Article

Post-quantum Two-party Adaptor Signature Based on Coding Theory

Jean Belo KLAMTI (ID) **and M. Anwar HASAN** (ID)

Department of Electrical and Computer Engineering, University of Waterloo, 200 University Ave W, Waterloo, ON, N2L 3G1; jbklamti@uwaterloo.ca and ahasan@uwaterloo.ca

* Correspondence: jbklamti@uwaterloo.ca
† Current address: University of Waterloo
‡ These authors contributed equally to this work.

Abstract: An *adaptor signature* can be viewed as a *signature* concealed with a *secret value* and, by design, any two of the trio yield the other. In a multiparty setting, an initial adaptor signature allows each party create additional adaptor signatures without the original secret. Adaptor signatures help address scalability and interoperability issues in blockchain. They can also bring some important advantages to cryptocurrencies, such as low on-chain cost, improved transaction fungibility, and less limitations of a blockchain's scripting language. In this paper, we propose a new two-party adaptor signature scheme that relies on quantum-safe hard problems in coding theory. The proposed scheme uses a hash-and-sign code-based signature scheme introduced by Debris-Alazard et al. and a code-based hard relation defined from the well-known syndrome decoding problem. To achieve all the basic properties of adaptor signatures formalized by Aumayr et al., we introduce further modifications to the aforementioned signature scheme. We also give a security analysis of our scheme and its application to the *atomic swap*. After providing a set of parameters for our scheme, we show that it has the smallest pre-signature size compared to existing post-quantum adaptor signatures.

Keywords: Post-quantum cryptography; Blockchain; Code-based cryptography; Adaptor signature; Scriptless scripts.

1. Introduction

In cryptocurrencies and other blockchain applications, transactions are validated by miners using decentralized consensus protocols. A transaction is akin to an application formed by scripts. The scripting language of a blockchain allows the encoding of potential functionalities and rules that make a transaction valid. Therefore, the fee for a transaction corresponds to the storage and computational cost of executing the transaction's script by a miner. The fee sometimes could be excessively high for some cryptocurrencies with a scripting language that enables a more complex transaction logic. In addition to the high fee issue, the public verifiability feature of transactions and the permissionless nature of consensus protocols pose some other challenges with regard to scalability, privacy and transaction throughput.

The main approach to addressing the aforementioned issues is to reduce the size of on-chain transactions by handing off some transactions to off-chains. The goal here is to use as few scripts as possible for on-chain transactions. To this end, Poelstra [1] introduced a technique called *scriptless script* that enables us to create smart contracts without a script. The technique was later formalized as an adaptor signature by Fournier [2]. Recently, Aumayr et al. [3] have presented a full formalization of the adaptor signature as a cryptographic primitive.

Adaptor signature is a two-step signing algorithm bound to a secret. It is defined from a digital signature scheme and a hard relation. In adaptor signature, the first *pre-signature* is generated by a user with knowledge of a witness of the hard relation.

The complete signature reveals the witness and can be verified by its corresponding verification algorithm. In blockchain applications, adaptor signatures bring some advantages to cryptocurrencies such that a reduction of on-chain cost and an improvement of each transaction's fungibility.

Since the work by Poelstra, several articles on adaptor signature have appeared, e.g., see [3–10]. In [3], the authors have introduced two adaptor signature schemes based on the Schnorr and ECDSA digital signatures, respectively. Authors of [5] have showed that signature schemes that are constructed from identification schemes with some additional homomorphic properties, can be transformed into adaptor signature schemes. In [7], the authors have showed how to provide an adaptor signature instance from any one-way homomorphic function. In [5] (respectively [10]), the authors have designed a post-quantum adaptor signature based on lattices (respectively isogenies).

1.0.1. Motivation

Adaptor signature is one of the central primitives in today's cryptocurrency-based payment ecosystem. A few exceptions aside, most of the existing adaptor signature schemes will however be broken with the arrival of sufficiently large quantum computers. Thus it is important to explore various ways to design efficient adaptor signature schemes that are quantum-safe. Code-based cryptography, which has been studied for many years, is considered resistant against quantum-computer attacks and is one of the finalists in the current post-quantum cryptography (PQC) standardization process undertaken by the National Institute of Standards and Technology (NIST). To our knowledge, no adaptor signature scheme based on coding theory exists in the literature. Therefore, even if key sizes are large in code-based cryptography, designing a code-based adaptor signature is of interest to ensure the post-quantum security of blockchain applications.

1.0.2. Our contributions

In this paper, we present a post-quantum adaptor signature scheme using cryptographic assumptions rooted in coding theory. To design our scheme, we use a hash-and-sign code-based signature scheme, called *Wave*, which was introduced by Debris-Alazard et al. [11]. The hard relation used in our scheme is defined from the well-known NP-complete problem in coding theory. However, in order to achieve the pre-signature correctness and pre-signature adaptability for adaptor signatures, we introduce a few modifications to Wave. After designing our scheme, we show that it satisfies the pre-signature correctness and the pre-signature adaptability property of adaptor. We present a security analysis of our scheme and compare the latter with existing post-quantum adaptor signature schemes. We also give an application of our scheme to the *atomic swap*.

1.0.3. Organization

The remainder of the paper is organized as follows. Section 2 provides some preliminaries on coding theory and adaptor signature. In Section 3, we present the design of our code-based adaptor signature scheme and its security analysis. In Section 4, we provide a set of parameters for our scheme and its comparison with other post-quantum adaptor signature schemes. In Section 5, we give the application of our scheme in atomic swap. Finally, we conclude in Section 6.

2. Preliminaries

2.1. Coding theory

Let $\mathbb{F}$ be the finite field $\mathbb{F}_q$ with $q = p^m$ and p a prime number. A linear code $\mathcal{C}$ of length n and dimension k over $\mathbb{F}$ is a vector subspace of dimension k of $\mathbb{F}^n$. It can be specified by a full rank matrix $\mathbf{G} \in \mathbb{F}^{k \times n}$ called generator matrix. The rows of $\mathbf{G}$ span the code $\mathcal{C}$. Specifically, a linear code can be defined by its generator matrix as follows:

$$\mathcal{C} = \left\{ \boldsymbol{m}\mathbf{G} \ \text{s.t.} \ \boldsymbol{m} \in \mathbb{F}^k \right\}$$

A linear code can be also defined by the right kernel of matrix $\mathbf{H}$ called *parity-check matrix* of $\mathcal{C}$:

$$\mathcal{C} = \left\{ x \in \mathbb{F}^n \text{ s.t. } x\mathbf{H}^T = \mathbf{0} \right\}$$

The Hamming distance between two codewords is the number of positions (coordinates) where they differ. The minimal distance of a code is the minimal distance of all codewords.

The weight of a word/vector $x \in \mathbb{F}^n$ denoted by $wt(x)$ is the number of its non-zero positions. Then the minimal *weight* of a code $\mathcal{C}$ is the minimal weight of all non-zero codewords. In the case of linear code $\mathcal{C}$, its minimal distance is equal to the minimal weight of the code.

In this paper, the set of vectors of length n and weight ω is denoted by $\mathcal{S}_{q,n,\omega} = \{x \in \mathbb{F}^n \text{ s.t. } wt(x) = \omega\}$. For two given integers a and b, where $a < b < n$, we denote the set of vectors of length n with $wt(x) \in [a, b]$ by $\mathcal{S}_{q,n,[a,b]} = \{x \in \mathbb{F}^n \text{ s.t. } a \leq wt(x) \leq b\}$.

2.2. Hard problems in coding theory

In this subsection, we recall some NP-complete problems in coding theory.

Problem 1. *(Binary Syndrome Decoding (SD) problem)*
Input: A matrix $\mathbf{H} \in \mathbb{F}_2^{r \times n}$, a vector $s \in \mathbb{F}_2^r$, and an integer $\omega > 0$.
Output: A vector $y \in \mathbb{F}_2^n$ such that $wt(y) \leq \omega$ and $s = y\mathbf{H}^T$.

The SD problem was proved to be NP-complete in 1978 by McEliece and Van Tilbord [12]. Some of its instances can be solved in polynomial time, depending on the input. In particular, when the parameter ω is in the interval $\left[\frac{r}{2}, n - \frac{r}{2}\right]$, solving it becomes easy – first determine a pseudo-inverse $\mathbf{H}^{-1}$ of the matrix $\mathbf{H}$ and then compute the product $s\mathbf{H}^{-1}$ to return a valid solution with a high probability. However, when the value of the parameter ω is not in $\left[\frac{r}{2}, n - \frac{r}{2}\right]$, if a single solution exists, finding it is much harder. For non-binary finite field $\mathbb{F}_q$, the corresponding interval is given by $\left[\frac{(q-1)r}{q}, n - \frac{r}{q}\right]$ [11]. We now give the following definition:

Definition 1. *Let n, k, and ω be non-zero integers. Let $\mathbf{H} \in \mathbb{F}_q^{r \times n}$ be a matrix where $r = n - k$. Let $e \in \mathcal{S}_{q,n,\omega}$ be an error vector such that $s = e\mathbf{H}^T$. We say that an instance of a syndrome decoding problem is ϵ-hard if for all probabilistic polynomial time (PPT) algorithm $\mathcal{A}$ with input $(\mathbf{H}, s)$ we have:*

$$\Pr[e \longleftarrow \mathcal{A}(\mathbf{H}, s)] \leq \epsilon$$

The syndrome decoding problem is equivalent to the following problem:

Problem 2. *(General Decoding (GBD) problem)*
Input: A matrix $\mathbf{G} \in \mathbb{F}_q^{k \times n}$, a vector $y \in \mathbb{F}_q^n$, and an integer $\omega > 0$.
Output: Two vectors $m \in \mathbb{F}_q^k$ and $e \in \mathbb{F}_q^n$ such that $wt(e) = \omega$ and $y = m\mathbf{G} + e$.

Problem 3. *(Generalized $(U, U + V)$ code distinguishing problem.)*
Input: A matrix $\mathbf{H} \in \mathbb{F}_q^{r \times n}$.
Output: Decide whether $\mathbf{H}$ is a parity-check matrix of a generalized $(U, U + V)$ code.

Problem 3 is one of the problems on which the security assumption of our adaptor signature scheme relies. It is hard in the worst case and for more information about its hardness or NP-completeness, we refer the reader to [11,13].

The following problem is used in the security proof of the underlying signature scheme that we use in this paper. It was first considered by Johansson and Jonsson in [14] and was analysed later by Sendrier in [15].

Problem 4. *(Decoding One Out of Many (DOOM) problem)*
Input: A matrix $\mathbf{H} \in \mathbb{F}_q^{r \times n}$, *a set of vectors* $s_1, s_2,...,s_N \in \mathbb{F}_q^r$ *and an integer* ω.
Output: A vector $e \in \mathbb{F}_q^n$ *and an integer* i *such that* $1 \leq i \leq N$, $wt(e) = \omega$ *and* $s_i = e\mathbf{H}^T$.

2.3. Hard relation

A hard relation is a relation $\mathcal{R}$ with a statement-witness pair such that:

- there is a PPT algorithm $\mathsf{Gen}(1^{\check{}})$ with input the security parameter λ and output a statement-witness pair (Y, x)
- the relation $\mathcal{R}$ is in poly-time decidable
- for all PPT adversaries $\mathcal{A}$, there is a negligible function ϵ such that

$$\Pr\left[(Y, x^*) \in \mathcal{R} \;\middle|\; \begin{array}{c} (Y, x) \longleftarrow \mathsf{Gen}(1^\lambda) \\ x^* = \mathcal{A}(Y) \end{array}\right] \leq \epsilon(\lambda)$$

The language associated to the relation $\mathcal{R}$ is the set denoted by $L_{\mathcal{R}}$ and defined by:

$$L_{\mathcal{R}} = \{Y \mid \exists x \text{ s.t. } (Y, x) \in \mathcal{R}\}$$

2.4. Code-based signature scheme

The first secure code-based signature is due to Courtois et al. (CFS) [16]. It uses two security assumptions: the indistinguishability of random binary linear codes and the hardness of syndrome decoding problem. This scheme is not considered practical due to the difficulty of finding a random decodable syndrome. It was later modified by Dallot [17] and became to be known as the mCFS (modified Courtois-Finiasz-Sendrier) signature scheme. One of the security assumptions in mCFS is the indistinguishability of random Goppa binary codes. This has led to the emergence of an attack [18]. Currently, the latest code-based signature scheme of this type is due to Debris-Alazard et al. [11]. Their scheme is called Wave and is based on generalized $(U, U + V)$ codes over $\mathbb{F}_q$ with $q \geq 3$. Wave is currently one of the secure and efficient code-based signature schemes designed from a framework other than the Fiat-Shamir transformation. A description of Wave is given in Figure 1.

Common parameters: Length n, dimensional k_U (resp. k_V) of U (resp. V), vector error weight ω, a cryptographic hash function $\mathcal{H} : \{0; 1\}^* \longrightarrow \mathbb{F}_3^{n-k}$, where $k = k_U + k_V$

Secret Key: $\mathsf{sk} := (\mathbf{S}, \mathbf{H}_{\mathsf{sk}}, \mathbf{P})$ where $\mathbf{S} \in \mathbb{F}_q^{(n-k) \times (n-k)}$ is an invertible matrix, $\mathbf{H}_{\mathsf{sk}} \in \mathbb{F}_q^{(n-k) \times n}$ a random generalized $(U, U + V)$ code over $\mathbb{F}_3$ of length n and dimension $k = k_u + k_v$, and $\mathbf{P} \in \mathbb{F}_2^{n \times n}$ is a permutation matrix.

Public Key: $\mathsf{pk} := \mathbf{H}_{\mathsf{pk}}$ where $\mathbf{H}_{\mathsf{pk}} = \mathbf{S}\mathbf{H}_{\mathsf{sk}}\mathbf{P}$.

$\mathsf{Sign}(\mathsf{sk}, m)$:

1. $r \xleftarrow{\$} \mathbb{F}_2^\lambda$
2. Compute $v := \mathcal{H}(m\|r)$
3. Compute $e := D_{\mathbf{H}_{\mathsf{sk}}}(v(\mathbf{S}^{-1})^T)$
4. Return $\sigma := (e\mathbf{P}, r)$

$\mathsf{Verif}(\mathsf{pk}, m, \sigma)$:

1. Parse σ as $(\tilde{e}, \tilde{r})$
2. Compute $s := \mathcal{H}(m\|\tilde{r})$
3. if $s \neq \tilde{e}\mathbf{H}_{\mathsf{pk}}^T$ or $wt(\tilde{e}) \neq \omega$:

    Return 0

4. Return 1

Figure 1. *Wave signature scheme* [11]

2.5. Adaptor signature scheme

In this subsection we recall the formal definition of adaptor signature followed by its basic security properties.

Definition 2. *(Adaptor signature [3])*

An adaptor signature $\Pi_{\mathcal{R},\Xi}$ defined with respect to hard relation $\mathcal{R}$ and a digital signature scheme $\Xi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verif})$ is a tuple of four algorithms $(\mathsf{PreSign}, \mathsf{PreVerif}, \mathsf{Adapt}, \mathsf{Ext})$, where

- $\mathsf{PreSign}(\mathsf{sk}, \boldsymbol{m}, \boldsymbol{Y})$ *is a PPT algorithm that takes as input a secret key* sk*, a statement* $\boldsymbol{Y}$ *and a message* $\boldsymbol{m} \in \mathbb{F}_2^*$*, and outputs a pre-signature* $\tilde{\sigma}$
- $\mathsf{PreVerif}(\mathsf{pk}, \boldsymbol{m}, \boldsymbol{Y}, \tilde{\sigma})$ *is a DPT algorithm that takes as input a public key* pk*, a statement* $\boldsymbol{Y}$*, and a pre-signature* $\tilde{\sigma}$*, and produces 0 or 1 as output.*
- $\mathsf{Adapt}(\tilde{\sigma}, \boldsymbol{x})$ *is a DPT algorithm that takes as input a pre-signature* $\tilde{\sigma}$ *and witness* $\boldsymbol{y}$ *and outputs a valid signature* σ*.*
- $\mathsf{Ext}(\boldsymbol{Y}, \sigma, \tilde{\sigma})$ *is a DPT algorithm that on input a signature* σ*, pre-signature* $\tilde{\sigma}$ *and statement* $\boldsymbol{Y} \in L_{\mathcal{R}}$*, outputs a witness* $\boldsymbol{x}$ *such that* $(\boldsymbol{Y}, \boldsymbol{x}) \in \mathcal{R}$*, or the symbol* $\perp$*.*

Note that adaptor signature schemes inherit the key generation, signature and verification algorithms of the underlying signature scheme, and hence acquire the correctness of the standard digital signature scheme. An adaptor signature scheme, however, has to verify some supplementary properties given by the following definitions.

Definition 3. *(Pre-signature correctness [3])*

An adaptor signature $\Pi_{\mathcal{R},\Xi}$ satisfies pre-correctness if for every $\lambda \in \mathbb{N}$, every message $\boldsymbol{m} \in \{0; 1\}^*$ and every statement/witness pair $(\boldsymbol{Y}, \boldsymbol{x}) \in \mathcal{R}$, the following holds:

$$\Pr\left[\begin{array}{c} \mathsf{PreVerif}(\mathsf{pk}, \boldsymbol{m}, \boldsymbol{Y}, \tilde{\sigma}) = 1 \\ \wedge \\ \mathsf{Verif}(\mathsf{pk}, \boldsymbol{m}, \sigma) = 1 \\ \wedge \\ (\boldsymbol{Y}, \boldsymbol{x}') \in \mathcal{R} \end{array} \left| \begin{array}{c} (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{Gen}(1^\lambda) \\ \tilde{\sigma} \leftarrow \mathsf{PreSign}(sk, \boldsymbol{m}, \boldsymbol{Y}) \\ \sigma := \mathsf{Adapt}(\mathsf{pk}, \boldsymbol{x}, \tilde{\sigma}) \\ \boldsymbol{x}' := \mathsf{Ext}(\sigma, \tilde{\sigma}, \boldsymbol{Y}) \end{array} \right. \right] = 1$$

More precisely, the pre-signature correctness states that a valid pre-signature $\tilde{\sigma}$, which is honestly generated w.r.t. a statement $\boldsymbol{Y} \in L_{\mathcal{R}_{\mathsf{H}_{\mathsf{pk}}}}$, could be adapted into a valid signature. From this signature we can extract a witness $\boldsymbol{x}$ for the statement $\boldsymbol{Y}$. The second basic required property for adaptor signature is the pre-signature adaptability. This second one is stronger than the pre-signature correctness property. It is given by the following definition.

Definition 4. *(Pre-signature adaptability)*

An adaptor signature $\Pi_{\mathcal{R},\Xi}$ satisfies pre-signature adaptability if for any $\lambda \in \mathbb{N}$, any message $\boldsymbol{m} \in \{0; 1\}^*$, any statement/witness pair $(\boldsymbol{Y}, \boldsymbol{x}) \in \mathcal{R}$, any key pair $(\mathsf{sk}, \mathsf{pk}) \longleftarrow \mathsf{Gen}(1^\lambda)$ and any pre-signature $\tilde{\sigma}$ with $\mathsf{PreVerif}(\mathsf{pk}, \boldsymbol{m}, \boldsymbol{Y}, \tilde{\sigma}) = 1$, we have:

$$\Pr[\mathsf{Verif}(\mathsf{pk}, \boldsymbol{m}, \mathsf{Adapt}(\mathsf{pk}, \boldsymbol{m}, \tilde{\sigma})] = 1$$

The pre-signature adaptability states that in reality all valid pre-signature w.r.t. a statement $\boldsymbol{Y} \in L_{\mathcal{R}}$ can be adapted to a valid one using a witness $\boldsymbol{x}$ such that $(\boldsymbol{Y}, \boldsymbol{x}) \in \mathcal{R}$.

For an adaptor signature, there are two main required security properties: the unforgeability under chosen message attacks and witness extractability. These security properties are defined formally by Aumayr et al. [3]. Below, we recall the formal definition of the existential unforgeability under chosen message attack for adaptor signature (*a*EUF-CMA) and that of witness extractability.

Definition 5. *(aEUF–CMA security [3])*

An adaptor signature scheme $\Pi_{\mathcal{R},\Xi}$ *is aEUF-CMA secure if for every PPT adversary* $\mathcal{A}$ *there exists a negligible function* ϵ *such that:*

$$\Pr\left[\mathsf{aSigForge}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}(\lambda) = 1\right] \leq \epsilon(\lambda)$$

where $\mathsf{aSigForge}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}$ *is the experiment given below in Figure 2.*

$\mathsf{aSigForge}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}(\lambda)$

1. $\quad \mathcal{Q} := \varnothing$
2. $\quad (\mathsf{sk}, \mathsf{pk}) \longleftarrow \mathsf{Gen}(1^\lambda)$
3. $\quad m \longleftarrow \mathcal{A}^{\mathcal{O}_S(\cdot), \mathcal{O}_{pS}(\cdot, \cdot)}(\mathsf{pk})$
4. $\quad (Y, x) \longleftarrow \mathsf{GenR}(1^\lambda)$
5. $\quad \tilde{\sigma} \longleftarrow \mathsf{PreSign}(\mathsf{pk}, m, Y)$
6. $\quad \sigma \longleftarrow \mathcal{A}^{\mathcal{O}_S(\cdot), \mathcal{O}_{pS}(\cdot, \cdot)}(\tilde{\sigma}, Y)$
7. $\quad$ Return $(m \notin \mathcal{Q} \wedge \mathsf{Verif}(\mathsf{pk}, m, \sigma))$

$\mathcal{O}_S(m)$

1. $\quad \sigma \longleftarrow \mathsf{Sign}(\mathsf{sk}, m)$
2. $\quad \mathcal{Q} := \mathcal{Q} \cup \{m\}$
3. $\quad$ Return σ

$\mathcal{O}_{pS}(m)$

1. $\quad \tilde{\sigma} \longleftarrow \mathsf{PreSign}(\mathsf{sk}, m, Y)$
2. $\quad \mathcal{Q} := \mathcal{Q} \cup \{m\}$
3. $\quad$ Return $\tilde{\sigma}$

Figure 2. *aEUF-CMA game*

The definition of the unforgeability in adaptor signature is similar to that of existential unforgeability under chosen message attacks in the standard digital signature. However, in the case of adaptor signature, there are some additional requirements: even given a pre-signature on m w.r.t. a random statement $Y \in L_{\mathcal{R}}$, producing a forgery σ has to be hard.

The witness extractability experiment and criteria for an adaptor signature are given by the following definitions.

Definition 6. *(Witness extractability [3])*

An adaptor signature scheme $\Pi_{\mathcal{R},\Xi}$ *is witness extractability if for every PPT adversary* $\mathcal{A}$, *there exists a negligible function* ϵ *such that:*

$$\Pr\left[\mathsf{aWitExt}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}(\lambda) = 1\right] \leq \epsilon(\lambda)$$

where $\mathsf{aWitExt}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}$ *is the experiment in Figure 3.*

$\mathsf{aSigForge}_{\mathcal{A},\Pi_{\mathcal{R},\Xi}}(\lambda)$

1. $\quad \mathcal{Q} := \varnothing$
2. $\quad (\mathsf{sk}, \mathsf{pk}) \longleftarrow \mathsf{Gen}(1^\lambda)$
3. $\quad (m, Y) \longleftarrow \mathcal{A}^{\mathcal{O}_S(\cdot), \mathcal{O}_{pS}(\cdot, \cdot)}(\mathsf{pk})$
4. $\quad \tilde{\sigma} \longleftarrow \mathsf{PreSign}(\mathsf{pk}, m, Y)$
5. $\quad \sigma \longleftarrow \mathcal{A}^{\mathcal{O}_S(\cdot), \mathcal{O}_{pS}(\cdot, \cdot)}(\tilde{\sigma}, Y)$
6. $\quad x' \longleftarrow \mathsf{Ext}(\mathsf{pk}, \sigma, \tilde{\sigma}, Y)$
7. $\quad$ Return $(m \notin \mathcal{Q} \wedge (Y, x') \notin \mathcal{R} \wedge$ $\mathsf{Verif}(\mathsf{sk}, m, \sigma))$

$\mathcal{O}_S(m)$

1. $\quad \sigma \longleftarrow \mathsf{Sign}(\mathsf{sk}, m)$
2. $\quad \mathcal{Q} := \mathcal{Q} \cup \{m\}$
3. $\quad$ Return σ

$\mathcal{O}_{pS}(m)$

1. $\quad \tilde{\sigma} \longleftarrow \mathsf{PreSign}(\mathsf{sk}, m, Y)$
2. $\quad \mathcal{Q} := \mathcal{Q} \cup \{m\}$
3. $\quad$ Return $\tilde{\sigma}$

Figure 3. *Witness extractability game*

The main difference between the witness extractability and the *a*EUF-CMA experiment is that in the first one the adversary is allowed to choose a forgery statement Y.

Assuming that the adversary knows a witness for Y, it can therefore generate a valid signature for the forgery message m. Then, it wins when the valid signature does not reveal a witness for Y.

The following is the definition of a secure adaptor signature.

Definition 7. *(Secure adaptor signature) An adaptor signature $\Pi_{\mathcal{R},\Xi}$ is said to be secure, if it is aEUF-CMA secure, pre-signature adaptable and witness extractable.*

3. Code-based adaptor signature

3.1. Description of our scheme

In this section, we present our code-based adaptor signature scheme $\Pi_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}},Wave}$. Its security relies on the hardness of the syndrome decoding problem.

Let $\mathcal{C}$ be a random q-ary linear code of length n, dimension k, with parity-check matrix $\mathbf{H}_{\mathsf{pk}}$ and error correction capability t. Let $\boldsymbol{x} \in \mathcal{S}_{q,n,t}$ and $Y \in \mathbb{F}_q^{n-k}$. Let the relation $\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$ be defined by:

$$\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}} = \{(Y, \boldsymbol{x}) \text{ s.t. } Y = \boldsymbol{x}\mathbf{H}_{\mathsf{pk}}^T \text{ and } wt(\boldsymbol{x}) = t\}$$

We denote the language associated to the relation $\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$ by $L_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}}$, which is defined by:

$$L_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}} = \{Y \in \mathbb{F}_q^{n-k} \mid \exists\, \boldsymbol{x} \in \mathcal{S}_{q,n,t} \text{ s.t. } (Y, \boldsymbol{x}) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}\}$$

For signing a message m in Wave, the sender chooses a random vector $\boldsymbol{r} \in \mathbb{F}_2^{2\lambda}$, computes $\boldsymbol{s} = \mathcal{H}_0(m\|\boldsymbol{r})$ and decodes $\boldsymbol{s}$ by using its secret key to find the error vector $\boldsymbol{e}$ of weight ω such that $\boldsymbol{s} = \boldsymbol{e}\mathbf{H}^T$. Therefore, the signature corresponding to the message m is given by the pair $\sigma = (\boldsymbol{e}\mathbf{P}, \boldsymbol{r})$.

In our scheme, we use the ternary finite field $\mathbb{F}_3$. We also use two different hash functions $\mathcal{H}_0 : \{0;1\}^* \longrightarrow \mathbb{F}_3^{n-k}$ and $\mathcal{H}_1 : \{0;1\}^* \longrightarrow \mathcal{S}_{3,n,\delta}$ for a well chosen value of the integer δ. In the PreSign algorithm of our adaptor signature, we first randomly choose $\boldsymbol{r}$ in $\mathbb{F}_2^{2\lambda}$. Then, for all given $(Y, \boldsymbol{y}) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$, we compute $\boldsymbol{s} = \mathcal{H}_0(m\|Y - \mathcal{H}_1(\boldsymbol{r})\mathbf{H}_{\mathsf{pk}}^T) \in \mathbb{F}_3^{n-k}$ instead of $\boldsymbol{s} = \mathcal{H}_0(m\|\boldsymbol{r})$. The PreVerif algorithm of our scheme is similar to the Verification algorithm Verif of Wave. Indeed, the receiver has to check that the following equality holds.

$$\boldsymbol{e}\mathbf{H}_{\mathsf{pk}}^T = \mathcal{H}_0(m\|Y - \mathcal{H}_1(\boldsymbol{r})\mathbf{H}_{\mathsf{pk}}^T)$$

Compared to Wave, the signature of a message m in our scheme is a pair $\sigma = (\boldsymbol{e}, \boldsymbol{r}')$ with $\boldsymbol{e}\mathbf{H}_{\mathsf{pk}}^T = \mathcal{H}_0(m\|\boldsymbol{r}'\mathbf{H}_{\mathsf{pk}}^T)$ and $\boldsymbol{r}' \in \mathbb{F}_2^n$ instead of $\boldsymbol{e}\mathbf{H}_{\mathsf{pk}}^T = \mathcal{H}_0(m\|\boldsymbol{r}')$ and $\boldsymbol{r}' \in \mathbb{F}_2^{2\lambda}$.

The Adapt algorithm in our scheme takes as input a tuple $((\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}}), \boldsymbol{x})$ and output the pair $(\boldsymbol{e}, \boldsymbol{r})$ where $\boldsymbol{r} = \boldsymbol{x} - \mathcal{H}_1(\tilde{\boldsymbol{r}})$ and $\boldsymbol{e} = \tilde{\boldsymbol{e}}$. To extract the witness corresponding to a statement Y, we execute the algorithm Ext which takes as input $(Y, \tilde{\sigma}, \sigma)$, where $\tilde{\sigma} = (\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}})$ is a pre-signature and $\sigma = (\boldsymbol{e}, \boldsymbol{r})$ is the corresponding signature. The algorithm Ext outputs $\boldsymbol{x}' = \mathcal{H}_1(\tilde{\boldsymbol{r}}) + \boldsymbol{r}$ if $Y = \boldsymbol{x}'\mathbf{H}_{\mathsf{pk}}^T$ and $wt(\boldsymbol{x}') = \omega$. Otherwise, it returns the abort symbol. See Figure 4 for the description of our scheme.

3.2. Security analysis

Before giving the security analysis of our scheme, let us verify the pre-signature correctness and the pre-signature adaptability of our scheme.

Proposition 1. *The code-based adaptor signature $\Pi_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}},Wave}$ described in Figure 4 satisfies the pre-signature adaptability.*

Proof. Let $\mathsf{sk} := (\mathbf{S}, \mathbf{H}_{\mathsf{sk}}, \mathbf{P})$ be an arbitrary secret key. Let $m \in \mathbb{F}_2^*$ be an arbitrary message. Let $\mathsf{pk} := \mathbf{H}_{\mathsf{pk}}$ be the corresponding public key of sk, where $\mathbf{H}_{\mathsf{pk}} := \mathbf{S}\mathbf{H}_{\mathsf{sk}}\mathbf{P}$ and $\mathbf{H}_{\mathsf{sk}}$ is the parity-check matrix of a $(U, U+V)$ code. Let us consider $(Y, \boldsymbol{x}) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$. Let $\tilde{\sigma}$

Common parameters:

Length n, dimensional k_U (resp. k_V) of a U (resp. V), vector error weight ω, witness weight t and integer δ such $0 < |\delta - t|$ and $\delta + t < \frac{2(n-k)}{3}$ where $k = k_U + k_V$. Two cryptographic hash functions $\mathcal{H}_0 : \{0; 1\}^* \longrightarrow \mathbb{F}_3^{n-k}$ and $\mathcal{H}_1 : \{0; 1\}^* \longrightarrow \mathcal{S}_{3,n,\delta}$

Secret Key: sk $:= (\mathbf{S}, \mathbf{H}_{\mathsf{sk}}, \mathbf{P})$, where $\mathbf{H}_{\mathsf{sk}}$ is a parity of a random $(U, U + V)$ code over $\mathbb{F}_3$ of length n, dimension $k = k_u + k_v$ and decoding algorithm $D_{\mathbf{H}_{\mathsf{sk}}}$.

Public Key: pk $:= \mathbf{H}_{\mathsf{pk}}$ where $\mathbf{H}_{\mathsf{pk}} = \mathbf{S}\mathbf{H}_{\mathsf{sk}}\mathbf{P}$, $\mathbf{S} \in \mathbb{F}_3^{(n-k)\times(n-k)}$ is an invertible matrix and $\mathbf{P}$ a permutation matrix of size $n \times n$.

PreSign$((\mathsf{sk}, \mathsf{pk}), \boldsymbol{m}, \boldsymbol{Y})$:

1. $\tilde{\boldsymbol{r}} \xleftarrow{\$} \mathbb{F}_2^{2\lambda}$
2. Compute $\boldsymbol{u} = \mathcal{H}_0(\boldsymbol{m}\|\boldsymbol{Y} - \mathcal{H}_1(\tilde{\boldsymbol{r}})\mathbf{H}_{\mathsf{pk}}^T)$
3. Compute $\tilde{\boldsymbol{s}} := \boldsymbol{u}(\mathbf{S}^{-1})^T$.
4. Compute $\boldsymbol{e} := D_{\mathbf{H}_{\mathsf{sk}}}(\tilde{\boldsymbol{s}})$. [Successful decoding satisfies $wt(\boldsymbol{e}) = \omega$]
5. $\tilde{\boldsymbol{e}} := \boldsymbol{e}\mathbf{P}$
6. Parse $\tilde{\sigma} := (\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}})$

Adapt$(\tilde{\sigma}, \boldsymbol{x})$:

1. Parse $\tilde{\sigma}$ as $(\boldsymbol{e}, \tilde{\boldsymbol{r}})$
2. Compute $\boldsymbol{r}' := \boldsymbol{x} - \mathcal{H}_1(\tilde{\boldsymbol{r}})$
3. Return $\sigma := (\boldsymbol{e}, \boldsymbol{r}')$

PreVerif$(\mathsf{pk}, \boldsymbol{m}, \boldsymbol{Y}, \tilde{\sigma})$:

1. Parse $\tilde{\sigma}$ as $(\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}})$
2. Compute $\boldsymbol{s} := \mathcal{H}_0(\boldsymbol{m}\|\boldsymbol{Y} - \mathcal{H}_1(\tilde{\boldsymbol{r}})\mathbf{H}_{\mathsf{pk}}^T)$
3. If $\boldsymbol{s} \neq \tilde{\boldsymbol{e}}\mathbf{H}_{\mathsf{pk}}^T$

   Return 0
4. Return 1

Ext$(\boldsymbol{Y}, \tilde{\sigma}, \sigma)$:

1. Parse $\tilde{\sigma}$ as $(\tilde{\boldsymbol{e}}\|\tilde{\boldsymbol{r}})$
2. Parse σ as $(\boldsymbol{e}\|\boldsymbol{r})$
3. If $wt(\boldsymbol{r}) \notin [|\delta - t|, \delta + t]$:

   Return $\bot$
4. Compute $\boldsymbol{z} := \tilde{\boldsymbol{r}} + \mathcal{H}_1(\boldsymbol{r})$
5. If $wt(\boldsymbol{z}) \neq t$ or $\boldsymbol{Y} \neq \boldsymbol{z}\mathbf{H}_{\mathsf{pk}}^T$:

   Return $\bot$
6. Return $\boldsymbol{z}$

Figure 4. *Code-based adaptor signature*

be a pre-signature generated w.r.t. $\boldsymbol{Y}$. Then $\tilde{\sigma}$ is the tuple $(\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}})$ so if PreVerif$(\mathsf{pk}, \boldsymbol{m}, \boldsymbol{Y}, \tilde{\sigma}) = 1$, we know that $\tilde{\boldsymbol{e}}$ is actually computed by the owner of the secret key sk.

According to the design of Adapt, we have $(\boldsymbol{e}, \boldsymbol{r}) := \mathsf{Adapt}(\tilde{\sigma} := (\tilde{\boldsymbol{e}}, \tilde{\boldsymbol{r}}), \boldsymbol{x})$ where $\boldsymbol{r} := \boldsymbol{x} - \mathcal{H}_1(\tilde{\boldsymbol{r}})$. We can therefore verify that

$$
\begin{aligned}
\mathcal{H}_0(\boldsymbol{m}\|\boldsymbol{r}\mathbf{H}_{\mathsf{pk}}^T) &= \mathcal{H}_0(\boldsymbol{m}\|(\boldsymbol{x} - \mathcal{H}_1(\tilde{\boldsymbol{r}}))\mathbf{H}_{\mathsf{pk}}^T) \\
&= \mathcal{H}_0(\boldsymbol{m}\|\boldsymbol{x}\mathbf{H}_{\mathsf{pk}}^T - \mathcal{H}_1(\tilde{\boldsymbol{r}})\mathbf{H}_{\mathsf{pk}}^T) \\
&= \mathcal{H}_0(\boldsymbol{m}\|\boldsymbol{Y} - \mathcal{H}_1(\tilde{\boldsymbol{r}})\mathbf{H}_{\mathsf{pk}}^T) \\
&= \tilde{\boldsymbol{e}}\mathbf{H}_{\mathsf{pk}}^T
\end{aligned}
$$

$\square$

Proposition 2. *The code-based adaptor signature $\Pi_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}, Wave}$ described in Figure 4 satisfies the pre-signature correctness.*

Proof. Let sk $:= (\mathbf{S}, \mathbf{H}_{\mathsf{sk}}, \mathbf{P})$ be a secret key. Let $\boldsymbol{m} \in \mathbb{F}_2^*$ be an arbitrary message. Let pk $= \mathbf{H}_{\mathsf{pk}}$ be the corresponding public key linked to sk, where $\mathbf{H}_{\mathsf{pk}} := \mathbf{S}\mathbf{H}_{\mathsf{sk}}\mathbf{P}$ and $\mathbf{H}_{\mathsf{sk}}$ is the parity-check matrix of a $(U, U + V)$ code with decoding algorithm $D_{\mathbf{H}_0}$. Let us consider $(\boldsymbol{Y}, \boldsymbol{x}) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$.

Using the public key pk (respectively the secret key sk), we can compute the syndrome $s := \mathcal{H}_0(m \| Y - \mathcal{H}_1(\tilde{r})\mathbf{H}_{\mathsf{pk}}^T)$ (respectively the corresponding error vector $\tilde{e}' := D_{\mathbf{H}_{\mathsf{sk}}}(s(\mathbf{U}^{-1})^T))$. Therefore, the pre-signature of the message m is given by $\tilde{\sigma} := (\tilde{e}, \tilde{r})$, where $\tilde{e} = \tilde{e}'\mathbf{P}$. For the pre-verification, we have to check the following equality:

$$\tilde{e}\mathbf{H}_{\mathsf{pk}}^T = \mathcal{H}_0(m \| Y - \mathcal{H}_1(\tilde{r})\mathbf{H}_{\mathsf{pk}}^T) \tag{1}$$

When $\tilde{e}$ is honestly computed, equality (1) always holds and then $\mathsf{PreVerif}(\mathsf{pk}, m, Y, \tilde{\sigma}) = 1$.

According to Figure 4, the output of the adaptor algorithm is given by $\sigma = (e, r) = \mathsf{Adapt}(\tilde{\sigma}, x)$, where $r = x - \mathcal{H}_1(\tilde{r})$ with $(Y, x) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$ and that of the extractor algorithm is given by $\mathcal{H}_1(\tilde{r}) + r = x - \mathcal{H}_1(\tilde{r}) + \mathcal{H}_1(\tilde{r}) = x$ with $(Y, x) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$. The fact that $\tilde{e}$ is honestly computed, we have

$$\mathcal{H}_0(m \| r\mathbf{H}_{\mathsf{pk}}^T) = \mathcal{H}_0(m \| (x - \mathcal{H}_1(\tilde{r}))\mathbf{H}_{\mathsf{pk}}^T) = \mathcal{H}_0(m \| Y - \mathcal{H}_1(\tilde{r})\mathbf{H}_{\mathsf{pk}}^T) = \tilde{e}$$

Therefore, in our scheme, $\sigma = (e, r)$ is a valid signature for the message m. $\quad\square$

For the security analysis of the scheme, below we state the assumptions which the security of our scheme relies on:

Assumption 1: The advantage of probabilistic polynomial time algorithm $\mathcal{A}$ to solve the syndrome decoding problem is negligible with respect to the length n and the dimension k of the code.

Assumption 2: The advantage of probabilistic polynomial time algorithm $\mathcal{A}$ to solve $(U, U + V)$ code distinguishing problem is negligible with respect to the length n and dimension k of the code.

Assumption 3: The advantage of probabilistic polynomial time algorithm $\mathcal{A}$ to solve the decoding out of many (DOOM) problem is negligible with respect to the length n and dimension k of the code.

Under Assumption 1, the relation $\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$ defined in Subsection 3.1 is hard relation and under Assumptions 1 and 2 the Wave signature is EUF-CMA secure [11]. Therefore we have the following.

Theorem 1. *(aEUF-CMA Security) Under Assumptions 1, 2 and 3, the code-based adaptor $\Pi_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}, Wave}$ defined in Figure 4 is aEUF-CMA secure.*

Proof. Let $\mathcal{A}$ be an adversary against our scheme in the aEUF-CMA game. Let ϵ_{aCMA} be the probability that a PPT adversary wins against our scheme in the aEUF-CMA game. The proof of Theorem 1 consists of coming up with a bound for the adversary advantage $\mathsf{Adv}(\mathcal{A})$. Suppose that there is a PPT adversary $\mathcal{A}$ which attacks the aEUF-CMA security of our code-based adaptor signature. That means that $\mathcal{A}$ is able to forge a valid signature $\sigma' = (e', r')$ on a targeted message m^* after receiving the pair pre-signature/statement $(\tilde{\sigma}, Y)$ from the challenger. $\tilde{\sigma} = (\tilde{e}, \tilde{r})$ is a pre-signature w.r.t. Y of the target message m^* sending to the challenger by $\mathcal{A}$. Let $\sigma = (e, r)$ be a valid signature obtained w.r.t. the witness x of the Y after executing the adaptor algorithm, i.e., $r = x - \mathcal{H}_1(\tilde{r})$. If σ' is a valid signature, then we have either $\sigma' = \sigma$ or $\sigma' \neq \sigma$.

- If $\sigma' = \sigma$, which is equivalent to $(e', r') = (e, r)$, then $\mathcal{A}$ is able to find $r' \in \mathcal{S}_{3,n,[|\delta-t|, \delta+t]}$, such that $r' = r = x - \mathcal{H}_1(\tilde{r})$ for a given $\tilde{r}$ and Y such that $Y = x\mathbf{H}_{\mathsf{pk}}^T$. The best way to find such a vector r' is to solve the equation $Y = x\mathbf{H}_{\mathsf{pk}}^T$, i.e., to solve a hard instance of syndrome decoding problem.
- If $\sigma' \neq \sigma$, we have two cases:

★　　$e' = e$ and $r' \neq r$: this case implies that

$$e'\mathbf{H}_{\mathsf{pk}}^T = \mathcal{H}_0(m^* \| r'\mathbf{H}_{\mathsf{pk}}^T) = \mathcal{H}_0(m^* \| r\mathbf{H}_{\mathsf{pk}}^T) = e\mathbf{H}_{\mathsf{pk}}^T$$

It means that either we have $r'\mathbf{H}_{\mathsf{pk}}^T = r\mathbf{H}_{\mathsf{pk}}^T$ or $\mathcal{A}$ is able to find a collision of the hash function $\mathcal{H}_0$. With collision resistant of the hash function $\mathcal{H}_0$, the probability that this case happen is less than

$$\frac{1}{3^{n-k}} + \nu(\lambda)$$

where $\frac{1}{3^{n-k}}$ is the probability for having the equality $r'\mathbf{H}_{\mathsf{pk}}^T = r\mathbf{H}_{\mathsf{pk}}^T$ (see [11]).

★　　$e' \neq e$ and $r \neq r'$: this last case means that the adversary $\mathcal{A}$ is able to forge a valid signature using the modify version of Wave that we use in our scheme which is EUF-CMA secure.

By putting it all together, we have have

$$\mathsf{Adv}(\mathcal{A}) \leq \frac{1}{3^{n-k}} + \mathsf{Adv}_{SD} + \mathsf{Adv}_{Wave} + \nu(\lambda)$$

where $\mathsf{Adv}_{\mathsf{Wave}}$ is the advantage of an adversary against Wave in EUF-CMA game and $\mathsf{Adv}_{\mathsf{SD}}$ is that for solving the syndrome decoding problem.　□

Theorem 2. *(Witness Extractability) Under Assumptions 2 and 3, the code-based adaptor* $\Pi_{\mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}, Wave}$ *defined in Figure 4 is witness extractable.*

Proof. Let $\tilde{\sigma} = (\tilde{e}, \tilde{r})$ be the pre-signature correctly computed w.r.t. a statement Y. Let $\sigma' = (e', r')$ be a valid signature. Let $x' = \mathsf{Ext}(Y, \tilde{\sigma}, \sigma)$ be the witness extracted from $\tilde{\sigma}$ and σ. According to the algorithm Ext in our scheme, we have $wt(x') = t$ and $Y = x'\mathbf{H}_{\mathsf{pk}}^T$. That means if Ext outputs x', we have $(Y, x') \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$ with a high probability.

Let $\sigma = (e, r)$ be a valid signature computed w.r.t. $\tilde{\sigma}$ by the honest witness owner. The fact that in the witness extractability game, we should have $(Y, x') \notin \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$, we have:

$$(Y, x') \notin \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}} \implies r \neq r' \implies \sigma \neq \sigma'.$$

Therefore, the rest of the proof corresponds to the second part of the proof of Theorem 1　□

4. Parameter set and experimental results
Parameter values and signature sizes:

Referring to Figure 4 and [19], we can see that the length of pre-signature is given by $|\tilde{\sigma}| = k + 2\lambda$ and that of the signature is given by $|\sigma| = k + n$. By using parameters of the Wave scheme [11,19], we can determine the exact sizes of the pre-signature and the signature as given in Table 1. These parameters correspond to security Level 1 of NIST PQC standard.

δ	λ	q	n	ω	t	k_U	k_V	d	PreS.	Sign.
517	128	3	8492	7980	12	3558	2047	81	1143	2793

Table 1: Parameters setting of our scheme [11]

Using the above-mentioned parameter values, we give in Table 2 a numerical comparison of the pre-signature and signature sizes of our scheme with those of [5,10]. In the table, we see that for these parameter values our scheme has a shorter pre-signature size but a slightly larger signature size. Specifically, for the parameter values in Table

1, the pre-signature size of the scheme described in Figure 4 is more than 16x and 2.8x smaller than those in [10] and [5], respectively. On the other hand, the signature size of the proposed scheme is 1.03x and 1.5x lager than those of [5] and [10], respectively.

Post-quantum adaptor signature	Pre-signature	Signature
Paper [10]	$18327 \leq |\tilde{\sigma}| \leq 19944$	$263 \leq |\sigma| \leq 1880$
Paper [5]	$|\tilde{\sigma}| = 3210$	$|\sigma| = 2701$
Our paper	$|\tilde{\sigma}| = 1143$	$|\sigma| = 2793$

Table 2: Comparison of pre-signature and signature sizes (in bytes) using parameters of [11,19]

Software prototype:

We have implemented the proposed scheme in software using the C programming language. For this, we adapted the source code of Bamegas et al. [19] by including necessary add-ons, such as our code for the adaptor and extractor algorithms, generation of random vectors of a given weight and transformation of their signature scheme to our pre-signature algorithm. The timing results of the execution of the code is based on VM intel@ core i7-1065G7CPU@1.30GHZx2 with 4GB RAM under Ubuntu 20.10 64-bit. The code is compiled with GCC 10.3. The source code is available at https://github.com/klambel-hash/Code-based-Adaptor-Signature.

	Key Gen.	Presig.	Preverif.	Adapt.	Extract.
Time in ms	3532	814	260	0.248	0.180

Table 3: Timing results of the proposed code-based adaptor signature

5. An application of code-based adaptor signature

In this section, we provide an example blockchain application, namely atomic swap, utilizing our adaptor signature. For this, we assume that the underlying blockchain is using the *m*CFS signature based on coding theory.

5.1. Atomic swap in a nutshell

Atomic swap is a peer-to-peer protocol which allows two different users to exchange cryptocurrencies without a trusted party. Its main goal is to allow an exchange of cryptocurrencies from two different blockchains.

During the atomic swap process, users have full ownership and control of their respective private keys. When one of the participants aborts a transaction or doesn't correctly fulfill the atomic swap process, funds are automatically returned to their original owners. This is possible in an atomic swap because of the use of a particular contract called hash timelock contract (HTLC). The main feature of HTLC is to technically enable the implementation of time-bound transactions between two users or participants. Indeed, when a user receives a HTLC transaction, it has to submit a cryptographic proof within a specific time-frame. Otherwise, the funds will be returned to the original sender.

5.2. Atomic swaps using code-based adaptor signature

Let $(\mathsf{sk}_i, \mathsf{pk}_i)$ be the key pair of user u_i for $i = 1, 2$. Below, we describe how atomic swap could be executed using our code-based adaptor signature.

- We start with user u_1 who randomly generates $(Y, x) \in \mathcal{R}_{\mathbf{H}_{\mathsf{pk}}}$. The user also generates transaction Tx_1 to spend currency c for user u_2.
- User u_1 computes the pre-signature $\tilde{\sigma}_1 = \mathsf{PreSign}(\mathsf{sk}_1, Tx_1, Y)$ and then sends $(\tilde{\sigma}_1, Tx_1, Y)$ to user u_2.

- User u_2 checks $\tilde{\sigma}_1$ using the pre-verification algorithm PreVerif. If the verification is successful, it generates transaction Tx_2 to spend currency c' for user u_1.
- User u_2 computes a pre-signature $\tilde{\sigma}_2 = \mathsf{PreSign}(\mathsf{sk}_2, Tx_2, Y)$ and sends the tuple $(\tilde{\sigma}_2, Tx_2, Y)$ to user u_1. Otherwise, it aborts the transaction.
- After receiving $(\tilde{\sigma}_2, Y, Tx_2)$, user u_1 computes the pre-verification algorithm on $\tilde{\sigma}_2$. If the pre-verification fails, it aborts the transaction. When the pre-verification on $\tilde{\sigma}_2$ is successful, user u_1 runs the adaptor algorithm Adapt to compute the signature $\sigma_2 = \mathsf{Adapt}(\tilde{\sigma}_2, x)$, publishes σ_2 on the blockchain and sends it to user u_2.
- After receiving σ_2, user u_2 computes the extractor algorithm Ext to extract the witness $x' = \mathsf{Ext}(Y, \sigma_2, \tilde{\sigma}_2)$. It then runs the adaptor algorithm Adapt to compute the signature $\sigma_1 = \mathsf{Adapt}(\tilde{\sigma}_1, x')$. To finish, u_2 publishes σ_1 on the blockchain.

The above procedure is depicted in Figure 5.

User $u_1((\mathsf{sk}_1, \mathsf{pk}_1), \mathsf{pk}_2,)$ User $u_2((\mathsf{sk}_2, \mathsf{pk}_2), \mathsf{pk}_1,)$

$(x, Y) \xleftarrow{\$} \mathsf{Gen}()$
Generates transaction Tx_1 to spend
c for user u_2
$\tilde{\sigma}_1 \leftarrow \mathsf{PreSign}(\mathsf{sk}_1, Tx_1, Y)$

$$\xrightarrow{\tilde{\sigma}_1, Y, Tx_1}$$

If Pre-verification of $\tilde{\sigma}_1$ fails, abort.
Generates transaction Tx_2 to spend
c' for user u_1
$\tilde{\sigma}_2 \leftarrow \mathsf{PreSign}(\mathsf{sk}_2, Tx_2, Y)$

$$\xleftarrow{\tilde{\sigma}_2, Tx_2}$$

If Pre-verification of $\tilde{\sigma}$ fails, abort.
$\sigma_2 \leftarrow \mathsf{Adapt}(\tilde{\sigma}_2, x)$
If σ_2 is not a valid signature, abort.
Publish σ_2, on blockchain

$$\xrightarrow{\sigma_2}$$

$x' \leftarrow \mathsf{Ext}(Y, \sigma_2, \tilde{\sigma}_2)$
$\sigma_1 \leftarrow \mathsf{Adapt}(\tilde{\sigma}_1, x')$
If σ_1 is not a valid signature, abort.
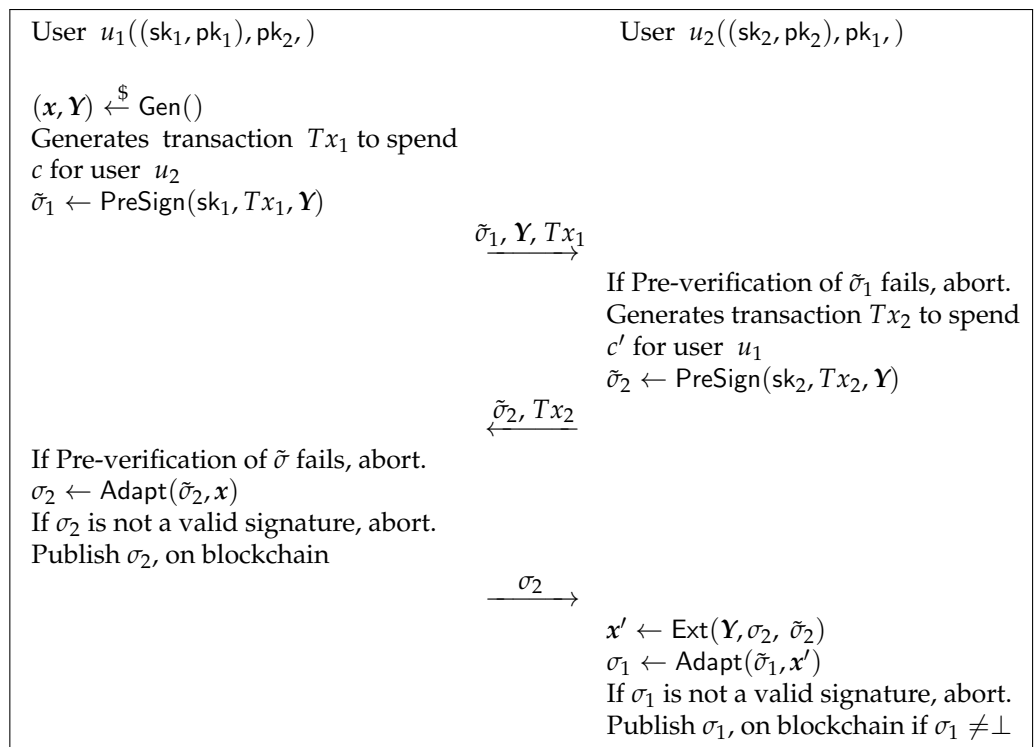Publish σ_1, on blockchain if $\sigma_1 \neq \bot$

Figure 5. Atomic swap using the proposed code-based adaptor signature.

6. Conclusion

In this paper, we have proposed an adaptor signature scheme based on hard problems in coding theory. We use the code-based signature scheme Wave as our underlying signature scheme. In order to equip our scheme with common features and security properties, we have presented some modifications to the Wave signature. We have showed that the proposed adaptor signature scheme is secure under the hardness of the SD and the indistinguishibility of generalized $(U, U + V)$ code problems, both of which are considered quantum-safe. We have also given a set of parameters for adaptor signature uses and compared the proposed scheme with other post-quantum adaptor signature schemes in terms of pre-signature and signature sizes. For parameter values corresponding to Level 1 NIST PQC security, our scheme has a slightly larger signature size, but considerably smaller pre-signature size than those of existing post-quantum adaptor signature schemes available in the literature. With the smaller pre-signature size, our scheme has the potential to reduce the overall communication cost in an atomic swap as there are more exchanges of pre-signatures than signatures.

362 **Institutional Review Board Statement:** Not applicable.

363 **Informed Consent Statement:** Not applicable.

364 **Conflicts of Interest:** The authors declare no conflict of interest.

References

1. Poelstra, A. Scriptless scripts, 2017. Presentation Slides, Available online: https://download.wpsoftware.net/bitcoin/wizardry/mw-slides/2017-05-milan-meetup/slides.pdf (accessed on 14-11-2021).
2. Fournier, L. One-Time Verifiably Encrypted Signatures AKA Adaptor Signatures, 2019. Available online : https://raw.githubusercontent.com/LLFourn/one-time-VES/master/main.pdf (accessed on 28-12-2021).
3. Aumayr, L.; Ersoy, O.; Erwig, A.; Faust, S.; Hostakova, K.; Maffei, M.; Moreno-Sanchez, P.; Riahi, S. Generalized Bitcoin-Compatible Channels. Technical report, Cryptology ePrint Archive: Report 2020/476, 2020.
4. Dryja, T. Discreet log contracts, 2017. Available online: https://dci.mit.edu/research/smart-contracts-discrete-log-contracts (accessed on 28-12-2021).
5. Esgin, M.F.; Ersoy, O.; Erkin, Z. Post-quantum adaptor signatures and payment channel networks. European Symposium on Research in Computer Security. Springer, 2020, pp. 378–397.
6. Erwig, A.; Faust, S.; Hostáková, K.; Maitra, M.; Riahi, S. Two-Party Adaptor Signatures From Identification Schemes. Public Key Cryptography (1), 2021, pp. 451–480.
7. Malavolta, G.; Moreno-Sanchez, P.; Schneidewind, C.; Kate, A.; Maffei, M. Anonymous multi-hop locks for blockchain scalability and interoperability. 26th Annual Network and Distributed System Security Symposium, NDSS 2019, 2019.
8. Moreno-Sanchez, P.; Blue, A.; Le, D.V.; Noether, S.; Goodell, B.; Kate, A. DLSAG: non-interactive refund transactions for interoperable payment channels in monero. International Conference on Financial Cryptography and Data Security. Springer, 2020, pp. 325–345.
9. Tairi, E.; Moreno-Sanchez, P.; Maffei, M. A2L: Anonymous Atomic Locks for Scalability in Payment Channel Hubs. Technical report, Cryptology ePrint Archive, Report 2019/589, 2019.
10. Tairi, E.; Moreno-Sanchez, P.; Maffei, M. Post-Quantum Adaptor Signature for Privacy-Preserving Off-Chain Payments. Technical report, Cryptology ePrint Archive, Report 2020/1345, 2020.
11. Debris-Alazard, T.; Sendrier, N.; Tillich, J.P. Wave: A new code-based signature scheme. Technical report, Cryptology ePrint Archive, Report 2018/996, 2018.
12. Berlekamp, E.; McEliece, R.; Van Tilborg, H. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information Theory* **1978**, *24*, 384–386.
13. Debris-Alazard, T.; Sendrier, N.; Tillich, J.P. The problem with the SURF scheme. *arXiv preprint arXiv:1706.08065* **2017**.
14. Johansson, T.; Jonsson, F. On the complexity of some cryptographic problems based on the general decoding problem. *IEEE Transactions on Information Theory* **2002**, *48*, 2669–2678.
15. Sendrier, N. Decoding one out of many. International Workshop on Post-Quantum Cryptography. Springer, 2011, pp. 51–67.
16. Courtois, N.T.; Finiasz, M.; Sendrier, N. How to achieve a McEliece-based digital signature scheme. International Conference on the Theory and Application of Cryptology and Information Security. Springer, 2001, pp. 157–174.
17. Dallot, L. Towards a concrete security proof of Courtois, Finiasz and Sendrier signature scheme. Western European Workshop on Research in Cryptology. Springer, 2007, pp. 65–77.
18. Faugere, J.C.; Gauthier-Umana, V.; Otmani, A.; Perret, L.; Tillich, J.P. A distinguisher for high-rate McEliece cryptosystems. *IEEE Transactions on Information Theory* **2013**, *59*, 6830–6844.
19. Banegas, G.; Debris-Alazard, T.; Nedeljković, M.; Smith, B. Wavelet: Code-based postquantum signatures with fast verification on microcontrollers. *arXiv preprint arXiv:2110.13488* **2021**.