e_i) for b_i
   detect collision on e_i and record e_i in L[i[
end for
for i from 0to n − 1 do
 tmp ← tmp · len(L[i])
end for
tmp = log_2 tmp
target_complexity ← tmp
end while
```

Fig. 4. Recovering the private key

실제 공격을 기존 CRS 전자서명에 수행한 결과는 다음 〔Table 2〕와 같다. 〔Table 2〕에서 n은 벡터의 차원을 의미하고, B는 개인키의 성분이 선택되는 범위를 의미한다. d는 랜덤한 벡터가 선택되는 범위를 의미한다. 또한, **attempts**는 개인키를 완전히 복원하는데 필요한 횟수를 의미하고 **success**는 성공여부를 의미한다. 0은 개인키 복원이 불가능함을 의미하고 1은 가능함을 의미한다.

〔Table 2〕의 결과에서 확인할 수 있듯이 rejection sampling을 사용하지 않을 경우 100%로 개인키를 복원할 수 있음을 의미한다. 특히, 해당 경우 각 성분의 index당 개인키로 가능한 값이 1개씩 밖에 남지 않아 추가 전수조사 없이 다항시간 안에 복원이 가능함을 의미한다. 또한, Fiat-Shamir를 활용한 전자서명의 특성상, 4번째 공격의 공격시도 11,832의 경우 SHA-256을 사용한다면 각 서명당 256개의 랜덤한 벡터 f가 사용되기 때문에 이는 실제로 대략 46개의 서명값을 획득하면 개인키가 완전히 복원된다는 것을 의미한다.

Table 2. Attack on original CRS signature

	1	2	3	4
n	74	74	74	74
B	5	5	5	5
d	5	10	100	1,000
attempts	60	131	1,380	11,832
Success	1	1	1	1

다음 〔Table 3〕는 rejection sampling이 사용될 경우에 공격복잡도를 나타낸다. 〔Table 3〕에서 **rej**는 $f-e$가 〔-rej,rej〕일 경우에만 공개되는 것을 의미한다.

〔Table 3〕에서 제시된 대로 rejection sampling을 사용할 경우 개인키를 복원하는데 성공하는 경우는 없음을 의미한다. 또한, 실제로 공격을 분석한 결과, 〔Table 2〕의 공격과 다르게, 〔Table 3〕에서는 가능한 개인키의 값으로 $[-B,B]$에 해당하는 모든 정수값이 저장되어, $f-e$를 통해 개인키에 대한 유의미한 결과를 얻어낼 수 없다는 것이 확인되었다.

Table 3. Attack results when rejection sampling is used

	1	2	3	4
n	74	74	74	74
B	5	5	5	5
d	55	105	505	1,005
rej	50	100	500	1,000
attempts	10^6	10^6	10^6	10^6
Success	0	0	0	0

3.2.2.2 제안하는 최적 파라미터

〔Table 3〕에서 확인할 수 있듯이, 랜덤한 벡터 f의 성분의 범위를 〔-55,55〕에 설정하여도 안전성에 문제는 되지 않지만, 실제 SeaSign 기반 전자서명에서는 해당 범위를 사용하지 않는다. 그 이유는 rejection sampling으로, n차원의 벡터에 대해 n개의 성분이 모두 〔-rej,rej〕안에 들어야 rejection 없이 서명이 가능해져서, 〔Table 3〕의 1에 해당하는 파라미터를 사용할 경우 각 성분에 대한 accept할 확률은 90%로, 전체 nt개의 성분이 다 성공할 확률은 9.2×10^{-195}정도로 매우 낮다. 따라서 d를 충분히 크게 하여 rejection rate를 허용 범위내로 설정한 부분이 SeaSign에서 제안한 파라미터이다.

하지만, 본 논문에서 제안한 Algorithm 3 방법으로 서명과정을 진행할 경우, 미리 범위 안에 들어있는 t개의 랜덤한 백터들을 선택하기 때문에 rejection이 일어나지 않아 δ의 크기를 효율적으로 작게 조정할 수 있다. 고려해야 할 부분은 δ의 크기가 작아 효율적이면서 $[-(\delta+1)B,(\delta+1)B]$가 충분한 서명 횟수에도 중복되는 벡터 f가 없도록 커야 한다. 마지막으로 rejection이 없더라도 t개의 벡터에 대한 총 $n\cdot t$개의 $[-(\delta+1)B,(\delta+1)B]$에서 선택된 성분들이 $[-(\delta-1)B,(\delta+1)B]$ 에 있도록 선택하는데 시간이 짧은 δ를 선택해야 한다. δ를 변화하며 성공률과 선택 시간을 확인한 결과 제안하는 두 개의 파라미터는 다음과 같다. 〔Table 4〕에서 rate은 벡터의 한 성분이 정해진 범위 안에 있을 확률을 의미한다.

Table 4. Proposed parameter for SeaSign-512 optimization

	B	δ	**rate**
Ours 1	5	100	99.09
Ours 2	5	90	98.90

IV. 구현 결과

본 장에서는 SeaSign 구현에 대한 비교를 제시한다. 본 논문에서는 기존 SeaSign, 〔11〕에서의 최적화, 그리고 본 논문에서의 최적화된 SeaSign의 총 세 가지의 버전의 서명 시간을 비교한다. 측정에 사용한 CPU는 3.40GHz의 동작 주파수를 가지는 Intel Core i7-6700를 사용했으며, Ubuntu 18.04.2 LTS 운영체제상에서 최적화 옵션 -O3과 clang-6.0.0 컴파일러를 이용했다.

4.1 SeaSign 파라미터

SeaSign기반 전자서명의 성능 비교를 위해 공통적으로 73개의 서로 다른 홀수 소수를 인수로 가지는 다음 511비트 소수를 사용하였다,

$$p=2^4\cdot3^3\cdot5\cdot7\cdot11^2\cdot13\cdots373-1$$

이 유한체 위에서 supersingular 몽고메리 곡선

$$M_0 : y^2 = x^3 + x$$

을 시작 곡선으로 사용하였으며, 각각의 최적화 버전에 대한 추가적인 파라미터는 다음과 같다.

Table 5. Parameters for comparison

	B	t	δ	u
SeaSign [8]	5	128	9,472	-
[11]	5	337	114	79
Ours 1	5	128	100	-
Ours 2	5	128	90	-

4.2 SeaSign 구현 결과

제시한 파라미터를 활용하여 구현한 SeaSign의 서명 생성 속도는 다음과 같다. 키 생성의 경우 모든 SeaSign 버전이 동일하기 때문에 한 번만 측정하였다. [Table 6] 에서의 Rej. 은 rejection 횟수를 의미한다.

Table 6. Implementation results

	SeaSign [8]	[11]	Ours1	Ours2
KeyGen	0.011 ms			
SigGen	120,481.31 s	2207.77s	321.13s	290.24s
SigVer	30,302.20 s	1,201.23s	320.60s	287.22s
Rej.	3	1	0	0

위의 표에서 확인할 수 있듯이, Ours1은 기존 SeaSign보다 약 290배 빠르고, [11]에서 제시된 방법보다 약 7.47배 빠르다. 특히, 본 논문에서 제안된 파라미터의 경우 rejection rate이 높기 때문에, rejection 이 일어나는 SeaSign 및 [11]의 제안방법에는 사용할 수 없고 본 논문의 변형된 서명 알고리즘에서만 사용할 수 있다는 점에서 장점을 가지게 된다. 특히, 기존 SeaSign의 경우 서명하는데 하루 정도 걸리는 반면, 본 논문에서 제안된 방법과 파라미터를 사용할 경우 5분 정도 걸리게 된다.

V. 결 론

본 논문에서는 rejection sampling의 변형을 통해서 rejection 없이 효율적이며 안전한 서명이 가능한 새로운 SeaSign 서명 방법을 제안하였다. 제안한 방법과 기존 SeaSign과 비교한 결과, 기존 SeaSign 대비 3배나 빠른 속도를 얻을 수 있었다. 추가로, CRS 기반 전자 서명 알고리즘에 대한 분석 도구를 개발하여, 본 논문에서 제안하는 알고리즘에 특화된 파라미터를 설정하였다. 해당 파라미터로 구현을 진행한 결과, 전자서명에 5분정도 걸리게 된다. 아직 다른 PQC 기반 암호에 비해서는 느린 속도이지만, SeaSign 기반 전자서명이 기존 1일 정도 걸린 것과 비교해보면 큰 향상이다. 한편, [11]에 추가로 제안한 여러 비트씩 암호화를 진행한다면 공개키의 크기는 커지지만, 서명 생성 속도는 더 향상될 수 있다. 앞으로는 여러 비트를 사용하는 환경에서도 추가로 안전성 분석을 통해 최적인 파라미터를 제안하고, 서명 검증과정의 최적화에 관한 연구를 진행할 예정이다.

References

[1] J.M. Couveignes, "Hard homogenous spaces," IACR Cryptology ePrint Archive, 2006:291, 2006

[2] A. Childs et al. "Constructing elliptic curve isogenies in quantum subexponential time," Journal of Mathematical Cryptology, vol. 8, no. 1, pp. 1-29, 2014

[3] D. Jao, L. De Feo "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies," PQCrypto, LNCS 7071, pp. 19-34, Aug. 2011

[4] W. Castryck and Tomas Decru, "An efficient key recovery attack on SIDH," Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 423-447, 2023

[5] W. Castryck et al. "CSIDH: An efficient post-quantum commutaitve

group action," ASIACRYPT, LNCS 11274, Dec. 2018
〔6〕 Y. Yoo et al. "A post-quantum digital signature scheme based on supersingular isogenies," FC 2017, pp. 163-181, 2017
〔7〕 S. Galbraith et al. "Identification protocols and signature schemes based on supersingular isogeny problems," ASIACRYPT 2017, pp. 3-33, 2017
〔8〕 L. De Feo and Steven Galbrith, "SeaSign: compact isogeny signatures from class group actions," EUROCRYPT 2019, pp. 759-789, 2019
〔9〕 W. Beullens et al. "CSI-FiSh: efficient isogeny based signatures through class group actions," International Conference on the Theory and Application of Cryptology and Information Security, pp. 227-247, 2019
〔10〕 Aurel Page and Damien Robert, "Introducing Clapoti(s): Evaluating the isogeny class group action in polynomial time," IACR Cryptology ePrint Archive, 2023:1766, 2023
〔11〕 T. Decru et al. "Faster SeaSign signatures through improved rejection sampling," PQCrypto 2019, pp. 271-285, 2019

〈저자소개〉

김 수 리 (Suhri Kim) 종신회원
2014년 2월: 고려대학교 수학과 이학사
2016년 8월: 고려대학교 정보보호대학원 공학석사
2020년 2월: 고려대학교 정보보호대학원 공학박사
2020년 3월~2021년 2월: 고려대학교 정보보호대학원 연구교수
2020년 7월~2021년 2월: KU Leuven ESAT/COSIC 박사후연구원
2021년 3월~현재: 성신여자대학교 수리통계데이터사이언스학부 조교수
〈관심분야〉 공개키 암호시스템, 후양자암호학