One-out-of-q OT Combiners

Oriol Farràs and Jordi Ribes-González

Universitat Rovira i Virgili, Tarragona, Spain
{oriol.farras,jordi.ribes}@urv.cat

Abstract. In 1-*out-of-q Oblivious Transfer (OT)* protocols, a sender is able to send one of $q \geq 2$ messages to a receiver, all while being oblivious to which message was actually transferred. Moreover, the receiver only learns one of these messages.
Oblivious Transfer combiners take n instances of OT protocols as input, and produce a single protocol that is secure if sufficiently many of the n original OT implementations are secure.
We present a generalization of an OT combiner protocol that was introduced by Cascudo et al. (TCC'17). We show a general 1-out-of-q OT combiner that is valid for any prime power $q \geq 2$. Our OT combiner is based on secret sharing schemes that are of independent interest.
Our construction achieves the strong notion of *perfect security against active* $(\mathcal{A}, \mathcal{B})$*-adversaries.* For $q \geq n$, we present a single-use, n-server, 1-out-of-q OT combiner that is perfectly secure against active adversaries that corrupt a minority of servers. The amount of bits exchanged during the protocol is $(q^2 + q + 1)n \log q$.

Keywords: Oblivious transfer · OT combiners · Secret sharing schemes

1 Introduction

In this section, we introduce OT protocols and OT combiners, we overview the related literature, and we sketch the aims and results of this article.

1.1 Oblivious Transfer

Oblivious Transfer (OT) protocols were first introduced by Rabin [53] in 1981. Oblivious transfer protocols involve two parties, a *sender* and a *receiver*, which we also respectively name Alice and Bob. The functionality provided by OT consists in allowing the sender to transfer part of its inputs to the receiver, while guaranteeing that the sender is oblivious to which part of its inputs is actually obtained by the receiver. It also guarantees that the receiver is not able learn more information than it is entitled to as per the protocol.

The results of this article are part of the second author's master's thesis.

The first example of an OT functionality, realized in the first OT protocol by Rabin [53], starts with Alice holding a single message. After the execution of the protocol, Bob learns this message with probability 1/2, and Alice is oblivious to whether or not Bob received it. Another flavor of OT is 1-*out-of*-2 *OT* [26], in which the sender holds two messages and where the receiver chooses to receive one of the two messages from the sender. The security guarantees here are that the sender is oblivious to the message that was actually transferred to the receiver, and that the receiver gets information on one of the messages only. The type of OT that we study here is called 1-*out-of*-q *OT*. It is a generalization 1-out-of-2 OT that lets the sender hold $q \geq 2$ messages instead of just two, and allows the receiver to fetch only one of those messages.

The relevance of OT protocols in cryptography lies in their role as a fundamental primitive in many cryptographic constructions. The main functionalities OT has found an application to are secure multi-party computation [57,39], zero-knowledge proofs [39,40,9] and bit commitment schemes [39]. Other related fields are private information retrieval [17] and oblivious linear function evaluation [47,25].

1.2 OT Combiners

The security of OT protocols is necessarily conditional, since perfectly secure OT protocols would yield unconditionally-secure two-party computation by [39], which is impossible to obtain for some functions (see [10,18]). Hence, OT protocols are built by imposing assumptions on security, such as the use of hardware tokens [31], assuming the existence of a noisy channel between both parties [20], or restricting the storage [13] or computational capabilities of the parties. In relation to this last assumption, there exist many computational hardness assumptions one can base OT protocols on, such as the hardness of RSA [53], the Decisional Diffie-Hellman assumption [9,1], the assumptions used in the McEliece encryption scheme [24] and also some worst-case lattice assumptions [51].

The conditional security of OT protocols implies that the security guarantees of OT could be compromised. For example, at some point a hardware token could become corrupted, or a computational assumption could be broken due to cryptanalytic developments. The standard method to mitigate this concern consists in grounding security on various assumptions at once, by simultaneously using several implementations. This motivates the introduction of *OT combiners*.

The notion of *combiner* consists of finding a way to blend various cryptographic implementations into a single one, so that the resulting combination is secure even if some of the original implementations are insecure. Combiners have been previously studied in many areas of cryptography, such as in the familiar context of multi-factor authentication, where many authentication methods are used concurrently, as well as in cascading of block ciphers. Also, previous works have studied combiners of encryption schemes [4,23], of PRGs [34], of hash functions [22] and, of course, of OT protocols.

Using an OT combiner, a set of n candidate implementations of OT can be merged to realize a single OT protocol, in such a way that the final protocol is secure as long as sufficiently many of the initial implementations were secure to begin with. In other words, an OT combiner can be used to instantiate a protocol between a sender Alice and a receiver Bob that realizes OT by internally using n candidate OT implementations. Moreover, the resulting protocol stays secure even if the security of few of the OT candidates is flawed.

1.3 Related Work

The study of OT combiners was initiated by Harnik, Kilian, Naor, Reingold and Rosen [33] in 2005. They define the notion of (n, t)-*OT combiner*, which consists in taking n candidate 1-out-of-2 OT implementations and combining them into a 1-out-of-2 OT protocol that is secure provided at most t of the OT candidates are faulty. They show that, when $t < n/2$, there exist (n, t)-OT combiners that are unconditionally secure against passive (i.e. semi-honest) adversaries. They prove the tightness of this bound and show that such OT combiners cannot exist for $n = 2, t = 1$, and they build an OT combiner for $n = 3, t = 1$. They introduce a second solution for the active (i.e. malicious) adversary model, but this variant has efficiency and security flaws (e.g. see [36, Section 5.4]).

Meier, Przydatek and Wullschleger [46] define the notion of (n, δ)-*uniform OT combiner*. These OT combiners implement the 1-out-of-2 OT functionality, and they are unconditionally secure against passive adversaries that corrupt either Alice and a number t_A of OT candidates, or Bob and t_B OT candidates, for any $t_A + t_B < n$. Their solution requires the roles of the sender and the receiver to be reversed during the protocol execution, and the corresponding combiner makes two calls to each OT candidate.

Later, Przydatek and Wullschleger [52] consider combiners that take a set of n OLFE candidate implementations and produce a 1-out-of-2 OT protocol. Their solution is also unconditionally secure for $t_A + t_B < n$. However, it requires the size of the message space to be greater than the number n of candidate implementations of OLFE to combine. Interestingly, we also consider this restriction in the analysis of our results (see Section 6).

Harnik, Ishai, Kushilevitz and Nielsen [32] present the first *single-use* OT combiner, meaning that one black-box call is made to each of the n OT implementations per protocol execution. They study (n, t_A, t_B)-*OT combiners*, which are secure against passive adversaries that corrupt either Alice and t_A OT candidates, or Bob and t_B OT candidates. A statistically secure (n, t, t)-OT combiner is given for $t = \Omega(n)$, which makes a constant number of calls to each OT candidate. Their solution is set in the 1-out-of-2 scenario. They also provide constant production rate, meaning that the number of secure OT protocols produced is not just one, but a constant fraction $\Theta(n)$ of the number n of OT candidates.

Additionally, [32] gives a computationally secure OT combiner against active adversaries. Subsequently, Ishai, Prabhakaran and Sahai [36] show that this construction can be turned into an (n, t, t)-OT combiner that is statistically secure against active adversaries for $t = \Omega(n)$, while leaving unconditional security as an open problem.

Ishai, Maji, Sahai and Wullschleger [35] present a single-use (n,t,t)-OT combiner in the 1-out-of-2 setting. Their solution is statistically secure against passive adversaries for $t = n/2 - \omega(\log \kappa)$, where κ is the security parameter.

Another variant of combiners for OT is that of *cross-primitive combiners*, studied by Meier and Przydatek in [45]. As in [52], here the combiner implements a different functionality than the candidates. They present a $(2,1)$-*PIR-to-OT combiner*, which takes two Private Information Retrieval (PIR) schemes and produces a 1-out-of-2 OT protocol that is unconditionally secure for the sender, provided one of the two PIR schemes is also secure. This result comes in contrast with the impossibility result of [33]. Their construction only guarantees the privacy of Alice against a honest-but-curious adversary corrupting Bob and one of the two candidates.

Following [35], Cascudo, Damgård, Farràs and Ranellucci [16] achieve single-use 1-out-of-2 OT combiners. They generalize the security notion of Harnik et al. [32] by defining the notion of *perfect security against active* $(\mathcal{A},\mathcal{B})$-*adversaries*, which we also adopt in this article. This definition considers a malicious adversary that can corrupt either Alice and a set $A \in \mathcal{A}$ of OT candidates, or Bob and a set $B \in \mathcal{B}$ of OT candidates, obtaining their inputs and full control of their outputs. The OT combiner in [16] achieves perfect (unconditional, zero-error) security against active adversaries.

In this article we present secret sharing schemes that are of independent interest. Given a function $f : \{0,1\}^n \to \{0,1\}$ be a function, we can define an access structure on $\{1,\ldots,n\} \times \{0,1\}$ whose minimal subsets are $\{(1,x_1),\ldots,(n,x_n)\}$ with $f(x_1,\ldots,x_n) = 1$ and $\{(i,0),(i,1)\}$. Efficient constructions for some of these structures were presented in [7,56]. Recently, Liu and Vaikuntanathan and Wee [44] presented more efficient general constructions for these access structres, and presented a connection between these schemes and Conditional Disclosure of Secrets (CDS) protocols [29] that was later used to construct better general constructions for secret sharing [43,3,2,8]. In this work, we study access structures determined by functions $f : \{0,\ldots,q-1\}^n \to \{0,1\}$, a case that has already been studied in some of these works like [29,2,3].

1.4 Our Work

In this work, we present a 1-*out-of-q OT combiner* that extends previous 1-out-of-2 OT combiners from [16,15] to the 1-out-of-q case, where $q \geq 2$ is an arbitrary prime power. In our setting, the underlying OT candidates and the produced OT protocol take q messages $m_0,\ldots,m_{q-1}$ from Alice and an element $b \in \mathbb{F}_q$ from Bob, and they output the message m_b to Bob.

As in [16,15], we view OT combiners as *server-aided* OT protocols. This means that each of the n OT candidates is modeled as a server that implements the OT functionality, i.e. that receives q messages $m_0,\ldots,m_{q-1} \in \mathbb{F}_q$ from Alice and an element $b \in \mathbb{F}_q$ from Bob, and outputs the message m_b to Bob. In the rest of this article we adopt this convention and refer to each of the n OT candidates as a *server*. In practice, a complete server transaction must be thought of as an OT protocol execution between Alice and Bob. We say an OT combiner is *n-server* if it takes n OT candidates as input.

Consider the case where adversaries can corrupt at most t out of the n OT candidates; i.e. $\mathcal{A} = \mathcal{B} = \binom{n}{t}$. In the single-use case, the OT combiner in [15] can only achieve perfect security against active adversaries for $t = \lfloor 0.11n \rfloor$. We obtain the following result (see Section 6 for more details).

Theorem 1. *Let $n \geq 2$ and $q \geq n$. There exists a single-use, n-server, 1-out-of-q OT combiner that is perfectly secure against active adversaries corrupting at most $t = \lceil n/2 \rceil - 1$ OT candidates. The amount of bits exchanged during the protocol is $(q^2 + q + 1)n \log q$.*

In the process of building our 1-out-of-q OT combiner, we study secret sharing schemes associated to affine spaces. Concretely, let $W \subseteq \mathbb{F}_q^n$ be an affine subspace, and let $f : \mathbb{F}_q^n \to \{0, 1\}$ be a function with $f(x_1, \ldots, x_n) = 1$ if and only if $(x_1, \ldots, x_n) \in W$. We study access structures on the set of nq participants $\{1, \ldots, n\} \times \mathbb{F}_q$ in which a subset $\{(1, v_1), \ldots, (n, v_n)\}$ is authorized if and only if $f(v_1, \ldots, v_n) = 1$. We present ideal $\mathbb{F}_q$-linear secret sharing schemes for access structures with this property. Moreover, from our schemes, it is possible to build n-server CDS protocols for f with domain of secrets $\mathbb{F}_q$ with optimal message size and certain robustness (in the sense of [2]).

This work is organized in six sections. In Section 1, we have given a brief introduction to OT protocols and OT combiners. In Section 2, we lay out the preliminaries on secret sharing schemes, OT and OT combiners needed in the rest of this article. Section 3 presents our 1-out-of-q OT combiner. In Sections 4 and 5 we respectively state the correctness and security definitions and proofs that assess the properties of our construction. At the end of Section 5 we are able to prove Theorem 1. Finally, in Section 6 we conclude the article by commenting on the achieved results and on some future research lines.

2 Preliminaries

In this section, we lay out the background theory needed in the rest of the article. We divide it in five sections. In Section 2.1, we introduce some basic definitions and notation. Section 2.2 presents the OT primitive, along with some examples and applications. Next, in Sections 2.3 and 2.4 we give an account of Secret Sharing, which is an essential primitive to our construction. Finally, in Section 2.5 we introduce OT combiners.

2.1 Notation and Basic Definitions

All through this work, q denotes an arbitrary positive prime power. We identify the set of representatives of the integer residue classes modulo q with the set of non-negative integers smaller than q. Hence, by abuse of notation, we denote $\mathbb{F}_q = \{0, \ldots, q - 1\}$. Given an integer $n \geq 2$, we denote by $\mathcal{P}_n$ the set of positive integers up to n, i.e. $\mathcal{P}_n := \{1, \ldots, n\}$. We define $\mathcal{P}_{n,q} := \mathcal{P}_n \times \mathbb{F}_q = \{(i,j) \,:\, i \in \mathcal{P}_n,\, j \in \mathbb{F}_q\}$. The power set of a set P is $2^P := \{A \,:\, A \subseteq P\}$.

In this work we deal with two-party protocols, and the aim of such protocols is to compute a certain functionality. The notion of functionality is formalized in the next definition.

Definition 1 ([42]). *A* functionality $\mathcal{F}$ *is a possibly random process* $\mathcal{F} : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^*$ *that takes a pair of inputs* $x, y \in \{0,1\}^*$ *and outputs a random variable* $(\mathcal{F}_1(x,y), \mathcal{F}_2(x,y))$.

We say a protocol between two parties Alice and Bob *implements a functionality* $\mathcal{F}$ when, assuming Alice and Bob behave honestly and have input x and y respectively, at the end of the protocol Alice obtains $\mathcal{F}_1(x,y)$ and Bob obtains $\mathcal{F}_2(x,y)$.

2.2 Oblivious Transfer

The main functionality studied in this work, called the 1-*out-of-*q *OT functionality*, was first presented by Crépeau, Brassard and Robert [19] in 1986, and it generalizes that of 1-out-of-2 OT by allowing Alice to hold multiple messages. In the 1-out-of-q OT functionality, the sender Alice is assumed to hold q messages $m_0, \ldots, m_{q-1}$, and the receiver Bob chooses a message index $b \in \mathbb{F}_q$. At the end of a protocol implementing this functionality, Bob receives m_b and Alice receives nothing. That is, in the notation of Definition 1, the functionality $\mathcal{F}(x,y) = (\mathcal{F}_1(x,y), \mathcal{F}_2(x,y))$ implemented by 1-out-of-q OT protocols is described by

$$\begin{array}{ll} x = (m_0, \ldots, m_{q-1}), & \mathcal{F}_1(x,y) = \perp, \\ y = b, & \mathcal{F}_2(x,y) = m_y. \end{array}$$

where $\perp$ stands for the empty bit string. This functionality is illustrated in Fig. 1.

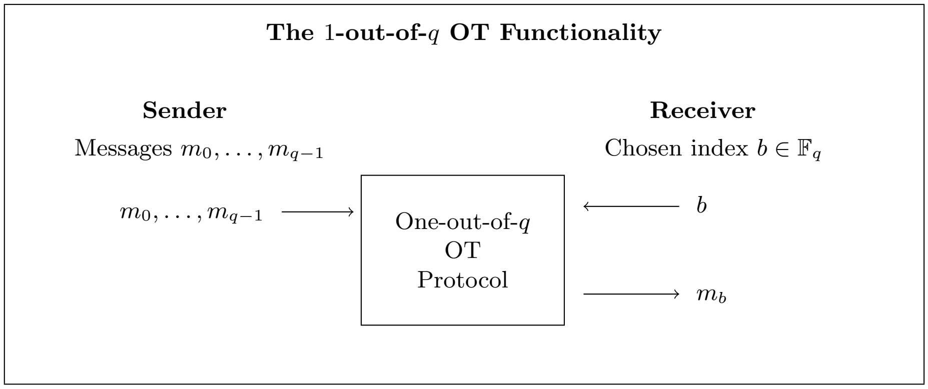


Fig. 1. One-out-of-q OT.

Given a 1-out-of-q OT protocol, it is possible to build a t-out-q OT protocol by invoking t runs of the original protocol [55]. In a t-out-q OT protocol, Bob recovers t messages out of the q that Alice holds. It is also possible to build 1-out-of-q OT protocols for bit messages by invoking a 1-out-of-2 OT protocol $q - 1$ times [19], or even just $\log q$ times [48].

The 1-out-of-q extension of OT enables applications such as private set intersection [41,49], private information retrieval [48] and multi-party computation [30] (where 1-out-of-4 OT is necessary to securely evaluate arithmetic multiplication gates).

2.3 Secret Sharing Schemes

Secret sharing schemes, introduced by Shamir [54] and Blakley [11], are cryptographic primitives used to protect a *secret value* by distributing it into *shares*. In the typical scenario, a user called the *dealer* holds the secret value and generates a set of shares. Then, it sends each share privately to a different *participant*. We next state a formal definition of secret sharing scheme, taken from [15]. See [6,50] for an introduction to secret sharing.

Definition 2. *Let $P = \{1, \ldots, n\}$ be the set of participants. A* Secret Sharing scheme Σ on P *consists of the following two algorithms*

$(x_1, \ldots, x_n) \leftarrow \texttt{Share}_\Sigma(s, \mathbf{r})$**:** *Probabilistic algorithm that takes as input a* secret *s, belonging to a finite set E_0, and some randomness $\mathbf{r}$. It returns an array of values $(x_1, \ldots, x_n)$, where each x_i belongs to some finite set E_i. This array is called a* sharing *of s, and each of its elements is a* share *of s.*

$s \leftarrow \texttt{Reconstruct}_\Sigma((i, x_i)_{i \in A})$**:** *Algorithm that takes a set of pairs $(i, x_i)_{i \in A}$ as input for some $A \subseteq P$, where $x_i \in E_i$. It returns either a secret s, or $\perp$.*

Following the notation of [16,15], given a secret s and randomness $\mathbf{r}$, we denote a sharing of the secret s by $[s, \mathbf{r}]_\Sigma = \texttt{Share}_\Sigma(s, \mathbf{r})$. Whenever we can safely drop the randomness $\mathbf{r}$, we denote this sharing by $[s]_\Sigma$. The indexes i of shares x_i in the input to $\texttt{Reconstruct}_\Sigma$ are omitted when implicitly clear.

We say a subset $A \subseteq P$ is *authorized* for Σ if, for every secret s, provided the shares $(x_i)_{i \in A}$ are part of a sharing of s, the function $\texttt{Reconstruct}((i, x_i)_{i \in A})$ recovers s with overwhelming probability. That is, if, for every secret s,

$$\mathsf{Pr}[\texttt{Reconstruct}_\Sigma(\texttt{Share}_\Sigma(s, \mathbf{r})) = s] = 1.$$

Similarly, we say that $A \subseteq P$ is *forbidden* for Σ when the shares $(x_i)_{i \in A}$ of participants in A do not reveal any information on the secret value s. That is, if, for every $s, s' \in E_0$,

$$\mathsf{Pr}[(\texttt{Share}_\Sigma(s, \mathbf{r}))_A = (x_i)_{i \in A}] = \mathsf{Pr}[(\texttt{Share}_\Sigma(s', \mathbf{r}))_A = (x_i)_{i \in A}].$$

We define the *adversary* (resp. *access*) *structure* of Σ as the collection of all forbidden (resp. authorized) subsets $A \subseteq P$ for Σ. We say that Σ is *perfect* if every subset $A \subseteq P$ is either authorized or forbidden for Σ.

Given a secret sharing scheme Σ on P, the *information ratio* $\sigma(\Sigma)$ of Σ is a quantity that measures the efficiency of secret sharing schemes. It is defined as the ratio of the maximum length in bits of the shares to the length of the secret

$$\sigma(\Sigma) = \frac{\max_{1\leq i\leq n} \log |E_i|}{\log |E_0|}.$$

The schemes with information ratio 1 are called *ideal*.

Given an access structure Γ, we define the *minimal access structure of* Γ by $\min \Gamma = \{A \in \Gamma : B \not\subset A \text{ for all } B \in \Gamma\}$. Similarly, given an adversary structure $\mathcal{A}$, we define the *maximal adversary structure of* $\mathcal{A}$ by $\max \mathcal{A} = \{A \in \mathcal{A} : A \not\subset B \text{ for all } B \in \mathcal{A}\}$.

If $\mathcal{A}, \mathcal{B} \subseteq 2^P$ are two adversary structures, we say that they are $\mathcal{R}_2$ when $A \cup B \neq P$ for every $A \in \mathcal{A}, B \in \mathcal{B}$. We need the following lemma.

Lemma 1 ([15]). *Let $(\mathcal{A}, \mathcal{B})$ be an $\mathcal{R}_2$ pair of adversary structures, and Σ a perfect secret sharing scheme with $\mathcal{A}$ as its adversary structure. Then, for every $B \in \mathcal{B}$, its complement $\overline{B}$ is authorized in Σ.*

2.4 Linear Secret Sharing Schemes

Linear Secret Sharing schemes (LSSS) are a type of secret sharing schemes that is key to building our 1-out-of-q OT construction. We now define LSSS, and we restate some of the previous properties. We also provide a result needed to prove the security of our construction.

Definition 3. *Let $\mathbb{K}$ be a finite field, $P = \{1, \ldots, n\}$, and let Σ be a secret sharing scheme, where secrets s take values in a finite set E_0 and sharings $(x_1, \ldots, x_n) \in E_1 \times \cdots \times E_n$.*

Then Σ is called $\mathbb{K}$-linear *(or a* $\mathbb{K}$-Linear Secret Sharing scheme*, written* $\mathbb{K}$-*LSSS) if the following conditions hold*

- *$\Omega, E_0, \ldots, E_n$ are vector spaces of finite dimension over $\mathbb{K}$,*
- *the randomness $\mathbf{r}$ is chosen uniformly over Ω, and*
- *$\mathtt{Share}_\Sigma$ is defined as a $\mathbb{K}$-linear surjective map*

$$\mathtt{Share}_\Sigma : E_0 \times \Omega \to E_1 \times \cdots \times E_n.$$

In this work we only consider $\mathbb{F}_q$-linear secret sharing schemes where $\dim E_0 = 1$. That is, we may assume that $E_0 = \mathbb{F}_q$ and that, for each $i \in P$, the i-th share space is $E_i = \mathbb{F}_q^{\ell_i}$ for some positive integer ℓ_i. These schemes are perfect.

The information ratio of a LSSS Σ with $\dim E_0 = 1$ is $\sigma(\Sigma) = \max_{i\in P} \dim E_i$. Every adversary structure admits an $\mathbb{F}_q$-LSSS for every q [37]. However, almost all access structures require $\mathbb{F}_q$-LSSS with information ratio at least $2^{n/3-o(n)}$ for every q [5]. The characterization of adversary structures admitting $\mathbb{F}_q$-LSSS with small share sizes is an open problem in secret sharing.

Given a secret value $x_0 \in \mathbb{F}_q$, we have that $[x_0]_\Sigma \in \mathbb{F}_q^{\ell_1} \times \cdots \times \mathbb{F}_q^{\ell_n}$. In this case, if we denote by V the set of all possible shares $[0]_\Sigma$ of $0 \in \mathbb{F}_q$, we have that

$V = \{\mathtt{Share}_\Sigma(0, \mathbf{r}) \ : \ \mathbf{r} \in \Omega\}$ is a vector subspace of $\mathbb{F}_q^{\ell_1} \times \cdots \times \mathbb{F}_q^{\ell_n}$. Similarly, if we denote by W_b the set of all possible shares $[b]_\Sigma$ of a secret value $b \in \mathbb{F}_q$, we have that $W_b = [b]_\Sigma + V$ is an affine subspace of $\mathbb{F}_q^{\ell_1} \times \cdots \times \mathbb{F}_q^{\ell_n}$, where $[b]_\Sigma$ denotes some share of b using Σ. We make explicit use of the affine subspaces V and W_b in our construction.

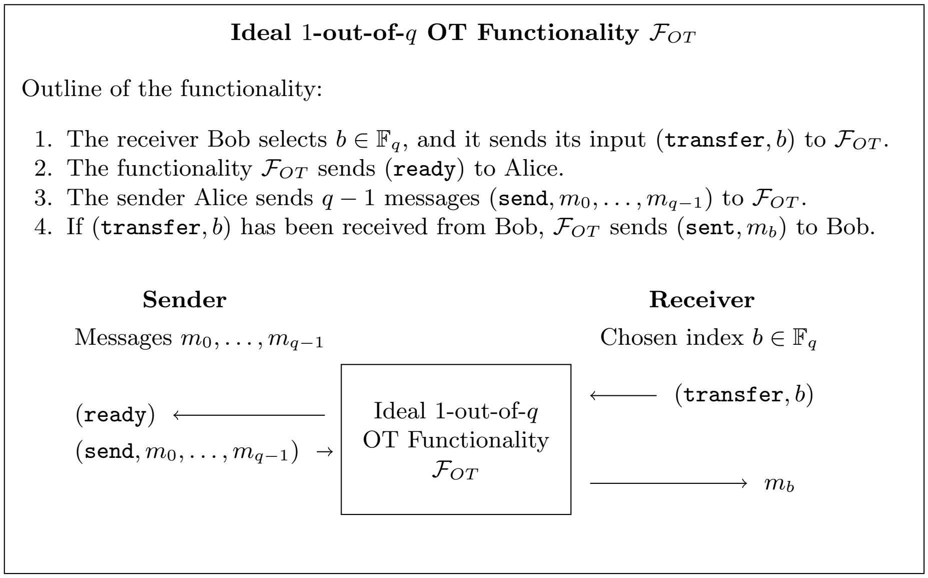
The following lemma follows from the definition of access structure above.

Lemma 2. *Let Σ be an $\mathbb{F}_q$-linear secret sharing scheme with* $\dim E_0 = 1$. *A subset $A \subseteq P$ is forbidden for Σ if and only if there exists a vector $\mathbf{r} \in \Omega$ for which $\mathtt{Share}_\Sigma(1, \mathbf{r}) = (x_1, \dots, x_n)$ satisfies $x_i = \mathbf{0}$ for every $i \in A$.*

2.5 OT combiners

Here we lay out the fundamental theory of OT combiners. We define them, we name some of their properties, and we fix notation.

Before proceeding further, and as in [15], we need to introduce the *ideal* 1*-out-of-q OT functionality* $\mathcal{F}_{OT}$. We make use of the ideal functionality $\mathcal{F}_{OT}$ in our correctness and security definitions. It consists of an ideal version of a 1-out-of-q OT protocol that implements the functionality correctly and that does not allow any kind of corruption. Hence, $\mathcal{F}_{OT}$ is an abstraction of an ideal OT protocol, and not a functionality in the sense of Definition 1. Without loss of generality, in this work all 1-out-of-q OT protocols that are considered secure are assumed to follow the footprint of $\mathcal{F}_{OT}$. Figure 2 depicts the $\mathcal{F}_{OT}$ ideal functionality.



Ideal 1-out-of-q OT Functionality $\mathcal{F}_{OT}$

Outline of the functionality:

1. The receiver Bob selects $b \in \mathbb{F}_q$, and it sends its input $(\mathtt{transfer}, b)$ to $\mathcal{F}_{OT}$.
2. The functionality $\mathcal{F}_{OT}$ sends $(\mathtt{ready})$ to Alice.
3. The sender Alice sends $q-1$ messages $(\mathtt{send}, m_0, \dots, m_{q-1})$ to $\mathcal{F}_{OT}$.
4. If $(\mathtt{transfer}, b)$ has been received from Bob, $\mathcal{F}_{OT}$ sends $(\mathtt{sent}, m_b)$ to Bob.

Sender
Messages $m_0, \dots, m_{q-1}$
Receiver
Chosen index $b \in \mathbb{F}_q$
$(\mathtt{ready})$
$(\mathtt{send}, m_0, \dots, m_{q-1})$
Ideal 1-out-of-q OT Functionality $\mathcal{F}_{OT}$
$(\mathtt{transfer}, b)$
m_b

Fig. 2. The ideal 1-out-of-q Oblivious Transfer functionality.

Next, we formally define OT combiners, following the notation of [16,15].

Definition 4. *Let $S_1, \dots, S_n$ be candidate OT implementations. An* OT combiner *is an efficient two-party protocol $\pi = \pi(S_1, \dots, S_n)$, with access to the candidates $S_1, \dots, S_n$, that implements the OT functionality.*

We say that an OT combiner is 1*-out-of-q* if it implements the 1-out-of-q OT functionality. An OT combiner is *black-box* if, during the protocol, the candidate OT implementations are used in a black-box way, i.e. ignoring their internal workings and making oracle calls as in the ideal OT functionality. Under the black-box assumption, as in [16,15], we refer to each of the OT candidate implementations as *servers* (as noted at the end of Section 1.4). An OT combiner is *single-use* if each server is used only once during the execution of the protocol.

From this point onward we assume OT combiners to be 1-out-of-q, n-server, single-use and black-box. Under this assumption, we can formalize the notion of OT combiner according to the following definition.

Definition 5. *We define a* 1*-out-of-q, n-server, single-use, black-box* OT combiner $\pi = \pi(S_1, \dots, S_n)$ *by means of the next three polynomial-time algorithms:*

$(b_1, \dots, b_n) \leftarrow \pi.\mathsf{Choose}(b)$**:** *Probabilistic algorithm run by the receiver Bob and taking as input a*

$$\texttt{Reconstruct}_{\mathcal{S}_k}\left((m_k^{(i,j)})_{(i,j)\in A}\right) = \sum_{i=1}^{n} m_k^{(i,b_i)}.$$

message index $b \in \mathbb{F}_q$. It returns an n-tuple $(b_1, \dots, b_n)$, where each $b_i \in \mathbb{F}_q$ is to be sent to server S_i.

$(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}} \leftarrow \pi.\mathsf{Send}(m_0, \dots, m_{q-1})$**:** *Probabilistic algorithm run by the sender Alice, taking as input q chosen messages $m_0, \dots, m_{q-1}$. It returns a qn-tuple $(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}}$, where each tuple $(u_i^0, \dots, u_i^{q-1})$ is to be sent to server S_i.*

$m \leftarrow \pi.\mathsf{Reconstruct}(b, (v_1, \dots, v_n))$**:** *Algorithm run by the receiver Bob, that takes as input the chosen message index $b \in \mathbb{F}_q$ and n elements $v_1, \dots, v_n$, where each v_i is received from server S_i. It returns a message m.*

Given an OT combiner $\pi = (\pi.\mathsf{Choose}, \pi.\mathsf{Send}, \pi.\mathsf{Reconstruct})$ and given n servers $S_1, \dots, S_n$ implementing the 1-out-of-q OT functionality, we can regard π as a protocol between a sender Alice and a receiver Bob. In this case, the resulting OT protocol $\pi(S_1, \dots, S_n)$ develops sequentially in five phases:

Choice Phase: The receiver Bob chooses a message index $b \in \mathbb{F}_q$.
Bob generates the tuple $(b_1, \dots, b_n) \leftarrow \pi.\mathsf{Choose}(b)$ where $b_i \in \mathbb{F}_q$.
Bob sends $(\texttt{transfer}, b_i)$ to server S_i for $i = 1, \dots, n$.

Ready Phase: On receiving b_i from Bob, the server S_i sends $(\texttt{ready})$ to Alice.

Sending Phase: The sender Alice chooses q messages $m_0, \dots, m_{q-1}$.
Alice generates the corresponding tuple

$$(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}} \leftarrow \pi.\mathsf{Send}(m_0, \dots, m_{q-1}).$$

After Alice has received $(\texttt{ready})$ from every server, she sends the generated shares $(\texttt{send}, u_i^0, \dots, u_i^{q-1})$ to S_i for $i = 1, \dots, n$.

Transfer Phase: The server S_i sends $(\texttt{sent}, u_i^{b_i})$ to Bob.
Output Phase: Bob reconstructs the message m_b from the shares $u_1^{b_1}, \ldots, u_n^{b_n}$ he received by executing $\pi.\mathsf{Reconstruct}(b, (u_1^{b_1}, \ldots, u_n^{b_n}))$.

The diagram of the protocol for the case $q = 4$ and $n = 3$ is presented in Figure 3.

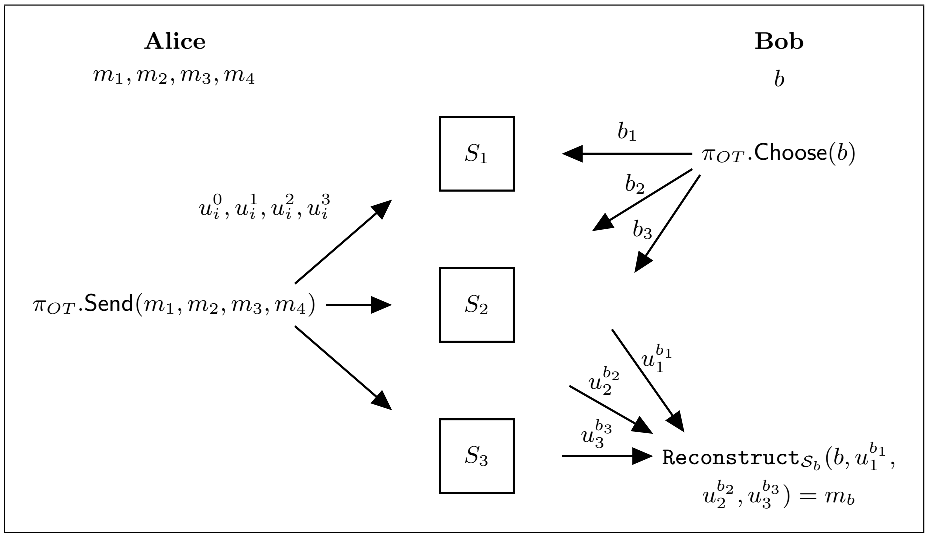

Alice
m_1, m_2, m_3, m_4
Bob
b
S_1
b_1
$\pi_{OT}.\mathsf{Choose}(b)$
b_2
$u_i^0, u_i^1, u_i^2, u_i^3$
b_3
$\pi_{OT}.\mathsf{Send}(m_1, m_2, m_3, m_4)$
S_2
$u_1^{b_1}$
$u_2^{b_2}$
$u_3^{b_3}$
S_3
$\texttt{Reconstruct}_{\mathcal{S}_b}(b, u_1^{b_1},$
$u_2^{b_2}, u_3^{b_3}) = m_b$

Fig. 3. Diagram of a 1-out-of-4 OT combiner for $n = 3$.

3 One-out-of-q OT Combiners

This section introduces our 1-out-of-q OT combiner, which can be seen as an extension of the OT combiner in [15] to the 1-out-of-q scenario. Here we introduce our construction for the particular case where the adversary structure $\mathcal{A}$ of the security definition admits an ideal $\mathbb{F}_q$-linear secret sharing scheme. In Appendix C we describe our construction in full generality, achieving an OT combiner with perfect security against active $(\mathcal{A}, \mathcal{B})$-adversaries, where $(\mathcal{A}, \mathcal{B})$ is an arbitrary $\mathcal{R}_2$ pair of adversary structures.

Later in Section 5, our 1-out-of-q OT protocol is proven secure against any $(\mathcal{A}, \mathcal{B})$-adversary (see Definition 12), where $\mathcal{A}, \mathcal{B} \subseteq 2^{\mathcal{P}_n}$ is any pair of $\mathcal{R}_2$ adversary structures such that $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS. Throughout this section, we assume that the pair $(\mathcal{A}, \mathcal{B})$ of adversary structures is fixed, and that $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS Σ. The efficiency of our OT combiner is affected by the size of the shares of Σ, and it is best in this ideal case. We note that the characterization of adversary structures that admit $\mathbb{F}_q$-LSSS with small share sizes is an open problem in secret sharing. See Section 2.4 or [50] for more details.

This section is organized as follows. First, in Section 3.1 we develop the notion of *OT-Compatibility*, necessary to extend the previous scheme of [15] to suit our purposes. The OT-compatible secret sharing scheme we make use of is described in Section 3.2. Then, in Section 3.3 we explicitly describe our 1-out-of-q OT combiner for the particular case where $\mathcal{A}$ admits a perfect ideal $\mathbb{F}_q$-LSSS.

3.1 Definition of OT-Compatibility

Let $\mathcal{A} \subseteq 2^{\mathcal{P}_n}$ be an adversary structure on the set $\mathcal{P}_n = \{1, \ldots, n\}$ of n participants, and let Σ be an ideal $\mathbb{F}_q$-LSSS for $\mathcal{P}_n$ with adversary structure $\mathcal{A}$. As in [15], the scheme Σ is used by the receiver Bob to request the message with the selected index $b \in \mathbb{F}_q$, simply by generating a sharing $[b]_\Sigma = (b_1, \ldots, b_n)$ of b under Σ and sending each share $b_i \in \mathbb{F}_q$ to the corresponding server S_i.

Denote by $V \subseteq \mathbb{F}_q^n$ the vector space consisting of all the sharings of 0 under the scheme Σ. Given any $b \in \mathbb{F}_q$, let $W_b \subseteq \mathbb{F}_q^n$ be the affine subspace of sharings of b for Σ. Note that, by this definition, $V = W_0$. Since Σ is an $\mathbb{F}_q$-LSSS, we can express $W_b = \mathbf{b} + V$, where $\mathbf{b} = [b]_\Sigma$ is a sharing of b for Σ. We can also express $\mathbb{F}_q^n$ as the disjoint union $\mathbb{F}_q^n = W_0 \cup \cdots \cup W_{q-1}$.

In order for Alice to send the messages $m_0, \ldots, m_{q-1}$ to each server, our construction follows the blueprint of [15] and makes use of secret sharing schemes related to affine subspaces $W \subseteq \mathbb{F}_q^n$. All such schemes proposed here are defined on the set of nq participants $\mathcal{P}_{n,q} = \mathcal{P}_n \times \mathbb{F}_q$. We also consider the partition

$$\mathcal{P}_{n,q} = P_1 \cup \ldots \cup P_n,$$

where $P_i = \{(i,0), (i,1) \ldots, (i, q-1)\}$ for $i = 1, \ldots, n$.

We associate an access structure $\Gamma_W \subseteq 2^{\mathcal{P}_{n,q}}$ to each $W \subseteq \mathbb{F}_q^n$ as follows.

Definition 6. *Let $W \subseteq \mathbb{F}_q^n$. We define Γ_W as the access structure on $\mathcal{P}_{n,q}$ determined by the minimal access structure*

$$\min \Gamma_W = \{\{(1,b_1), (2,b_2), \ldots, (n,b_n)\} \, : \, \mathbf{b} = (b_1, b_2, \ldots, b_n) \in W\} .$$

In the 1-out-of-q scenario, Alice holds q messages $m_0, \ldots, m_{q-1}$. To generalize the construction in [15] to this scenario, we would need to instantiate q $\mathbb{F}_q$-LSSS $\mathcal{S}_0, \ldots, \mathcal{S}_{q-1}$ on the set of participants $\mathcal{P}_{n,q} = \mathcal{P}_n \times \mathbb{F}_q$, so that $\mathcal{S}_k$ has access structure Γ_{W_k} for each $k \in \mathbb{F}_q$. Then, Alice would generate a sharing

$$[m_k]_{\mathcal{S}_k} = (m_k^{(i,j)})_{(i,j) \in \mathcal{P}_{n,q}}$$

of each message m_k, and she would send q of these shares, $m_k^{(i,0)}, \ldots, m_k^{(i,q-1)}$, to each OT server S_i for each message m_k. Since this requires exactly q shares per server, we would need the $\mathbb{F}_q$-LSSS $\mathcal{S}_k$ for Γ_{W_k} to be ideal for each $k \in \mathbb{F}_q$.

In [15] Cascudo et al. prove that, if $W \subseteq \mathbb{F}_2^n$ is an affine subspace, then the access structure Γ_W described above always admits an ideal $\mathbb{F}_2$-LSSS. However, in general, given an affine subspace $W \subseteq \mathbb{F}_q^n$, ideal $\mathbb{F}_q$-LSSS for the access structure Γ_W are not expected to exist. While $\mathbb{F}_q$-LSSS are guaranteed to exist for any such access structure thanks to [37], the ideality requirement may prove harder to obtain. Hence, we can not just take the course of action described above.

The main idea of this work is that, instead of aiming for $\mathbb{F}_q$-LSSS with access structures of the form Γ_W, it is possible to relax the conditions on the access structure and still be able to construct ideal schemes that fit our security needs. We accordingly propose the notion of *W-OT-compatibility*.

Definition 7. *Let $W \subseteq \mathbb{F}_q^n$. Let $\Delta \subseteq 2^{\mathcal{P}_{n,q}}$ be the family of subsets defined by*

$$\Delta = \{A_1 \cup \ldots \cup A_n \,:\, A_i \subseteq P_i \text{ and } |A_i| = 0, 1 \text{ or } q \text{ for } i = 1, \ldots, n\}.$$

We say that an access structure $\Gamma \subseteq 2^{\mathcal{P}_{n,q}}$ is W-OT-compatible *if $\Gamma \cap \Delta = \Gamma_W \cap \Delta$. Similarly, we say that a secret sharing scheme is* W-OT-compatible *if its access structure is* W-OT-compatible.

The motivation behind this definition is the following: the $\mathbb{F}_q$-LSSS to be used by Alice that we design are built so that an adversary controlling Bob, and possibly some servers, can learn from each server S_i either

- no shares, e.g. in the case where an active adversary corrupts Alice and S_i,
- one share, e.g. in the case that the server S_i is not corrupted, or
- all q shares sent to S_i, in the case that an adversary corrupts Bob and S_i.

In particular, the obtained $\mathbb{F}_q$-LSSS $\mathcal{S}_k$ satisfy the condition that the knowledge of any two distinct shares sent to server S_i leads to the knowledge of all q of them. Under this assumption, the shares that an adversary controlling Bob is able to see in any execution of the OT combiner are always determined by some subset of Δ. Therefore, even if the obtained $\mathbb{F}_q$-LSSS has an access structure Γ other than Γ_W, it serves our security purposes as long as Γ coincides with Γ_W when restricting it to Δ. That is, as long as Γ is W-OT-compatible.

We next state some properties of W-OT-compatible access structures.

Remark 1. If an access structure $\Gamma \subseteq 2^{\mathcal{P}_{n,q}}$ is W-OT-compatible, then

- $\min \Gamma_W \subseteq \min \Gamma$. So $\{(1, b_1), \ldots, (n, b_n)\} \in \Gamma$ for every $(b_1, \ldots, b_n) \in W$.
- $\{(1, v_1), \ldots, (n, v_n)\} \notin \Gamma$ for every $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{F}_q^n \setminus W$.
- If $A \in \mathcal{P}_{n,q}$ has size $|A| < n$, then $A \notin \Gamma$.
- If $A \in \Gamma$ has size $|A| = n$, then $A \in \min \Gamma_W$ and $A \in \min \Gamma$.
- $\mathcal{P}_{n,q} \setminus P_i \notin \Gamma$ for $i = 1, \ldots, n$.

In Appendix B we give some examples of W-OT-compatible access structures.

3.2 Our W-OT-Compatible Linear Secret Sharing Scheme

Given $k \in \mathbb{F}_q$, we instantiate the $\mathbb{F}_q$-LSSS $\mathcal{S}_k$ associated to the affine subspace W_k in Figure 4. The scheme $\mathcal{S}_k$ is used by Alice to generate the input to each OT server for a single message m_k. It is defined on the set of nq participants $\mathcal{P}_{n,q}$ and it is $\mathbb{F}_q$-linear and ideal.

The Secret Sharing Scheme $\mathcal{S}_k$

To share a message $m \in \mathbb{F}_q$, first

- let $\mathbf{k} = (k_1, \ldots, k_n) \in \mathbb{F}_q^n$ be a sharing of k using Σ
- sample $r_1, \ldots, r_{n-1} \in \mathbb{F}_q$ uniformly at random, and let $r_n = m - \sum_{i=1}^{n-1} r_i$
- sample $\mathbf{h} = (h_1, \ldots, h_n)$ uniformly at random from $V^\perp$

For every $i \in \mathcal{P}_n$ and for every $j \in \mathbb{F}_q$, define the (i,j)-th share as

$$m^{(i,j)} = r_i + (k_i - j)h_i.$$

Fig. 4. The $\mathbb{F}_q$-LSSS $\mathcal{S}_k$ related to the affine subspace $W_k \subseteq \mathbb{F}_q^n$.

Assuming $A \subseteq \mathcal{P}_{n,q}$ contains a set $A' \in \min \Gamma_{W_k}$, which are of the form $A' = \{(1, b_1), \ldots, (n, b_n)\}$ where $\mathbf{b} = (b_1, \ldots, b_n) \in W_k$, we can define the function $\texttt{Reconstruct}_{\mathcal{S}_k}$ on the shares $(m_k^{(i,j)})_{(i,j)\in A}$ of the message m_k as

$$\texttt{Reconstruct}_{\mathcal{S}_k}\left((m_k^{(i,j)})_{(i,j)\in A}\right) = \sum_{i=1}^{n} m_k^{(i,b_i)}.$$

To see that this function effectively reconstructs m_k, note that

$$\sum_{i=1}^{n} m_k^{(i,b_i)} = \sum_{i=1}^{n} (r_i + (k_i - b_j)h_i) = \sum_{i=1}^{n} r_i + \langle \mathbf{k} - \mathbf{b}, \mathbf{h} \rangle = m_k$$

since we know that $\sum_{i=1}^{n} r_i = m$, that $\mathbf{k}, \mathbf{b} \in W_k$ (so $\mathbf{k} - \mathbf{b} \in V$) and $\mathbf{h} \in V^\perp$.

The following theorem states that the $\mathbb{F}_q$-LSSS $\mathcal{S}_k$ satisfies the properties required for our purposes. Its proof is in Appendix A

Theorem 2. *For every $k \in \mathbb{F}_q$, the secret sharing scheme $\mathcal{S}_k$ defined in Figure 4 is $\mathbb{F}_q$-linear, perfect, ideal and W_k-OT-compatible.*

3.3 Our One-out-of-q OT Combiner in the Ideal Case

Let Σ be an ideal $\mathbb{F}_q$-LSSS for n participants, with adversary structure $\mathcal{A} \subseteq 2^{\mathcal{P}_n}$. The shares generated with this scheme are used by Bob to query each server. Also, denote by $\mathcal{S}_k$ the ideal $\mathbb{F}_q$-LSSS defined previously in Figure 4 for $k \in \mathbb{F}_q$. Remind that the scheme $\mathcal{S}_k$ is attached to the affine subspace $W_k \subseteq \mathbb{F}_q^n$ determined by $W_k = \mathbf{k} + V$, where $\mathbf{k}$ is a sharing of k for the scheme Σ and $V \subseteq \mathbb{F}_q^n$ is the vector space consisting of all the sharings of 0 for the scheme Σ.

We are now in position to describe our 1-out-of-q OT combiner in the case that the adversary structure $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS. The protocol runs between a sender Alice and a receiver Bob, who communicate through a set of n servers $S_1, \ldots, S_n$ that implement the ideal 1-out-of-q OT functionality $\mathcal{F}_{OT}$ (described in Figure 2). The proposed construction is defined in Figure 5 below.

Our 1-out-of-q OT Combiner π_{OT}

π_{OT}.$\mathsf{Choose}(b)$: Given $b \in \mathbb{F}_q$, compute a sharing $[b]_\Sigma = (b_1, \ldots, b_n)$ of b using Σ.
Note that each $b_i \in \mathbb{F}_q$ because Σ is ideal.
Output $(b_1, \ldots, b_n)$.

π_{OT}.$\mathsf{Send}(m_0, \ldots, m_{q-1})$: For each message m_k, independently compute a sharing

$$[m_k]_{\mathcal{S}_k} = (m_k^{(i,j)})_{(i,j)\in\mathcal{P}_{n,q}}.$$

Then, for each $(i,j) \in \mathcal{P}_{n,q}$, compute the values

$$u_i^j := m_0^{(i,j)} || m_1^{(i,j)} || \cdots || m_{q-1}^{(i,j)}.$$

Output $(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}}$.

π_{OT}.$\mathsf{Reconstruct}(b, (v_1, \ldots, v_n))$: Parse each v_i as

$$v_i = n_0^{(i)} || n_1^{(i)} || \cdots || n_{q-1}^{(i)},$$

where $n_k^{(i)} \in \mathbb{F}_q$ for each $i \in \mathcal{P}_n$.
If $b = k$, retrieve m_b by evaluating

$$\texttt{Reconstruct}_{\mathcal{S}_k}((n_k^{(i)})_{i\in\mathcal{P}_n}).$$

If the reconstruction fails at any step, output $\mathbf{0}$.
Otherwise, output the reconstructed message m_b.

Fig. 5. Our 1-out-of-q OT combiner π_{OT} in the case where the access structure $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS Σ.

Remark 2. In the Choice phase, Bob sends a total of $n \log q$ bits to servers. In the Sending phase, Alice sends a total of $q^2 n \log q$ bits to the servers. In the Transfer phase, servers send a total of $n \log q$ bits to Bob. Hence, the communication complexity is $(q^2 + q + 1)n \log q$.

Remark 3. For $q \geq n$, there exists an ideal threshold secret sharing scheme Σ with adversary structure $\mathcal{A} = \{A \subseteq \mathcal{P}_n \ : \ |A| < n/2\}$.

4 Correctness of our OT Combiner

This section deals with the correctness of our 1-out-of-q OT combiner. In Section 4.1, we present the used correctness definitions. Then, in Section 4.2 we prove the correctness of our construction.

4.1 Correctness Definitions

The correctness property of OT combiners refers to the fact that, in the eyes of the receiver Bob, the produced protocol should always implement the OT functionality correctly. To define correctness, we need to consider two scenarios:

one where the sender Alice follows the protocol honestly, and one where she may act maliciously.

In the first scenario all participants behave honestly. Here, we must ensure that, assuming all servers correctly implement the OT functionality and that parties follow the protocol honestly, the protocol produced by the combiner implements the OT functionality correctly. Hence, we have to show that the message retrieved by Bob in the execution of the OT combiner is exactly the one that he should receive as per the OT functionality.

This first approach to correctness is expressed by the zero-error property, which we formalize in the following definition.

Definition 8. *An OT combiner π is* zero-error *if for every message index $b \in \mathbb{F}_q$ and for any q messages $m_0, \ldots, m_{q-1}$ we have that*

$$m_b \leftarrow \pi.\mathsf{Reconstruct}(b, (u_1^{b_1}, \ldots, u_n^{b_n})),$$

where $(b_1, \ldots, b_n) \leftarrow \pi.\mathsf{Choose}(b)$ and $(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}} \leftarrow \pi.\mathsf{Send}(m_0, \ldots, m_{q-1})$.

In the second scenario, we consider a malicious sender Adv and an honest receiver $\mathbb{B}$. We assume that Adv corrupts a set $A \in \mathcal{A}$ of servers, where $\mathcal{A} \subseteq 2^{\mathcal{P}_n}$ is an adversary structure preset according to the threat model of Adv, and where $\mathcal{P}_n$ represents the set of servers. We assume that Adv can see the inputs $(b_i)_{i\in A}$ of $\mathbb{B}$, and that she can also fix the messages $(z_i)_{i\in A}$ that $\mathbb{B}$ receives. Furthermore, she arbitrarily chooses inputs $(u_i^0, \ldots, u_i^{q-1})_{i\in\overline{A}}$ for the non-corrupted servers in $\overline{A}$.

Here correctness states that, regardless of how the malicious sender generates input for each server, the obtained protocol is still an OT protocol. That is, the message index b chosen by $\mathbb{B}$ should determine one and only one message, even if it is malformed (i.e. $\perp$, due to the malicious behavior of Alice). In particular, the received message, which is computed using $\pi.\mathsf{Reconstruct}$, should exclusively depend on b (and not on the randomness associated to the sharing of b sent by $\mathbb{B}$).

This second approach to correctness is formalized in the following definition, which uses the simulation paradigm [42], and which compares the execution of the protocol in the real world and in the ideal world.

In the real world, Adv and $\mathbb{B}$ interact through an OT combiner protocol π. The receiver $\mathbb{B}$ starts by choosing a message index $b \in \mathbb{F}_q$, and distributes each element b_i of the output of $\pi.\mathsf{Choose}(b)$ to each server. The adversary Adv is assumed to completely corrupt every server in a set $A \in \mathcal{A}$, and so she sees all the inputs $(b_i)_{i\in A}$ of $\mathbb{B}$ on those servers. Since the corruption is malicious, Adv also controls the outputs of servers in A, and so she chooses which output values z_i are received by $\mathbb{B}$ for $i \in A$. Non-corrupted servers $i \in \overline{A}$ are assumed to behave as the ideal $\mathcal{F}_{OT}$ functionality, so Adv sends q messages $u_i^0, \ldots, u_i^{q-1}$ to each of them and learns no information from that interaction.

In the ideal world, the whole view and output of Adv is controlled by the simulator Sim, and Sim and $\mathbb{B}$ interact exclusively through the ideal OT functionality $\mathcal{F}_{OT}$. Because of this, the adversary Adv does not receive anything from the interaction. By processing all the output that the adversary Adv generates, Sim produces a set of messages $\tilde{m}_0,\dots,\tilde{m}_{q-1}$ and handles them to the $\mathcal{F}_{OT}$ functionality, which outputs the message $\tilde{m}_b$ to $\mathbb{B}$ for the requested message index $b \in \mathbb{F}_q$.

In order to ensure that π behaves as an OT protocol in this setting, we should guarantee the indistinguishability between the reconstruction output by $\mathbb{B}$ in the real world and the view of $\mathbb{B}$ in the ideal world.

Definition 9. *Let π be a* 1*-out-of-q, n-server OT combiner protocol, and let $\mathcal{F}_{OT}$ denote the ideal* 1*-out-of-q OT functionality. Let* Adv *denote the adversary-controlled malicious sender, which is assumed to corrupt the set of servers indexed by some set $A \in \mathcal{A}$. Let $\mathbb{B}$ denote the honest receiver, and let* $\mathsf{Sim} = (\mathsf{Sim}_1, \mathsf{Sim}_2)$ *be a stateful simulator. We define the probabilistic experiments* $\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ *as follows:*

```
$\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ :
    $b \leftarrow \mathbb{B}()$
    $(b_1,\dots,b_n) \leftarrow \pi.\mathsf{Choose}(b)$
    $\left((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A}\right) \leftarrow \mathsf{Adv}\left((b_i)_{i\in A}\right)$
    $\textit{output}\ \pi.\mathsf{Reconstruct}\left(b,\left((u_i^{b_i})_{i\in\overline{A}},(z_i)_{i\in A}\right)\right)$

$\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ :
    $b \leftarrow \mathbb{B}()$
    $(\boldsymbol{ready}) \leftarrow \mathcal{F}_{OT}(\boldsymbol{transfer}, b)$
    $(b_i)_{i\in A} \leftarrow \mathsf{Sim}_1()$
    $\left((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A}\right) \leftarrow \mathsf{Adv}((b_i)_{i\in A})$
    $(m_0,\dots,m_{q-1}) \leftarrow \mathsf{Sim}_2\left((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A}\right)$
    $(\boldsymbol{sent}, m_b) \leftarrow \mathcal{F}_{OT}(\boldsymbol{send}, m_0,\dots,m_{q-1})$
    $\textit{output}\ m_b$
```

We say that π implements the OT functionality correctly for the receiver against active $\mathcal{A}$-adversaries *if, for every set $A \in \mathcal{A}$, for all adversarial senders* Adv *corrupting the set of servers indexed by A, and for all honest receivers $\mathbb{B}$, there exists a simulator* Sim *such that the output values of* $\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ *are identically distributed, where the probabilities are taken over the random coins of π,* Adv*, $\mathbb{B}$ and* Sim*.*

4.2 Correctness Proofs

We start with the proof of correctness in the setting where all parties follow the OT combiner protocol honestly.

Theorem 3. *The OT combiner π_{OT} defined in Figure 5 is zero-error. That is, π_{OT} implements the* 1*-out-of-q OT functionality correctly provided both Alice and Bob are semi-honest.*

Proof. If Alice and Bob follow the protocol honestly, at the end of the protocol Bob receives the values $m_b^{(1,b_1)}, \dots, m_b^{(n,b_n)}$ for some sharing $[b]_\Sigma = (b_1, \dots, b_n) \in W_b$ of his input b. Since $\mathcal{S}_b$ is W_b-OT-compatible by Theorem 2, the set $\{(1, b_1), \dots, (n, b_n)\}$ is authorized for $\mathcal{S}_b$, and thus Bob can use $\texttt{Reconstruct}_{\mathcal{S}_b}$ to reconstruct the message m_b. □

Now, we consider the case of Definition 9, where Alice is controlled by an active adversary Adv.

Theorem 4. *Let $(\mathcal{A}, \mathcal{B})$ be an $\mathcal{R}_2$ pair of adversary structures, and assume that the adversary structure $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS Σ. Then the OT combiner π_{OT} defined in Figure 5 implements the OT functionality correctly for the receiver against active $\mathcal{A}$-adversaries (see Definition 9).*

Proof. We start by defining the simulator appearing in Definition 9, and we then compare the output of the ideal experiment to that of the real experiment in the security definition.

$\mathsf{Sim}_1()$: Generate a uniformly random sharing of $0 \in \mathbb{F}_q$,

$$[0]_\Sigma = (b_1^0, \dots, b_n^0).$$

Output $(b_i^0)_{i \in A}$.

$\mathsf{Sim}_2((u_i^j)_{i \in \overline{A}, j \in \mathbb{F}_q}, (u_i)_{i \in A})$: Retrieve, from the state of Sim, the previously generated sharing $[0]_\Sigma = (b_i^0)_{i \in \mathcal{P}_n}$, that was computed in the previous execution of Sim_1.

Generate uniformly random sharings of every nonzero element of $\mathbb{F}_q$,

$$\begin{aligned}[1]_\Sigma =& (b_1^1, \dots, b_n^1),\\ &\vdots\\ [q-1]_\Sigma =& (b_1^{q-1}, \dots, b_n^{q-1}),\end{aligned}$$

subject to the restriction that $b_i^k = b_i^0$ for every $k \in \mathbb{F}_q \backslash \{0\}$ and for every $i \in A$. Note that these sharings exist, because A is forbidden for Σ. In practice, this step requires showing a solution of a compatible system of $|A|$ linear equations.

Parse each u_i^j as $u_i^j = m_0^{(i,j)} || \cdots || m_{q-1}^{(i,j)}$ whenever it is possible. If some u_i^j is not of the specified form (as it has been malformed by Alice), set $m_k = \mathbf{0}$ for every $k \in \mathbb{F}_q$ such that $b_i^k = j$.

For every $k \in \mathbb{F}_q$, if m_k has not already been set to $\mathbf{0}$ in the previous step, then try to reconstruct Alice's input by executing

$$\texttt{Reconstruct}_{\mathcal{S}_k}\left(\{(m_k^{(i,b_i^k)}) : i \in \mathcal{P}_n\}\right).$$

If the reconstruction succeeds, let m_k be its output. Otherwise, set $m_k = \mathbf{0}$. Output $(m_0, \dots, m_{q-1})$.

In order to prove indistinguishability remind that, in the real world, Bob generates a sharing $[b]_\Sigma = (b_1, \ldots, b_n)$ of his input $b \in \mathbb{F}_q$. Note that the shares $(b_i)_{i\in A}$ correspond to the set $A \in \mathcal{A}$, which is forbidden for Σ. Hence, they are distributed identically to the A-shares in a uniformly random sharing of any other $b' \neq b$.

Because of the previous observation, the messages $\left((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q}, (z_i)_{i\in A}\right)$ generated by Adv are identically distributed in both the real and the ideal world. Also because of the previous observation, the A-shares of the sharing $[b]_\Sigma = (b_1, \ldots, b_n)$ generated in the real world and the shares $(b_i)_{i\in A}$ generated by Sim in the ideal world are indistinguishable.

Therefore, the reconstruction process of the messages m_b is carried in exactly the same way in the real world and in the ideal world. This proves indistinguishability. □

5 Security of our OT Combiner

This section deals with the security of our 1-out-of-q OT combiner. In Section 5.1, discuss the security definition used to capture the security properties of our construction. Then, in Section 5.2 we prove the security of our construction.

5.1 Security Definitions

The security notion considered by Cascudo et al. [15] is called unconditional security. An OT combiner is *unconditionally secure* if its security rests solely on the security assumptions of the OT candidate implementations. That is, if, provided the security of sufficiently many OT candidates holds, the resulting OT protocol is perfectly secure. Therefore, unconditional security guarantees that any attack on an OT combiner must forcibly break the security of sufficiently many of the OT candidate implementations in order to be successful.

As in [15], an OT combiner is called *perfectly secure* if it is both unconditionally secure and zero-error (see Definition 8 above).

In order to capture the notion of unconditional security, we formalize it into a simulator-based security definition [42]. We now give the definition of security that we employ in our work, namely *perfect security against active* $(\mathcal{A}, \mathcal{B})$*-adversaries*, which is adapted from [16,15] and uses the Universal Composability framework [14].

Given two adversary structures $\mathcal{A}, \mathcal{B} \subseteq 2^{\mathcal{P}_n}$, where $\mathcal{P}_n$ represents the set of servers, our security definition protects against two types of malicious adversaries: one that corrupts the sender Alice and a set of servers $A \in \mathcal{A}$, and one that corrupts the receiver Bob and a set of servers $B \in \mathcal{B}$. This respectively corresponds to the case that a set $A \in \mathcal{A}$ of the OT candidates are insecure for the receiver, and to the case that a set $B \in \mathcal{B}$ of the OT candidates are insecure for the sender. To deal with the Alice corruption case, we define the notion of *perfect security for the receiver against active* $\mathcal{A}$*-adversaries*, and in the Bob corruption case we define the notion of *perfect security for the sender against active $\mathcal{B}$-adversaries.*

In the Alice corruption case, we consider a malicious (i.e., active) adversary Adv that controls the sender Alice, that interacts with an honest receiver $\mathbb{B}$, and that is able to eavesdrop and fully operate each server in a set $A \in \mathcal{A}$. Our security aim here is to protect the confidentiality of the receiver's choice $b \in \mathbb{F}_q$. Hence, the ability to corrupt the servers in $A \in \mathcal{A}$ must give Adv no information on b.

This definition uses the simulation paradigm [42], and compares the execution of the protocol in the real world and in the ideal world. In the real world, Adv and $\mathbb{B}$ interact through an OT combiner protocol π. The setting of this experiment is equivalent to that of Definition 9. In the ideal world, the whole view and output of Adv is controlled by the simulator Sim, and Sim and $\mathbb{B}$ interact exclusively through the ideal OT functionality $\mathcal{F}_{OT}$. Because of this, in the ideal experiment the adversary Adv does not receive anything from the interaction.

To provide security against malicious senders, Sim takes all the information viewed by Adv in the ideal world, which is the one herself produced, so as to transform it to a view that should be indistinguishable to the information seen by Adv in the real world, which includes the private inputs of $\mathbb{B}$ on the corrupted servers.

Definition 10. *Let π be a* 1*-out-of-q, n-server OT combiner protocol, and let $\mathcal{F}_{OT}$ denote the ideal* 1*-out-of-q OT functionality. Let* Adv *denote an adversary-controlled malicious sender, which is assumed to corrupt all the servers indexed by some set $A \in \mathcal{A}$. Let $\mathbb{B}$ denote an honest receiver, and let* $\mathsf{Sim} = (\mathsf{Sim}_1, \mathsf{Sim}_{\mathsf{out}})$ *be a stateful simulator. We define the probabilistic experiments* $\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ *as follows:*

```
$\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ :
    $b \leftarrow \mathbb{B}()$
    $(b_1, \ldots, b_n) \leftarrow \pi.\mathsf{Choose}(b)$
    $\left((u_i^j)_{i \in \overline{A}, j \in \mathbb{F}_q}, (z_i)_{i \in A}\right) \leftarrow \mathsf{Adv}\left((b_i)_{i \in A}\right)$
    $\mathit{output}\ \left((b_i)_{i \in A}, (u_i^j)_{i \in \overline{A}, j \in \mathbb{F}_q}, (z_i)_{i \in A}\right)$
```

```
$\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ :
    $b \leftarrow \mathbb{B}()$
    $(\boldsymbol{\mathit{ready}}) \leftarrow \mathcal{F}_{OT}(\boldsymbol{\mathit{transfer}}, b)$
    $(b_i)_{i \in A} \leftarrow \mathsf{Sim}_1()$
    $\left((u_i^j)_{i \in \overline{A}, j \in \mathbb{F}_q}, (z_i)_{i \in A}\right) \leftarrow \mathsf{Adv}((b_i)_{i \in A})$
    $\mathit{output}$ $\mathsf{Sim}_{\mathsf{out}}\left((u_i^j)_{i \in \overline{A}, j \in \mathbb{F}_q}, (z_i)_{i \in A}\right)$
```

We say that π is perfectly secure for the receiver against active $\mathcal{A}$-adversaries *if, for every set $A \in \mathcal{A}$, for all adversarial senders* Adv *corrupting the set of servers indexed by A, and for all honest receivers $\mathbb{B}$, there exists a simulator* Sim *such that the output values of* $\mathsf{Real}^{\pi}_{\mathsf{Adv},\mathbb{B}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathsf{Adv},\mathbb{B},\mathsf{Sim}}()$ *are identically distributed, where the probabilities are taken over the random coins of π,* Adv*, $\mathbb{B}$ and* Sim.

In the Bob corruption case, we consider a malicious (i.e., active) adversary Adv that controls the receiver Bob, that interacts with an honest sender $\mathbb{A}$, and that is able to eavesdrop on and fully operate each server in a set $B \in \mathcal{B}$. Our security aim here is to protect the confidentiality of the sender's messages $m_0, \ldots, m_{q-1}$. Hence, the ability to corrupt the servers in $B \in \mathcal{B}$ must give Bob no information on $m_0, \ldots, m_{q-1}$ other than possibly one chosen message. As the previous definition, this definition uses the simulation paradigm [42] and compares the execution of the protocol in the real world and in the ideal world.

In the real world, $\mathbb{A}$ and Adv interact through an OT combiner protocol π. The sender $\mathbb{A}$, who is assumed to act honestly, holds messages $m_0, \ldots, m_{q-1}$ and generates the input $u_i^0, \ldots, u_i^{q-1}$ that is sent to server S_i for every $i \in \mathcal{P}_n$. The adversary Adv is assumed to completely corrupt every server in a set $B \in \mathcal{B}$, and so he sees all the inputs $(u_i^j)_{i \in B, j \in \mathbb{F}_q}$. He also acts as the receiver, generating an input b_i for the rest of servers $i \in \overline{B}$. Since the servers $i \in \overline{B}$ are assumed to behave as the ideal $\mathcal{F}_{OT}$ functionality, Adv receives $(u_i^{b_i})_{i \in \overline{B}}$ and learns no other information from that interaction.

In the ideal world, the whole view and output of Adv is controlled by the simulator Sim, and Sim and $\mathbb{A}$ interact through the ideal OT functionality $\mathcal{F}_{OT}$. By processing all the output that the adversary Adv generates, Sim produces a message index $\tilde{b}$ and handles it to the $\mathcal{F}_{OT}$ functionality. Then, after the sender $\mathbb{A}$ has sent the messages $m_0, \ldots, m_{q-1}$ to $\mathcal{F}_{OT}$, the adversary Adv receives the message $m_{\tilde{b}}$. To provide security against malicious receivers, Sim takes all the information viewed by Adv in the ideal world, so as to transform it to a view that should be indistinguishable to the one of the real world.

Definition 11. *Let π be a* 1*-out-of-q, n-server OT combiner, and let $\mathcal{F}_{OT}$ denote the* 1*-out-of-q OT functionality. Let* Adv *denote an adversary-controlled malicious receiver, which is assumed to corrupt all the servers indexed by some set $B \in \mathcal{B}$. Let $\mathbb{A}$ denote an honest sender, and let* $\mathsf{Sim} = (\mathsf{Sim}_1, \mathsf{Sim}_2, \mathsf{Sim}_{\mathsf{out}})$

be a stateful simulator. We define the probabilistic experiments $\mathsf{Real}^{\pi}_{\mathbb{A},\mathsf{Adv}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathbb{A},\mathsf{Adv},\mathsf{Sim}}()$ *as follows:*

```
$\mathsf{Real}^{\pi}_{\mathbb{A},\mathsf{Adv}}()$ :
    $(m_0, \ldots, m_{q-1}) \leftarrow \mathbb{A}()$
    $(u_i^j)_{(i,j)\in\mathcal{P}_{n,q}} \leftarrow \pi.\mathsf{Send}(\boldsymbol{send}, m_0, \ldots, m_{q-1})$
    $(b_i)_{i\in\overline{B}} \leftarrow \mathsf{Adv}\left((u_i^j)_{i\in B, j\in\mathbb{F}_q}\right)$
    $\textit{output}\ \left((u_i^j)_{i\in B, j\in\mathbb{F}_q}, (u_i^{b_i})_{i\in\overline{B}}, (b_i)_{i\in\overline{B}}\right)$

$\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathbb{A},\mathsf{Adv},\mathsf{Sim}}()$ :
    $(u_i^j)_{i\in B, j\in\mathbb{F}_q} \leftarrow \mathsf{Sim}_1()$
    $(b_i)_{i\in\overline{B}} \leftarrow \mathsf{Adv}\left((u_i^j)_{i\in B, j\in\mathbb{F}_q}\right)$
    $\tilde{b} \leftarrow \mathsf{Sim}_2\left((b_i)_{i\in\overline{B}}\right)$
    $(\boldsymbol{ready}) \leftarrow \mathcal{F}_{OT}(\boldsymbol{transfer}, \tilde{b})$
    $(m_0, \ldots, m_{q-1}) \leftarrow \mathbb{A}()$
    $(\boldsymbol{sent}, m_{\tilde{b}}) \leftarrow \mathcal{F}_{OT}(\boldsymbol{send}, m_0, \ldots, m_{q-1})$
    $\textit{output}\ \mathsf{Sim}_{\mathsf{out}}\left(\tilde{b}, m_{\tilde{b}}, (b_i)_{i\in\overline{B}}\right)$
```

We say that π *is* perfectly secure for the sender against active $\mathcal{B}$-adversaries *if, for every* $B \in \mathcal{B}$*, for all adversarial receivers* Adv *corrupting the set of servers indexed by* B*, and for all honest senders* $\mathbb{A}$*, there exists a simulator* Sim *such that the output values of* $\mathsf{Real}^{\pi}_{\mathbb{A},\mathsf{Adv}}()$ *and* $\mathsf{Ideal}^{\mathcal{F}_{OT}}_{\mathbb{A},\mathsf{Adv},\mathsf{Sim}}()$ *are identically distributed, where the probabilities are taken over the random coins of* π*,* $\mathbb{A}$*,* Adv *and* Sim*.*

The two previous definitions, on top of the correctness definitions, make up the security definition considered in this work, namely perfect security against active $(\mathcal{A}, \mathcal{B})$-adversaries. We formally state this as follows.

Definition 12. *Let* π *be a* 1*-out-of-*q*,* n*-server OT combiner, and let* $\mathcal{A}, \mathcal{B} \subseteq 2^{\mathcal{P}_n}$*. We say that* π *is* perfectly secure against active $(\mathcal{A}, \mathcal{B})$-adversaries *if it is perfectly secure for the sender against active* $\mathcal{B}$*-adversaries and for the receiver against active* $\mathcal{A}$*-adversaries, it is zero-error, and it implements the OT functionality correctly for the receiver against active* $\mathcal{A}$*-adversaries.*

Finally, we state a result that characterizes the pairs $(\mathcal{A}, \mathcal{B})$ of adversary structures for which perfectly secure OT combiners are known to be impossible to attain.

Proposition 1 ([16]). *If* $(\mathcal{A}, \mathcal{B})$ *is not an* $\mathcal{R}_2$ *pair of adversary structures, then perfectly secure OT combiners against active* $(\mathcal{A}, \mathcal{B})$*-adversaries cannot exist.*

5.2 Security Proofs

The following theorem states the security properties of our construction.

Theorem 5. *Let $(\mathcal{A}, \mathcal{B})$ be an $\mathcal{R}_2$ pair of adversary structures, and assume that the adversary structure $\mathcal{A}$ admits an ideal $\mathbb{F}_q$-LSSS Σ. Then the OT combiner π_{OT} defined in Figure 5 is perfectly secure against active $(\mathcal{A}, \mathcal{B})$-adversaries (see Definition 12).*

Before proceeding with a proof, we need to prove the following lemma. Suppose that an adversary controlling Bob corrupts a set $B \in \mathcal{B}$ of servers. As a consequence of this lemma, if the shares $(b_i)_{i \in \overline{B}}$ sent to non-corrupted servers in $\overline{B}$ do not correspond to any sharing $[b]_\Sigma$ of b, the adversary can not get any information on the message m_b.

Lemma 3. *Let $m_0, \dots, m_{q-1} \in \mathbb{F}_q$ be arbitrary messages, and fix independent sharings $[m_k]_{\mathcal{S}_k} = (m_k^{(i,j)})_{(i,j) \in \mathcal{P}_{n,q}}$ for every $k \in \mathbb{F}_q$. Let $B \subseteq \{1, \dots, n\}$ and $(b'_1, \dots, b'_n) \in \mathbb{F}_q^n$, and define the set $\mathcal{H} \subseteq \mathcal{P}_{n,q}$ by*

$$\mathcal{H} = \{(i, b'_i) \, : \, i \in \overline{B}\} \cup \{(i, j) \, : \, i \in B, \ j \in \mathbb{F}_q\}.$$

Fix $b \in \mathbb{F}_q$. Then, if the shares $(b'_i)_{i \in \overline{B}}$ are not part of any sharing $[b]_\Sigma$, the shares

$$\{m_k^{(i,j)} \, : \, (i,j) \in \mathcal{H}, \ k \in \mathbb{F}_q\}$$

give no information about m_b.

Proof. Since the sharing of every message is done independently, the only shares that could potentially give any information on m_b are $(m_b^{(i,j)})_{(i,j) \in \mathcal{H}}$. Hence, we need to prove that $\mathcal{H}$ is forbidden for $\mathcal{S}_b$. Since $\mathcal{S}_b$ is W_b-OT-compatible and since $\mathcal{H} \in \Delta$, if $\mathcal{H}$ were authorized for $\mathcal{S}_b$ then $\mathcal{H} \in \Gamma_{W_b}$, and thus it would contain a set $\{(1, b_1), \dots, (n, b_n)\}$ for some $(b_1, \dots, b_n) \in W_b$. However, then necessarily $b_i = b'_i$ for all $i \in \overline{B}$, and this would mean that $(b'_i)_{i \in \overline{B}}$ belongs to a sharing $[b]_\Sigma$, a contradiction. □

We can now proceed to the proof of Theorem 5.

Proof. Correctness is proved in Theorems 3 and 4. The rest of the proof is split in two parts, corresponding to Definitions 10 and 11. In each case, we define the simulators and compare the output of the ideal experiment to that of the real experiment.

Perfect security for the receiver against active $\mathcal{A}$-adversaries:

$\mathsf{Sim}_1()$: Generate a uniformly random sharing of $0 \in \mathbb{F}_q$,

$$[0]_\Sigma = (b_1^0, \dots, b_n^0).$$

Output $(b_i^0)_{i \in A}$.

$\mathsf{Sim}_{\mathsf{out}}((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A})$: Retrieve, from the state of Sim, the sharing $[0]_\Sigma = (b_i^0)_{i\in\mathcal{P}_n}$ that was generated in the previous execution of Sim_1.
Output $\Big((b_i^0)_{i\in A},(u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A}\Big)$.

We prove indistinguishability in a similar fashion than in Theorem 4.

Note that the shares $(b_i)_{i\in A}$ that the adversary Adv takes as input correspond to the set $A\in\mathcal{A}$, which is forbidden for Σ. Because of this, these shares are distributed identically to the A-shares in a uniformly random sharing of any other $b'\neq b$ (in particular, of $0\in\mathbb{F}_q$). Moreover, they do not carry any information on b, so the messages $\Big((u_i^j)_{i\in\overline{A},j\in\mathbb{F}_q},(z_i)_{i\in A}\Big)$ generated by Adv are identically distributed in both worlds.

Since the shares $(b_i)_{i\in A}$ do not allow to distinguish between the real and the ideal world, we have proved indistinguishability.

Perfect security for the sender against active $\mathcal{B}$-adversaries:

$\mathsf{Sim}_1()$: For every $k\in\mathbb{F}_q$, choose $m'_k\in\mathbb{F}_q$ at random and generate the sharing

$$[m'_k]_{\mathcal{S}_k} = ({m'}_k^{(i,j)})_{(i,j)\in\mathcal{P}_{n,q}}.$$

Then, create the values $u_i^j = {m'}_0^{(i,j)}||\cdots||{m'}_{q-1}^{(i,j)}$ for every $(i,j)\in B\times\mathbb{F}_q$.
Output $(u_i^j)_{i\in B,j\in\mathbb{F}_q}$.

$\mathsf{Sim}_2((b_i)_{i\in\overline{B}})$: Try to reconstruct the input b of the adversary Adv by executing the $\texttt{Reconstruct}_\Sigma$ function over the input to non-corrupted servers, i.e., by executing $\texttt{Reconstruct}_\Sigma((b_i)_{i\in\overline{B}})$.
If the reconstruction succeeds, output the reconstructed message index $\tilde{b}$.
If the reconstruction fails, output $\perp$.

$\mathsf{Sim}_{\mathsf{out}}(\tilde{b},m_{\tilde{b}},(b_i)_{i\in\overline{B}})$: Retrieve, from the state of Sim and for every k, the messages m'_k, the sharings $[m'_k]_{\mathcal{S}_k} = ({m'}_k^{(i,j)})_{(i,j)\in\mathcal{P}_{n,q}}$ and the messages $(u_i^j)_{i\in B,j\in\mathbb{F}_q}$ that were generated in the previous execution of Sim_1.
Proceed as follows, depending on whether the reconstruction in Sim_2 failed or not:

- If $\tilde{b}\neq\perp$, let $\tilde{m}_{\tilde{b}} = m_{\tilde{b}}$ and $\tilde{m}_k = {m'}_k$ for $k\in\mathbb{F}_q\setminus\{\tilde{b}\}$. Then, generate a sharing
$$[\tilde{m}_{\tilde{b}}]_{\mathcal{S}_{\tilde{b}}} = ({m'}_{\tilde{b}}^{(i,j)})_{(i,j)\in\mathcal{P}_{n,q}}$$
subject to the restriction that $\tilde{m}_{\tilde{b}}^{(i,j)} = {m'}_{\tilde{b}}^{(i,j)}$ for every $(i,j)\in B\times\mathbb{F}_q$ (note that this is possible, since $B\times\mathbb{F}_q$ is forbidden for $\mathcal{S}_0,\dots,\mathcal{S}_{q-1}$).
For every $k\in\mathbb{F}_q\setminus\{\tilde{b}\}$, set
$$\tilde{m}_k^{(i,j)} = {m'}_k^{(i,j)} \text{ for every } (i,j)\in\mathcal{P}_{n,q}.$$
- If $\tilde{b}=\perp$ then, for every $k\in\mathbb{F}_q$, let
$$\tilde{m}_k = {m'}_k$$
$$\tilde{m}_k^{(i,j)} = {m'}_k^{(i,j)} \text{ for every } (i,j)\in\mathcal{P}_{n,q}.$$

Create the values $u_i^{b_i} = \tilde{m}_0^{(i,b_i)}||\cdots||\tilde{m}_{q-1}^{(i,b_i)}$ for every $i \in \mathcal{P}_n$.
Output $\Big((u_i^j)_{i\in B, j\in\mathbb{F}_q}, (u_i^{b_i})_{i\in\overline{B}}, (b_i)_{i\in\overline{B}}\Big)$.

In order to prove indistinguishability we first note that, by Lemma 1, the set $\overline{B}$ is authorized for Σ. By the definition of $\mathcal{S}_k$, we see that at least one share per server is needed to reconstruct a message. Hence, the set $B \times \mathbb{F}_q$ is forbidden for $\mathcal{S}_0, \dots, \mathcal{S}_{q-1}$, and so the shares $(u_i^j)_{i\in B, j\in\mathbb{F}_q}$ do not hold any information on the messages $m_0, \dots, m_{q-1}$. Therefore, the shares $(b_i)_{i\in\overline{B}}$ generated by the adversary Adv in the real world and in the ideal world are identically distributed.

Now, since $\overline{B}$ is authorized for Σ, we have two possibilities regarding the shares $(b_i)_{i\in\overline{B}}$ received by Sim: either they are part of a sharing $[b]_\Sigma$, or they are not part of any sharing under Σ (due to the malicious behavior of Adv).

In the first case, Sim_2 successfully reconstructs b. The set

$$\{(i, b_i) \ : \ i \in \overline{B}\} \cup (B \times \mathbb{F}_q)$$

is then authorized for $\mathcal{S}_b$ and, by Lemma 3, it is forbidden for all the other $\mathbb{F}_q$-LSSS $\mathcal{S}_k$. Since the sharings for m_b generated by $\mathsf{Sim}_{\mathsf{out}}$ are distributed identically to those of the real world, this proves indistinguishability.

In the second case, Lemma 3 shows that the shares output by $\mathsf{Sim}_{\mathsf{out}}$ give no information about m_b. Therefore, since here $\mathsf{Sim}_{\mathsf{out}}$ generates them from random messages, they obey the same distribution as in the real world, as required. □

Finally, we can prove Theorem 1.

Proof (Theorem 1). It follows from Remark 2, Remark 3, and Theorem 5.

6 Conclusions

This work tackles OT combiners for 1-out-of-q OT protocols in the case that $q \geq 2$ is a prime power. In this case, we build a 1-out-of-q OT combiner by extending the work of Cascudo, Damgård, Farràs and Ranellucci [15], which in turn is based in the construction by Ishai, Maji, Sahai and Wullschleger [35]. Our OT combiner is black-box and single-use. The construction in [15], as ours, is proved secure against malicious adversaries corrupting either one of the parties and a certain set of OT candidates.

The main obstacle when trying to extend the construction in [15] was building an ideal $\mathbb{F}_q$-linear secret sharing scheme for Γ_W where $W \subseteq \mathbb{F}_q^n$ is an affine space, because it is not possible. That is, Γ_W is not ideal, in general. We circumvent this problem by relaxing the restrictions on the access structure. We introduce the notion of W-OT-compatible secret sharing schemes, and we present one such scheme that fits our needs. We think that this construction is of independent interest. Characterizing the $W \subseteq \mathbb{F}_q^n$ for which Γ_W admits efficient schemes is an interesting open problem that is also related to the efficiency of CDS protocols.

We also extend the security and consistency notions of [15] to the 1-out-of-q case, and we present them in an explicit and formal form. The consistency and the security of our construction are proved according to these definitions. In particular, our construction uses the security notion of [16,15], called perfect security against active $(\mathcal{A}, \mathcal{B})$-adversaries.

Consider the particularly interesting case where adversaries are allowed to corrupt at most t servers for some $t < n/2$, or more generally, where $(\mathcal{A}, \mathcal{B})$ is an $\mathcal{R}_2$ pair of adversary structures satisfying $\binom{\mathcal{P}_n}{t} \subseteq \mathcal{A}, \mathcal{B}$. By [16], there exists a single-use 1-out-of-q OT combiner that is perfectly security against active adversaries for $t = \lfloor 0.11n \rfloor$. Increasing t in their case could take away the single-use property. In our construction, by choosing $q \geq n$ and $t = \lceil n/2 \rceil - 1$, we can take Σ as the $(t+1)$-threshold Shamir $\mathbb{F}_q$-LSSS and achieve perfect security while keeping the single-use property.

In the Sending phase of our protocol, we share each of the q messages independently. For $q = 2$, this process was improved in [16] by creating sharings of the two messages at the same time, which reduces the number of shares from $4n$ to $2n$. The scheme in [16] can be seen as a *multi-secret* sharing scheme [38,12,28]. In such schemes, n shares are generated from a sequence of $k > 1$ secrets, and each secret can be recovered from the shares, but each secret has its own access structure. Observe that we can define our 1-out-of-q construction from multi-secret sharing schemes. Since our multi-secret sharing scheme is just a combination of independent secret sharing schemes, we decided simplify the notation. However, a research line in the direction of this work is to build more efficient 1-out-of-q OT-combiners with multi-secret sharing schemes.

Acknowledgments

This article is supported by the Government of Catalonia (grant 2017 SGR 705), by the Spanish Government (project RTI2018-095094-B-C21, “CONSENT”), and by the DRAC project, which is co-financed by the European Union Regional Development Fund within the framework of the ERDF Operational Program of Catalonia 2014-2020 with a grant of 50% of total cost eligible.

References

1. Aiello, B., Ishai, Y., Reingold, O.: Priced oblivious transfer: How to sell digital goods. In: Pfitzmann, B. (ed.) Advances in Cryptology — EUROCRYPT 2001. pp. 119–135. Springer, Berlin, Heidelberg (2001)
2. Applebaum, B., Beimel, A., Nir, O., Peter, N.: Better secret sharing via robust conditional disclosure of secrets. In: Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing. pp. 280–293 (2020)
3. Applebaum, B., Beimel, A., Farràs, O., Nir, O., Peter, N.: Secret-sharing schemes for general and uniform access structures. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2019. pp. 441–471. Springer International Publishing (2019)
4. Asmuth, C.A., Blakley, G.R.: An efficient algorithm for constructing a cryptosystem which is harder to break than two other cryptosystems. Computers & Mathematics with Applications **7**(6), 447–450 (1981)

5. Beimel, A., Farràs, O.: The share size of secret-sharing schemes for almost all access structures and graphs. In: Theory of Cryptography Conference. pp. 499–529. Springer (2020)
6. Beimel, A.: Secret-sharing schemes: A survey. In: Proceedings of the Third International Conference on Coding and Cryptology. p. 11–46. IWCC'11, Springer-Verlag, Berlin, Heidelberg (2011)
7. Beimel, A., Ishai, Y.: On the power of nonlinear secret-sharing. SIAM Journal on Discrete Mathematics **19**(1), 258–280 (May 2005)
8. Beimel, A., Othman, H., Peter, N.: Degree-2 secret sharing and conditional disclosure of secrets. IACR Cryptol. ePrint Arch. **2021**, 285 (2021), `https://eprint.iacr.org/2021/285`
9. Bellare, M., Micali, S.: Non-interactive oblivious transfer and applications. In: Brassard, G. (ed.) Advances in Cryptology — CRYPTO' 89 Proceedings. pp. 547–557. Springer, New York (1990)
10. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation. In: Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing. pp. 1–10. STOC '88, ACM, New York, USA (1988)
11. Blakley, G.R.: Safeguarding cryptographic keys. In: Proceedings of the AFIPS 1979 National Computer Conference. vol. 48, pp. 313–317. AFIPS Press (1979)
12. Blundo, C., De Santis, A., Di-Crescenzo, G., Giorgio Gaggia, A., Vaccaro, U.: Multi-secret sharing schemes. In: CRYPTO94. LNCS, vol. 839, pp. 150–163 (1994)
13. Cachin, C., Crépeau, C., Marcil, J.: Oblivious transfer with a memory-bounded receiver. In: Proceedings of the Annual Symposium on Foundations of Computer Science. pp. 493–502 (12 1998)
14. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067 (2000), `https://eprint.iacr.org/2000/067`
15. Cascudo, I., Damgård, I., Farràs, O., Ranellucci, S.: Server-aided two-party computation with minimal connectivity in the simultaneous corruption model. Cryptology ePrint Archive, Report 2014/809 (2014), `https://eprint.iacr.org/2014/809`
16. Cascudo, I., Damgård, I., Farràs, O., Ranellucci, S.: Resource-efficient OT combiners with active security. In: 15th International Conference on Theory of Cryptography, Part II. Lecture Notes in Computer Science, vol. 10678, pp. 461–486. Springer (2017)
17. Chor, B., Goldreich, O., Kushilevitz, E.: Private information retrieval. Journal of the ACM pp. 41–50 (1995)
18. Chor, B., Kushilevitz, E.: A zero-one law for boolean privacy. In: Proceedings of the Twenty-first Annual ACM Symposium on Theory of Computing. pp. 62–72. STOC '89, ACM, New York, USA (1989)
19. Crépeau, C., Brassard, G., Robert, J.M.: Information theoretic reductions among disclosure problems. In: 27th Annual Symposium on Foundations of Computer Science (sfcs 1986)(FOCS). vol. 00, pp. 168–173 (10 1986)
20. Crépeau, C., Kilian, J.: Achieving oblivious transfer using weakened security assumptions (extended abstract). In: 29th Annual Symposium on Foundations of Computer Science. pp. 42–52 (1988)
21. Csirmaz, L.: The size of a share must be large. J. Cryptology **10**, 223–231 (1997)
22. Dierks, T., Rescorla, E.: The Transport Layer Security (TLS) Protocol Version 1.1. RFC 4346 (Apr 2006), `https://rfc-editor.org/rfc/rfc4346.txt`
23. Dodis, Y., Katz, J.: Chosen-ciphertext security of multiple encryption. In: Kilian, J. (ed.) Theory of Cryptography. pp. 188–209. Springer, Berlin, Heidelberg (2005)

24. Dowsley, R., van de Graaf, J., Müller-Quade, J., Nascimento, A.C.A.: Oblivious transfer based on the McEliece assumptions. In: Safavi-Naini, R. (ed.) Information Theoretic Security. pp. 107–117. Springer, Berlin, Heidelberg (2008)
25. Döttling, N., Kraschewski, D., Müller-Quade, J.: David & goliath oblivious affine function evaluation - asymptotically optimal building blocks for universally composable two-party computation from a single untrusted stateful tamper-proof hardware token. Cryptology ePrint Archive, Rep. 2012/135 (2012), `https://eprint.iacr.org/2012/135`
26. Even, S., Goldreich, O., Lempel, A.: A randomized protocol for signing contracts. Communications of the ACM **28**(6), 637–647 (Jun 1985)
27. Farràs, O., Kaced, T., Martín, S., Padró, C.: Improving the linear programming technique in the search for lower bounds in secret sharing. IEEE Transactions on Information Theory (2020). https://doi.org/10.1109/TIT.2020.3005706, full version of [**?**]
28. Farràs, O., Gracia, I., Molleví, S.M., Padró, C.: Linear threshold multisecret sharing schemes. Inf. Process. Lett. **112**(17-18), 667–673 (2012)
29. Gertner, Y., Ishai, Y., Kushilevitz, E., Malkin, T.: Protecting data privacy in private information retrieval schemes. J. Comput. Syst. Sci. **60**(3), 592–629 (2000)
30. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game. In: Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing. pp. 218–229. STOC '87, ACM, New York, USA (1987)
31. Goyal, V., Ishai, Y., Sahai, A., Venkatesan, R., Wadia, A.: Founding cryptography on tamper-proof hardware tokens. IACR Cryptology ePrint Archive **2010**, 153 (2010)
32. Harnik, D., Ishai, Y., Kushilevitz, E., Buus Nielsen, J.: OT-combiners via secure computation. In: Canetti, R. (ed.) Theory of Cryptography. pp. 393–411. Springer, Berlin, Heidelberg (2008)
33. Harnik, D., Kilian, J., Naor, M., Reingold, O., Rosen, A.: On robust combiners for oblivious transfer and other primitives. In: Cramer, R. (ed.) Advances in Cryptology – EUROCRYPT 2005. pp. 96–113. Springer, Berlin, Heidelberg (2005)
34. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM Journal on Computing **28**(4), 1364–1396 (Mar 1999)
35. Ishai, Y., Maji, H.K., Sahai, A., Wullschleger, J.: Single-use OT combiners with near-optimal resilience. In: International Symposium on Information Theory. pp. 1544–1548. IEEE (2014)
36. Ishai, Y., Prabhakaran, M., Sahai, A.: Founding cryptography on oblivious transfer — efficiently. In: Proceedings of the 28th Annual Conference on Cryptology: Advances in Cryptology. pp. 572–591. CRYPTO 2008, Springer-Verlag, Berlin, Heidelberg (2008)
37. Ito, M., Saito, A., Nishizeki, T.: Secret sharing scheme realizing general access structure. Electronics and Communications in Japan, Part III **72**(9), 56–64 (1989)
38. Jackson, W., Martin, K.M., O'Keefe, C.M.: Multisecret threshold schemes. In: CRYPTO93. LNCS, vol. 773, pp. 126–135 (1994)
39. Kilian, J.: Founding cryptography on oblivious transfer. In: Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing. pp. 20–31. STOC '88, ACM, New York, USA (1988)
40. Kilian, J., Micali, S., Ostrovsky, R.: Minimum resource zero-knowledge proofs. In: Brassard, G. (ed.) Advances in Cryptology — CRYPTO' 89 Proceedings. pp. 545–546. Springer, New York (1990)

41. Kolesnikov, V., Kumaresan, R., Rosulek, M., Trieu, N.: Efficient batched oblivious prf with applications to private set intersection. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 818–829. CCS '16, ACM, New York, USA (2016)
42. Lindell, Y.: How to simulate it - a tutorial on the simulation proof technique. In: Tutorials on the Foundations of Cryptography (2016)
43. Liu, T., Vaikuntanathan, V.: Breaking the circuit-size barrier in secret sharing. In: Diakonikolas, I., Kempe, D., Henzinger, M. (eds.) Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018. pp. 699–708. ACM (2018)
44. Liu, T., Vaikuntanathan, V., Wee, H.: Towards breaking the exponential barrier for general secret sharing. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part I. Lecture Notes in Computer Science, vol. 10820, pp. 567–596. Springer (2018)
45. Meier, R., Przydatek, B.: On robust combiners for private information retrieval and other primitives. In: Dwork, C. (ed.) Advances in Cryptology - CRYPTO 2006. pp. 555–569. Springer, Berlin, Heidelberg (2006)
46. Meier, R., Przydatek, B., Wullschleger, J.: Robuster combiners for oblivious transfer. In: Vadhan, S.P. (ed.) Theory of Cryptography. pp. 404–418. Springer, Berlin, Heidelberg (2007)
47. Naor, M., Pinkas, B.: Oblivious transfer and polynomial evaluation. In: Proceedings of the Thirty-first Annual ACM Symposium on Theory of Computing. pp. 245–254. STOC '99, ACM, New York, USA (1999)
48. Naor, M., Pinkas, B.: Computationally secure oblivious transfer. Journal of Cryptology **18**(1), 1–35 (2005)
49. Orrù, M., Orsini, E., Scholl, P.: Actively secure 1-out-of-n OT extension with application to private set intersection. In: Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings. pp. 381–396 (2017)
50. Padró, C.: Lecture notes in secret sharing. IACR Cryptology ePrint Archive p. 674 (2012), http://eprint.iacr.org/2012/674
51. Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) Advances in Cryptology – CRYPTO 2008. pp. 554–571. Springer, Berlin, Heidelberg (2008)
52. Przydatek, B., Wullschleger, J.: Error-tolerant combiners for oblivious primitives. In: Aceto, L., Damgård, I., Ann Goldberg, L., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) Automata, Languages and Programming. pp. 461–472. Springer, Berlin, Heidelberg (2008)
53. Rabin, M.O.: How to exchange secrets with oblivious transfer (2005), http://eprint.iacr.org/2005/187, harvard University Technical, Report 81
54. Shamir, A.: How to share a secret. Communications of the ACM **22**(11), 612–613 (1979)
55. Tzeng, W.G.: Efficient 1-out-n oblivious transfer schemes. In: Naccache, D., Paillier, P. (eds.) Public Key Cryptography. pp. 159–171. Springer, Berlin, Heidelberg (2002)
56. Vaikuntanathan, V., Nalini Vasudevan, P.: Secret sharing and statistical zero knowledge. In: Proceedings, Part I, of the 21st International Conference on Advances in Cryptology – ASIACRYPT 2015 - Volume 9452. pp. 656–680. Springer-Verlag, Inc., New York, USA (2015)

57. Yao, A.C.C.: Protocols for secure computations. In: Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. pp. 160–164. SFCS '82, IEEE Computer Society, Washington, DC, USA (1982)

A Proof of Theorem 2

To prove this theorem, we first need the following technical lemma.

Lemma 4. *Let $\mathbb{F}_q$ be a finite field with $q \geq 2$ and $V \subset \mathbb{F}_q^n$ be a vector subspace. Let $t \leq n$ and $y_1, \dots, y_t \in \mathbb{F}_q$. If $(y_1, \dots, y_t, x_{t+1}, \dots, x_n) \notin V$ for every $x_{t+1}, \dots, x_n \in \mathbb{F}_q$, then there exists $\mathbf{h} \in V^\perp$ such that $y_1 h_1 + \cdots + y_t h_t = 1$ and $h_{t+1} = \cdots = h_n = 0$.*

Proof. The lemma holds for $t = n$ since, given $y = (y_1, \dots, y_n) \notin V$, there always exists an $\mathbf{h} \in V^\perp$ such that $\langle y, \mathbf{h} \rangle = 1$.

Now, assume that $t < n$, and that we have $y_1, \dots, y_t \in \mathbb{F}_q$ such that

$$(y_1, \dots, y_t, x_{t+1}, \dots, x_n) \notin V \text{ for all } x_{t+1}, \dots, x_n \in \mathbb{F}_q.$$

By induction hypothesis we have that, for every $x \in \mathbb{F}_q$, there exists an $\mathbf{h}^x = (h_1^x, \dots, h_n^x) \in V^\perp$ such that

$$\begin{aligned} \sum_{i=1}^{t} y_i h_i^x + x h_{t+1}^x = 1 \\ h_{t+2}^x = \cdots = h_n^x = 0. \end{aligned}$$

If $h_{t+1}^x = 0$, for some $x \in \mathbb{F}_q$, then $\mathbf{h}^x$ satisfies the lemma. Otherwise, by the pigeonhole principle, let x and x' be two distinct elements of $\mathbb{F}_q$ such that $h_{t+1}^x = h_{t+1}^{x'} \neq 0$. Define

$$\mathbf{h} = \frac{\mathbf{h}^x - \mathbf{h}^{x'}}{h_{t+1}^x (x' - x)} \in V^\perp.$$

Since $\mathbf{h} = (h_1, \dots, h_n)$ satisfies $h_{t+1} = \cdots = h_n = 0$ and

$$\begin{aligned} y_1 h_1 + \cdots + y_t h_t &= \frac{1}{h_{t+1}^x (x' - x)} \left(\sum_{i=1}^{t} y_i h_i^x - \sum_{i=1}^{t} y_i h_i^{x'} \right) \\ &= \frac{1}{h_{t+1}^x (x' - x)} \Big((1 - x h_{t+1}^x) - (1 - x' h_{t+1}^{x'}) \Big) \\ &= 1 \end{aligned}$$

we have that $\mathbf{h}$ satisfies the lemma. □

We next prove Theorem 2.

Proof. In order to share a secret $m \in \mathbb{F}_q$ in the considered scheme $\mathcal{S}_k$, the sender chooses $r_1, \ldots, r_{n-1} \in \mathbb{F}_q$ uniformly at random, sets $r_n = m - \sum_{i=1}^{n-1} r_i$ and chooses $\mathbf{h} = (h_1, \ldots, h_n) \in V^{\perp}$ uniformly at random. The share of participant (i, j) is, then $m^{(i,j)} = r_i + (k_i - j)h_i$, where $\mathbf{k} = (k_1, \ldots, k_n)$ is a sharing of k using the ideal $\mathbb{F}_q$-LSSS Σ, and we denote $W_k = \mathbf{k} + V$. This scheme is ideal, since each participant in $\mathcal{P}_{n,q}$ is assigned a single share in $\mathbb{F}_q$, and it is $\mathbb{F}_q$-linear.

Now we prove that the access structure Γ of the considered secret sharing scheme is W_k-OT-compatible.

On one hand, we must prove that $\Gamma_{W_k} \cap \Delta \subseteq \Gamma \cap \Delta$. Let $\mathbf{w} = (w_1, \ldots, w_n) \in W$ and set $A = \{(1, w_1), \ldots, (n, w_n)\}$. Since $\mathbf{w} = \mathbf{k}+\mathbf{v}$ for some $\mathbf{v} = (v_1, \ldots, v_n) \in V$, we have

$$\sum_{(i,j)\in A} m^{(i,j)} = \sum_{i=1}^{n}(r_i + (k_i - w_i)h_i) = \sum_{i=1}^{n} r_i - \langle \mathbf{v}, \mathbf{h} \rangle = \sum_{i=1}^{n} r_i = m$$

and so $\{(1, w_1), \ldots, (n, w_n)\} \in \min \Gamma$ for every $\mathbf{w} \in W_k$. Hence, $\Gamma_{W_k} \subseteq \Gamma$.

On the other hand, we prove $\Gamma \cap \Delta \subseteq \Gamma_{W_k} \cap \Delta$ by showing that, for every $A \in \Delta$, if $A \notin \Gamma_{W_k}$ then $A \notin \Gamma$. Assume, without loss of generality, that

$$A = \{(1, v_1), \ldots, (t, v_t)\} \cup P_{t+1} \cup \cdots \cup P_n.$$

Hence, we have that $(v_1, \ldots, v_t, x_{t+1}, \ldots, x_n) \notin W_k$ for every $x_{t+1}, \ldots, x_n \in \mathbb{F}_q$. By the previous lemma, there exists an $\mathbf{h} = (h_1, \ldots, h_n) \in V^{\perp}$ such that $\sum_{i=1}^{t}(v_i - k_i)h_i = 1$ and $h_{t+1} = \cdots = h_n = 0$.

By considering such an $\mathbf{h} \in V^{\perp}$ and the following choice of randomness

$$\begin{aligned} r_i &= (v_i - k_i)h_i && \text{for } i = 1, \ldots, t, \\ r_i &= 0 && \text{for } i = t+1, \ldots, n, \end{aligned}$$

we get a sharing of the message $m = 1$ such that $m^{(i,j)} = 0$ for every $(i, j) \in A$. The theorem follows by applying Lemma 2. □

B Examples of W-OT-Compatible Access Structures

We now give examples of W-OT-compatible access structures.

Example 1. Consider the Γ_W access structure defined as follows. Let $n = q = 3$. Then,

$$P_1 = \{(1,0), (1,1), (1,2)\}, P_2 = \{(2,0), (2,1), (2,2)\}, P_3 = \{(3,0), (3,1), (3,2)\}$$

$$\mathcal{P}_{3,3} = P_1 \cup P_2 \cup P_3$$

Let $W = \langle (0, 1, 2), (1, 0, 2) \rangle \subseteq \mathbb{F}_3^3$. The vector subspace W has 9 vectors, and so Γ_W has 9 minimal authorized subsets. It can be checked that Γ_W does not admit an ideal linear secret sharing scheme [21,27]. Indeed, Γ_W is not a matroid port, and so the information ratio of schemes realizing it is at least 3/2.

Example 2. Let $n = q = 3$ as in the previous example. Then,

$$\Delta = \{A \subseteq \mathcal{P}_{3,3} \,:\, |A \cap P_i| = 0, 1 \text{ or } 3 \text{ for } i = 1, 2, 3\}.$$

Note that $|\Delta| = \left(\binom{3}{0} + \binom{3}{1} + \binom{3}{3}\right)^3 = 125$. Let $W \subseteq \mathbb{F}_3^3$ be the affine subspace defined by $W = \mathbf{k} + V$, where

$$\begin{aligned}&\mathbf{k} = (1, 1, 1)\\&V = \langle (1, 0, 2)\rangle_{\mathbb{F}_3} = \{(0, 0, 0), (1, 0, 2), (2, 0, 1)\},\end{aligned}$$

so $W = \{(1, 1, 1), (2, 1, 0), (0, 1, 2)\}$. The access structure Γ_W on $\mathcal{P}_{3,3}$ is defined by the minimal access structure

$$\min \Gamma_W = \{\{(1, 1), (2, 1), (3, 1)\}, \{(1, 2), (2, 1), (3, 0)\}, \{(1, 0), (2, 1), (3, 2)\}\}.$$

We note that Γ_W is trivially W-OT-compatible. To illustrate W-OT-compatibility, consider now the access structures $\Gamma_1, \Gamma_2, \Gamma_3$ on $\mathcal{P}_{3,3}$ determined by

$$\begin{aligned}&\min \Gamma_1 = \{\{(1, 1), (2, 1), (3, 1)\}, \{(1, 2), (2, 1), (3, 0)\}\}\\&\min \Gamma_2 = \{(2, 1)\}\\&\min \Gamma_3 = \min \Gamma_W \cup \{(1, 1), (2, 1), (3, 2), (3, 3)\}\end{aligned}$$

Since $\{(1, 0), (2, 1), (3, 2)\}$ is in $\Gamma_W \cap \Delta$ but not in Γ_1, we have that Γ_1 is not W-OT-compatible. As for Γ_2, while $\Gamma_W \subseteq \Gamma_2$, we have sets of $\Gamma_2 \cap \Delta$, such as P_2 or $\{(1, 1), (2, 1), (3, 2)\}$, that do not belong to Γ_W. In general, any W-OT-compatible access structure Γ must satisfy $\min \Gamma_W \subseteq \min \Gamma$.

Lastly, we see that Γ_3 is W-OT-compatible. This is because, for any set $A \in \Gamma_3 \cap \Delta$ that contains $\{(1, 1), (2, 1), (3, 2), (3, 3)\}$, we have that $(1, 1) \in A \cap P_1$, that $(2, 1) \in A \cap P_2$ and $A \cap P_3 = P_3$. Hence, A contains $\{(1, 1), (2, 1), (3, 1)\}$, and so $\Gamma_3 \cap \Delta \subseteq \Gamma_W \cap \Delta$. This demonstrates that W-OT-compatible access structures may have minimal access structure outside of $\min \Gamma_W$.

C Our One-out-of-q OT Combiner in the Non-Ideal Case

In this section, we show how our protocol π_{OT} from Section 3.3 extends to the general case where the adversary structure $\mathcal{A}$ does not necessarily admit an ideal $\mathbb{F}_q$-linear secret sharing scheme.

C.1 OT-Compatible Secret Sharing Schemes

Let Σ be an $\mathbb{F}_q$-linear secret sharing scheme for n participants with adversary structure $\mathcal{A}$. Since Σ is now not necessarily ideal, if $[b]_\Sigma = (\tilde{b}_1, \dots, \tilde{b}_n)$ is a sharing of b using Σ, we note that each share $\tilde{b}_i$ belongs to some vector space $E_i = \mathbb{F}_q^{\ell_i}$ for some integer $\ell_i \geq 1$. Hence, unlike in the ideal case, $\tilde{b}_i$ may not correspond to a message index, and in this case Bob can not just send the share $\tilde{b}_i$ to each server S_i.

Instead, denote by $\ell = \sum_{i=1}^{n} \ell_i$ the complexity of Σ. Rather than looking at the sharings $(\tilde{b}_1, \dots, \tilde{b}_n)$ as elements of $\mathbb{F}_q^{\ell_1} \times \cdots \times \mathbb{F}_q^{\ell_n}$, we concatenate their components and we see them as elements of the vector space $\mathbb{F}_q^{\ell}$. Denote the corresponding vector space isomorphism by

$$\varphi : \mathbb{F}_q^{\ell_1} \times \cdots \times \mathbb{F}_q^{\ell_n} \to \mathbb{F}_q^{\ell}.$$

According to this, given Σ with the $\texttt{Share}_\Sigma$ function, we can define the scheme Σ' on $\{1, \dots, \ell\}$ with $E'_i = \mathbb{F}_q$ for every i, satisfying that $[b]_{\Sigma'} = \varphi([b]_\Sigma) = (b_1, \dots, b_\ell)$ for every $b \in \mathbb{F}_q$, where each $b_i \in \mathbb{F}_q$.

As in the previous section, let $V' \subseteq \mathbb{F}_q^{\ell}$ denote the vector space consisting of all the sharings of 0 under the scheme Σ'. Given any $b \in \mathbb{F}_q$, let $W'_b \subseteq \mathbb{F}_q^{\ell}$ be the affine subspace of sharings of b for Σ'.

Given $k \in \mathbb{F}_q$, we instantiate the $\mathbb{F}_q$-LSSS $\mathcal{S}'_k$ associated to the affine subspace W'_k in Figure 6. The scheme $\mathcal{S}'_k$ is now defined on the set of ℓq participants $\mathcal{P}_{\ell,q}$ and it is $\mathbb{F}_q$-linear and ideal.

The Secret Sharing Scheme $\mathcal{S}'_k$

To share a message $m \in \mathbb{F}_q$, first

- let $\mathbf{k} = (k_1, \dots, k_\ell) \in \mathbb{F}_q^{\ell}$ be a sharing of k using Σ'
- sample $r_1, \dots, r_{\ell-1} \in \mathbb{F}_q$ uniformly at random, and let $r_\ell = m - \sum_{i=1}^{\ell-1} r_i$
- sample $\mathbf{h} = (h_1, \dots, h_\ell)$ uniformly at random from $(V')^{\perp}$

For every $i \in \mathcal{P}_\ell$ and for every $j \in \mathbb{F}_q$, define the (i, j)-th share as

$$m^{(i,j)} = r_i + (k_i - j)h_i.$$

Fig. 6. The $\mathbb{F}_q$-LSSS $\mathcal{S}'_k$ related to the affine subspace $W'_k \subseteq \mathbb{F}_q^{\ell}$.

As in the previous case, if $A \subseteq \mathcal{P}_{\ell,q}$ contains a set $A' \in \min \Gamma_{W'_k}$ of the form $A' = \{(1, b_1), \dots, (\ell, b_\ell)\}$, where $\mathbf{b} = (b_1, \dots, b_\ell) \in W'_k$, we can then define the function $\texttt{Reconstruct}_{\mathcal{S}'_k}$ on the shares $(m_k^{(i,j)})_{(i,j)\in A}$ of the message m_k as

$$\texttt{Reconstruct}_{\mathcal{S}'_k}\left((m_k^{(i,j)})_{(i,j)\in A}\right) = \sum_{i=1}^{\ell} m_k^{(i,b_i)}$$

To see that this function effectively retrieves m_k, note that

$$\sum_{i=1}^{\ell} m_k^{(i,b_i)} = \sum_{i=1}^{\ell} (r_i + (k_i - b_j)h_i) = \sum_{i=1}^{\ell} r_i + \langle \mathbf{k} - \mathbf{b}, \mathbf{h} \rangle = m_k$$

since $\sum_{i=1}^{\ell} r_i = m$, $\mathbf{k}, \mathbf{b} \in W'_k$ (so $\mathbf{k} - \mathbf{b} \in V'$) and $\mathbf{h} \in V'^{\perp}$.

As a direct consequence of Theorem 2 we have that, for every $k \in \mathbb{F}_q$, the secret sharing schemes $\mathcal{S}'_k$ are $\mathbb{F}_q$-linear, perfect, ideal and W'_k-OT-compatible.

C.2 Our One-out-of-q OT Combiner in the Non-Ideal Case

We now generalize the 1-out-of-q OT combiner presented previously to the case where Σ is not ideal. The obtained 1-out-of-q OT combiner is now ℓ-server (instead of n-server), and it is still single-use and black-box. We describe it in Figure 7.

Our 1-out-of-q OT Combiner Protocol π'_{OT}

$\pi'_{OT}.\mathsf{Choose}(b)$**:** Given $b \in \mathbb{F}_q$, compute a sharing $[b]_{\Sigma'} = (b_1, \dots, b_\ell)$ of b using Σ'.
Note that each $b_i \in \mathbb{F}_q$ because Σ' is ideal.
Output $(b_1, \dots, b_\ell)$.

$\pi'_{OT}.\mathsf{Send}(m_0, \dots, m_{q-1})$**:** For each message m_k, independently compute a sharing

$$[m_k]_{\mathcal{S}'_k} = (m_k^{(i,j)})_{(i,j)\in\mathcal{P}_{\ell,q}}.$$

Then, for every $(i,j) \in \mathcal{P}_{\ell,q}$, compute the values

$$u_i^j := m_0^{(i,j)} || m_1^{(i,j)} || \cdots || m_{q-1}^{(i,j)}.$$

Output $(u_i^j)_{(i,j)\in\mathcal{P}_{\ell,q}}$.

$\pi'_{OT}.\mathsf{Reconstruct}(b, (v_1, \dots, v_n))$**:** Parse each v_i as

$$v_i = n_0^{(i)} || n_1^{(i)} || \cdots || n_{q-1}^{(i)},$$

where $n_k^{(i)} \in \mathbb{F}_q$ for each $i \in \mathcal{P}_\ell$.
If $b = k$, retrieve m_b by evaluating

$$\mathtt{Reconstruct}_{\mathcal{S}'_k}((n_k^{(i)})_{i\in\mathcal{P}_\ell}).$$

If the reconstruction fails at any step, output $\mathbf{0}$.
Otherwise, output the reconstructed message m_b.

Fig. 7. Our 1-out-of-q OT combiner π'_{OT} for a general access structure $\mathcal{A}$.

When considering this extension there is, however, a subtlety to take into account. We originally assumed that we have n OT implementations at our disposal, and an $\mathcal{R}_2$ pair $(\mathcal{A}, \mathcal{B})$ of adversary structures representing the capabilities of malicious readers and receivers. Now, the adversary structure $\mathcal{A}'$ is a family of subsets of $\mathcal{P}_\ell$. Hence, in practice, some of the ℓ servers may correspond to the same OT primitive (for example, the first ℓ_1 servers if $\ell_1 \geq 2$). Given $A \in \mathcal{A}$, if a malicious sender corrupts one of such servers, all of the servers implementing the same OT candidate should also be considered as corrupted and be placed into A. And conversely, if one of the servers is not corrupted by the sender, none of them should be placed into A. The same observation applies for the sets $B \in \mathcal{B}$ of servers corrupted by a malicious receiver.

More formally, note that the set of servers is $\mathcal{P}_\ell$, which is in bijection with

$$P' = \{(i,j) \ : \ i \in \mathcal{P}_n,\, j = 1, \ldots, \ell_i\}.$$

Given $i \in \mathcal{P}_n$ denote $P'_i = \{(i,j) \ : \ j = 1, \ldots, \ell_i\}$, so we can express the disjoint union $P' = P'_1 \cup \ldots \cup P'_n$. As stated earlier, we may assume that we have n OT candidates at our disposal, and that the servers in P'_i implement the i-th OT candidate. Since they implement the same OT candidate, they are either corrupted or non-corrupted. To account for this, we can replace the adversary structures in our security and consistency definitions by

$$\mathcal{A}'' = \{\cup_{i \in A} P'_i \ : \ A \in \mathcal{A}\}, \quad \mathcal{B}'' = \{\cup_{i \in B} P'_i \ : \ B \in \mathcal{B}\}.$$

Note that, while the actual adversary structure $\mathcal{A}'$ of Σ' depends on the share spaces $E_1, \ldots, E_n$ of Σ, we know that $\mathcal{A}'' \subseteq \mathcal{A}'$. Therefore, this is consistent with the use of Σ'. Moreover, since $(\mathcal{A}, \mathcal{B})$ is an $\mathcal{R}_2$ pair, so is $(\mathcal{A}'', \mathcal{B}'')$.

The notions of correctness and of security introduced earlier, and all their proofs, translate mutatis mutandis to the non-ideal case by replacing n with ℓ, $\mathcal{A}$ and $\mathcal{B}$ with the adversary structures $\mathcal{A}''$ and $\mathcal{B}''$, V with V', and W_b with W'_b for every $b \in \mathbb{F}_q$.