Zero-Knowledge Systems from MPC-in-the-Head and Oblivious Transfer*

Cyprien Delpech de Saint Guilhem[1], Ehsan Ebrahimi[2], and Barry van Leeuwen[1]

[1] COSIC, KU Leuven, Leuven, Belgium,
firstname.lastname@kuleuven.be

[2] Department of Computer Science, University of Luxembourg, Luxembourg
eebrahimi.pqc@gmail.com

Abstract. Zero-knowledge proof or argument systems for generic NP statements (such as circuit satisfiability) have typically been instantiated with cryptographic commitment schemes; this implies that the security of the proof system (e.g., computational or statistical) depends on that of the chosen commitment scheme. The MPC-in-the-Head paradigm (Ishai et al., JoC 2009) uses the same approach to construct zero-knowledge systems from the simulated execution of secure multiparty computation protocols.

This paper presents a novel method to construct zero-knowledge protocols which takes advantage of the unique properties of MPC-in-the-Head and replaces commitments with an oblivious transfer protocol. The security of the new construction is proven in the Universal Composability framework of security and suitable choices of oblivious transfer protocols are discussed together with their implications on the security properties and computational efficiency of the zero-knowledge system.

Keywords: Zero-Knowledge · Oblivious Transfer · MPC-in-the-Head

1 Introduction

An *interactive proof system* [24] is a two-party protocol for an unbounded prover and a verifier with the goal of convincing the verifier that a certain statement is true. Such a proof system must fulfil two properties: (1) *completeness*, if the statement is true, an honest prover is able to convince the verifier; and (2) *soundness*, if the statement is not true, no (malicious) prover is able to convince the verifier. A relaxation of an interactive proof system is an *interactive argument system* in which the prover is computationally bounded [10].

The notion of *zero-knowledge* for a proof or argument system, introduced by Goldwasser, Micali and Rackoff [24], ensures that a malicious verifier interacting

* This preprint has not undergone any post-submission improvements or corrections. The Version of Record will appear in the proceedings of IMACC 2023 published by Springer.

with an honest prover is not able to learn any information beyond the veracity of the statement. Generally, such a construction allows for two inputs: the receiver holds a statement x belonging to some NP Language while the prover holds a witness w with the intent of proving some relation $\mathcal{R}$ about x and w.

Most of the existing zero-knowledge protocols are constructed from *commitment schemes*, relying on their hiding and binding properties. Furthermore, there is some evidence that such commitment schemes may be necessary to construct a zero-knowledge proof system [34]. In this work, we show that a zero-knowledge protocol can alternatively be constructed from an *oblivious transfer protocol*.

To obtain a zero-knowledge protocol using oblivious transfers we use the MPC-in-the-Head (MPCitH) paradigm [27]. In this framework, the prover simulates a secure n-party *multi-party computation (MPC) protocol* which *verifies* that w is a correct witnes for x. To do this the prover creates an additive sharing of its witness w, which means that the prover samples w_i, for $i \in [n]$, uniformly at random under the condition that $w = w_1 + w_2 + \ldots + w_n$. The execution of this protocol assumes n parties, P_i, with each party's private input defined as w_i. The result of the simulation of this protocol is n views $\{\mathsf{view}_i\}_{i\in[n]}$. The prover then commits to these views by sending them to the verifier. The verifier responds with some randomly chosen indices $I \subset [n]$ for which the prover opens the commitments $(\mathsf{view}_i)_{i\in I}$, thus demonstrating the correct verification of w by the MPC protocol.

We show, however, that the commitment scheme is unneccesary and one can obtain a zero-knowledge protocol in the MPCitH paradigm by using an oblivious transfer protocol instead. Instead of committing to n views $\{\mathsf{view}_i\}_{i\in[n]}$, the prover, in this OT-hybrid paradigm, engages in an oblivious transfer protocol which has inputs $\{\mathsf{view}_i\}_{i\in[n]}$ submitted by the prover and $I \subset [n]$ submitted by the verifier. At the end of the Oblivious Transfer protocol, the verifier has a subset of views, which it can then check for consistency. Below, we show how this gives us a zero-knowledge protocol in the *Universal Composability* framework [12].

1.1 Technical Overview

In Figure 3 we describe an MPC-in-the-Head protocol which realises the zero-knowledge proof functionality described in Figure 1 in the $\mathcal{F}_{\mathrm{OT}}$-hybrid model (see Figure 2). Due to the arbitrary number of parties that the verifier can choose to open (as long as it does not break the secrecy of the MPC protocol) we use an arbitrary k-out-of-n OT functionality.

We prove the UC-security of our protocol and show that its security holds in the $\mathcal{F}_{\mathrm{OT}}$-hybrid model. First, the completeneness of the proof follows from the *correctness* of the MPC protocol; if the latter is perfectly correct, then so is the resulting proof system, in the $\mathcal{F}_{\mathrm{OT}}$-hybrid model.

Secondly, the soundness of the proof system holds unconditionally in the $\mathcal{F}_{\mathrm{OT}}$-hybrid model since a malicious prover is caught whenever its cheating behaviour is observed in the MPC protocol by the verifier; here, the *robustness* of the MPC protocol matters, since a robust MPC protocol will still output a correct rejection of an invalid witness despite a certain number of cheating parties. The property we prove is in fact *knowledge soundess* since the definition of the ZK functionality requires a valid witness to be provided in order to inform the verifier of a valid proof. The UC simulator of our security proof is therefore able to extract a valid witness (with some soundness error) in cases where a malicious prover is able to make an honest verifier accept.

Finally, the zero-knowledge property of the proof system follows from the *privacy* property of the MPC protocol which guarantees that no information is learnt about a secret-shared witness when too few shares are known. Since the OT functionality guarantees that exactly k views out of a possible n will be opened, even for malicious verifiers, the k-privacy of the MPC protocol guarantees malicious-verifier zero-knowledge for the proof system.

When instantiating our protocol with a specific oblivious transfer protocol to realise $\mathcal{F}_{\mathrm{OT}}$, the security type (perfect, statistical or computational) of the OT protocol must then also be taken into account to establish the final security guarantees of the proof (or argument) system.

To this effect, in Section 4 we list several OT protocols that could be suitable to instantiate our protocol. Given that generic k-out-of-n OT protocols are more difficult to come by in practice, we discuss several options to use simpler 1-out-of-n and even 1-out-of-2 OT protocols based on existing efficient MPCitH protocols from the literature.

1.2 Comparison and Theoretical Value

We discuss how our work differs from the existing zero-knowledge constructions and how it contributes to the theoretical research regarding the round complexity of zero-knowledge protocols.

1. Given that the *rewinding* proof technique is troublesome in the quantum setting [2], our work benefits from *straight-line* extraction, especially since using rewinding of the adversary to prove UC-security of OT protocols is also not allowed. This would be beneficial to construct post-quantum zero-knowledge protocols.
2. The round complexity of a zero-knowledge protocol has been a topic of research since the introduction of the zero-knowledge notion in 1989 (see Appendix A for a brief survey). However, the round complexity of post-quantum zero-knowledge protocols is a recent research direction and it is not as developed as in the classical case.[3] We emphasize that our approach in this paper would be valuable to construct a constant-round post-quantum zero-knowledge protocol since the round complexity of our protocol depends on the (post-quantum) implementation of the OT functionality and it benefits from a *straight-line* extraction.
3. We prove the zero-knowledge in the *Universal Composability* framework [12].

[3] Even with some (apparently) contradictory results: the impossibility [13] and the possibility [32] of constructing constant-round post-quantum black-box zero-knowledge.

2 Preliminaries

This section introduces notations and recalls standard definitions.

2.1 Notation

We denote by λ the security parameter. For elements $n \in \mathbb{Z}$ we denote by $[n]$ the set of integers $\{1, \ldots, n\}$. We say that a function $f : \mathbb{N} \to \mathbb{N}$ is negligible if, for every positive polynomial $p(\cdot)$ and all sufficiently large integers k it holds that $f(k) < \frac{1}{p(k)}$. We abbreviate a probabilistic polynomial time machine by PPT.

For any element $a \in \mathbb{K}$, we will denote a random sampling of a from a distribution D_α as $a \leftarrow D_\alpha$. Furthermore, we shall denote by U_α the uniform distribution with variance α. If an element a is drawn uniformly random from a set, or according to a protocol, A, where the distribution used to sample from A is known, we may abbreviate by writing $a \leftarrow D_A$.

2.2 Zero-Knowledge Proof and Argument Systems

For an NP language $\mathcal{L}$, we denote by $\mathcal{R}$ the relation consisting of pairs (x, w) such that x is an instance in $\mathcal{L}$ and w is a corresponding candidate witness. In an interactive proof or argument protocol, a prover wishes to demonstrate that some NP statement $x \in \mathcal{L}$ is true using a valid witness w such that $(x, w) \in \mathcal{R}$.

The proof or argument protocol is *correct* if an honest prover always successfully convinces an honest verifier of the veracity of a true statement. The protocol is *sound* if a malicious prover cannot convince an honest verifier that a false statement $x^* \notin \mathcal{L}$ is in fact true; it is additionally *knowledge sound* if a malicious prover cannot convince a verifier even of a true statement $x \in \mathcal{L}$ without knowing at least one valid witness w such that $(x, w) \in \mathcal{R}$.

For both notions of soundness, it is tolerated that a malicious prover can successfully convince an honest verifier with a negligible probability called the *soundness error*. If this error is negligible even for computationally unbounded malicious provers, then the protocol is called a *proof system*; if the soundness error is negligible only for PPT malicious provers, then the protocol is called an *argument* system.

A proof or argument system can also be *zero-knowledge* (ZK) if the interaction of an honest prover with a verifier reveals no information about the witness w other than its validity. This property can hold against either honest verifiers or fully malicious ones.

UC-secure ZK systems for circuit satisfiability. This work focuses on UC-secure protocols for proving in zero-knowledge the satisfiability of an arbitrary circuit C. We assume that the reader is familiar with the terminology of UC security and proofs [12]. In Figure 1 we recall the zero-knowledge functionality [16].

In this figure, C is a circuit, with format depending on ty, such that for a given x, $C_x(w) = 1 \Leftrightarrow \mathcal{R}(x, w) = 1$. At the end of the protocol, the verifier then accepts or rejects the proof. We denote by $(\mathcal{P}(x, w), \mathcal{V}(x)) = b$, $b \in \mathbb{F}_2$,

The ZK Functionality: $\mathcal{F}_{\mathsf{ZK}}$

The functionality runs with a prover $\mathcal{P}$, a verifier $\mathcal{V}$, and an adversary $\mathcal{A}$.
It is parameterized by $\mathsf{ty} = \{\mathsf{Boolean}, \mathsf{Arithmetic}\}$

Proof: On input of $(\mathsf{sid}, \mathsf{prove}, \mathcal{P}, \mathcal{V}, C, \mathsf{ty}, w)$ from the prover, compute $y = C(w) \in \{0, 1\}$, send (prove, C) to $\mathcal{V}$ and $\mathcal{A}$, and store (C, y).
Verify: On receiving $(\mathsf{sid}, \mathsf{verify}, C, \mathsf{ty})$ from $\mathcal{V}$, query $\mathcal{A}$; If $\mathcal{A}$ returns fail, or if (C, y) has not been stored, send $(\mathsf{sid}, C, 0)$ to $\mathcal{V}$. Otherwise, send (sid, C, y) to $\mathcal{V}$.

Fig. 1. Ideal functionality for circuit-based ZK proofs

the verifier's decision such that $b = 1$ means the verifier accepts and otherwise rejects.

While not explicitly defined in the functionality, the knowledge soundness and zero-knowledge properties of a protocol that securely UC-realizes $\mathcal{F}_{\mathrm{ZK}}$ follow from the different proof cases. Namely, knowledge soundness follows from security against a malicious prover: the UC simulator must input (C, w) to $\mathcal{F}_{\mathrm{ZK}}$ acting as the ideal-world malicious prover such that $\mathcal{F}_{\mathrm{ZK}}$ then induces the ideal-world verifier to accept or reject the proof with the same distribution as the real-world verifier. The simulator must then extract the witness (valid or not) from the real-world malicious prover and this simulation will fail (i.e., the ideal-world verifier will reject a false statement when the real-world verified will incorrectly accept it) exactly with the knowledge soundness error of the protocol.

Similarly, zero-knowledge follows from the security against a dishonest verifier: the UC simulator must produce a protocol transcript, without knowledge of the witness (only of its validity), that cannot be distinguished as a simulation.

2.3 Oblivious Transfer Protocols

Oblivious Transfer (OT) is a well known primitive within cryptography, which has been extensively researched since its introduction by Rabin [36]. In an OT Protocol a sender, $\mathcal{S}$, and a receiver, $\mathcal{R}$, execute the transfer of a subset of messages, $m = \{m_0, \ldots, m_{k-1}\}$, out of a total set of n messages. Depending on the protocol these messages could be bits or strings. Generally OT protocols are divided, broadly, into three different categories depending on k and n: $(k, n) = (1, 2)$, $(k, n) = (1, n)$, and $k, n \in \mathbb{N}, k < n$. For an Oblivious Transfer protocol to be secure the following two properties have to be obtained:

- Sender Security: Upon committing to n messages, $m_1, \ldots, m_n$, the sender is assured that $\mathcal{R}$ receives no more than k messages and will only learn the contents of these k messages.
- Receiver Security: Upon committing to the k indices I, $I \subset [n]$, the receiver is assured that $\mathcal{S}$ does not learn which messages the receiver has learnt.

In [14] they define a 1-out-of-n OT protocol which is easily adapted to the k-out-of-n variant. We describe this adapted variant in Figure 2. As you can see this is exactly what we would expect from an OT protocol.

The k-out-of-n OT functionality $\mathcal{F}_{\mathrm{OT}}$

The functionality runs with a sender $\mathcal{S}$, a receiver $\mathcal{R}$, and an adversary $\mathcal{A}$.
Commit: On input of $(\mathsf{sid}, \mathsf{commit}, \mathcal{S}, \mathcal{R}, (c_1, \ldots, c_n))$ by $\mathcal{S}$, store $(\mathsf{sid}, c_1, \ldots, c_n)$ and send $(\mathsf{sid}, \mathsf{ready})$ to $\mathcal{A}$
Abort: On receipt of $(\mathsf{sid}, \mathsf{ready})$ $\mathcal{A}$ sends (sid, b) to $\mathcal{F}_{\mathrm{OT}}$ where $b \in \{0, 1\}$. If $b = 1$ abort the functionality, else proceed.
Query: On input of $(\mathsf{sid}, \mathsf{query}, \mathcal{S}, \mathcal{R}, I)$ by $\mathcal{R}$, check if for every $i, j \in I : i \neq j$ and $|I| = k$, moreover check if $(c_1, \ldots, c_n)$ are stored. Then do one of the following:

- If I does not fulfill the requirements return $\perp$ to $\mathcal{R}$.
- If no values are stored, do nothing.
- Else send $(\{c_i\}_{i \in I})$ to $\mathcal{R}$

Fig. 2. Ideal functionality for k-out-of-n OT.

2.4 MPC

In this paper the standard definitions of MPC from the literature will be followed, [11,20,27]. To this extent let n be the number of parties and let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be the set of identified parties. A public input x is known to all parties, while each party individually supplies their private input w_i. To securely realize an n-party functionality f, f takes as input $(x, w_1, \ldots, w_n)$ and produces n outputs. Any protocol, $\mathit{\Pi}$, takes as input the party that wishes to execute the protocol, P_i, their private input, w_i, their random input, r_i, and the public parameter x, and possibly a security parameter k in the case of statistical or computational security. Moreover, for the protocol called in round $j + 1$, the protocol will additionally require the messages that P_i received in the previous j rounds. The protocol will then output n messages, and, if required, a broadcast message. Specifically, if the broadcast message of $\mathit{\Pi}$ is abort then the protocol terminates immediately, only outputting P_i's local output. Throughout the execution of a protocol the view of a player P_i, denoted view_i, is constructed. This view includes w_i,r_i, and the messages that P_i received during the execution of $\mathit{\Pi}$. The following definition follows naturally:

Definition 1. *Let* view_i *and* view_j *be produced by protocol* $\mathit{\Pi}$ *with respect to some public input* x*. Then two views can be called consistent if the outgoing messages implicit in* view_i *are identical to the incoming messages reported in* view_j *and vice versa.*

Note that this is a natural definition as we can take view_i, $\mathit{\Pi}$, and x and reconstruct the local output for P_i and the messages sent. In [27] it is shown that there is no difference between consistency of the views from a global perspective and a local perspective.

Lemma 1 (Lemma 2.3, [27]). *Let* $\mathit{\Pi}$ *be an* n*-party protocol with public input* x*. Let* $\{\mathsf{view}_1, \ldots, \mathsf{view}_n\}$ *be the set of (not necessarily correct) views. Then for any* $i, j \in [n]$*, it holds that* view_i *and* view_j *are consistent with respect to* $\mathit{\Pi}$ *and*

x if and only if there exists and honest execution of Π with public input x in which view_i is the view of P_i for every $i \in [n]$.

For our MPC constructions we will consider both the semi-honest and the malicious models. For the semi-honest model, also known as "Honest, but curious", the parties will execute a protocol Π as is prescribed, however the parties will attempt to learn more information from the protocol than is intended to. In the malicious model such restrictions are lifted and the parties are allowed to act arbitrarily in regards to the protocols and each other.

In the semi-honest case security can be broken into the following two properties:

Definition 2 (Correctness (Definition 2.4 [27])). *We say that Π realize a deterministic n-party functionality $f(x, w_1, \dots, w_n)$ with* perfect (resp., statistical) correctness *if for all inputs $x, w_1, \dots, w_n$ the probability that the output of some player is different from the output of f is* 0 *(resp., negligible in λ), where the probability is over the independent choices of the random inputs $r_1, \dots, r_n$.*

Definition 3 (t-Privacy (Definition 2.5 [27])). *Let $1 \leq t < n$. We say that Π realizes f with* perfect t-privacy *if there is a PPT simulator* Sim *such that for any inputs $x, w_1, \dots, w_n$ and every set of corrupted players $T \subseteq [n]$, where $|T| \leq t$, the joint view $\mathsf{View}_T(x, w_1, \dots, w_n)$ of players in T is distributed identically to $\mathsf{Sim}(T, x, (w_i)_{i \in T}, f_T(x, w_1, \dots, w_n))$, where $f_T(\cdot)$ denotes the view of the output of f of the parties in T.*

For relaxations to statistical (resp., computational) t-privacy*, we require that for every distinguisher D (resp., D with circuit size $\mathsf{poly}(\lambda)$), there is a negligible function $\delta(\cdot)$ such that*

$$\begin{aligned} &|\Pr[D(\mathsf{View}_T(\lambda, x, w_1, \dots, w_n)) = 1] \\ &\quad - \Pr[D(\mathsf{Sim}(\lambda, T, x, (w_i)_{i \in T}, f_T(x, w_1, \dots, w_n))) = 1]| \leq \delta(\lambda) \end{aligned}$$

For the malicious model, however, correctness is not sufficient. Instead we adopt notion that Π is secure if and only if the protocol is t-private, as defined above, and r-robust.

Definition 4 (r-Robustness (Definition 2.6 [27])). *We say that Π realizes f with* perfect (resp., statistical) r-robustness *if it is perfectly (resp,. statistically) correct in the presence of a semi-honest adversary as in Definition 2, and furthermore for any computationally unbounded malicious adversary corrupting a set R of at most r players, and for any inputs $(x, w_1, \dots, w_n)$, the following robustness property holds. If there is no $(w'_1, \dots, w'_n)$ such that $f(x, w'_1, \dots, w'_n) = 1$, then the probability that some uncorrupted players outputs* 1 *in an execution of Π in which the inputs of the honest players are consistent with $(x, w_1, \dots, w_n)$ is 0 (resp., negligible in λ).*

The MPCitH-based protocol: $\mathit{\Pi}_{\mathsf{ZK}}$

This protocol is parametrized by an n-party MPC protocol $\mathit{\Pi}_{\text{MPC}}$ which computes an arithmetic or Boolean circuit C and is t-private and s-robust.

Prove: On input of $(\mathsf{sid}, \mathsf{prove}, \mathcal{P}, \mathcal{V}, C, w)$,

1. $\mathcal{P}$ secret-shares $\langle w \rangle$ such that $\sum_{i=1}^{n} w_i = w$;
2. $\mathcal{P}$ generates the random coins r_i for each player P_i, for $i \in [n]$;
3. $\mathcal{P}$ simulates an execution of $\mathit{\Pi}_{\text{MPC}}$ between $P_1, \ldots, P_n$ and records $\mathsf{view}_i = (w_i, r_i, \{m_j^i\})$ where $\{m_j^i\}$ is the set of messages received by P_i during the execution of $\mathit{\Pi}_{\text{MPC}}$;
4. $\mathcal{P}$ sends $(\mathsf{sid}, \mathsf{send}, \mathcal{S}, \mathcal{R}, (\mathsf{view}_1, \ldots, \mathsf{view}_n))$ to $\mathcal{F}_{\text{OT}}$.

Verify: On input of $(\mathsf{sid}, \mathsf{verify}, \mathcal{P}, \mathcal{V}, C)$,

1. $\mathcal{V}$ samples at random a subset $I \subset [n]$ of size t;
2. $\mathcal{V}$ sends $(\mathsf{sid}, \mathsf{receive}, \mathcal{S}, \mathcal{R}, I)$ to $\mathcal{F}_{\text{OT}}$ and obtains $\{\mathsf{view}_i\}_{i \in I}$;
3. $\mathcal{V}$ outputs $(\mathsf{sid}, C, 1)$ if (1) the messages contained in and implied by the received views are consistent with each other, and (2) the opened views are consistent with the MPC protocol outputting $C(\langle w \rangle) = 1$ to each party; otherwise $\mathcal{V}$ outputs $(\mathsf{sid}, C, 0)$.

Fig. 3. MPC-in-the-head ZK protocol in the $\mathcal{F}_{\text{OT}}$-hybrid model.

3 Zero-Knowledge from MPCitH and Oblivious Transfer

It were these definitions that led to MPC-in-the-Head (MPCitH) paradigm, as introduced in [27], where any honest-majority MPC protocol, i.e. $t < \frac{n}{2}$ corruptions, can be used to obtain a zero-knowledge proof for an arbitrary relation $\mathcal{R}$.

The idea is as follows: Let $\mathcal{P}$ be the prover and let $\mathcal{V}$ be the verifier. Given a public parameter x, $\mathcal{P}$ submits a witness w, which upon compution of $\mathcal{R}(x, w)$ shows that x belongs to a language $\mathcal{L}$ or not, specifically: $\mathcal{R}(x, w) \in \mathbb{F}_2$ such that, for a valid witness, if $x \in \mathcal{L}$, $\mathcal{R}(x, w) = 1$, otherwise $\mathcal{R}(x, w) = 0$.

Now assume that $\mathcal{P}$ generates a sharing, $\langle w \rangle = (w_1, \ldots, w_n)$, and computes $\mathcal{R}(x, \langle w \rangle)$ by choosing random coins r_i uniformly at random. By regarding each pair (w_i, r_i) as parties in an n-party MPC protocol, as described in Section 2.4, we then obtain a set of views, view_i, corresponding to the output of the MPC protocol.

Having obtained the views, the $\mathcal{P}$ submits the views to an oracle $\mathcal{O}$, which the verifier, $\mathcal{V}$, then queries a set of indices, I, to obtain $\{\mathsf{view}_i\}_{i \in I}$. By Lemma 1 we then obtain that the verifier can conclude if the computation was done correctly by checking that the opened views are all consistent with each other and that the protocol outputs a positive result.

One way to realize such an oracle is by implementing an oblivious transfer protocol. Figure 3 presents an MPCitH-based ZK proof system in the $\mathcal{F}_{\text{OT}}$-hybrid model.

This protocol, $\mathit{\Pi}_{\mathrm{ZK}}$, proceeds exactly as described when instantiated with $\mathcal{F}_{\mathrm{OT}}$. In the $\mathcal{F}_{\mathrm{OT}}$-hybrid model, $\mathit{\Pi}_{\mathrm{ZK}}$ can be shown to UC-securely realize $\mathcal{F}_{\mathrm{ZK}}$.

Theorem 1. *Let $\mathit{\Pi}_{\mathrm{MPC}}$ be an n-party protocol with perfect correctness, t-privacy and perfect r-robustness, with $t = \Omega(\lambda)$ and $n = c \cdot t$ for some constant $c > 1$. $\mathit{\Pi}_{\mathrm{ZK}}$ of Figure 3 UC-realises $\mathcal{F}_{\mathrm{ZK}}$ of Figure 1 with soundness error $\epsilon = \max\{p_1(n,t,r), p_2(n,t,r)\}$, where*

$$p_1(n,t,r) = \binom{r}{t}\binom{n}{t}^{-1}, \quad \textit{and}$$

$$p_2(n,t,r) = \begin{cases} 0 & \textit{otherwise} \\ \left(\sum_{j=0}^{k}\binom{k}{j}\binom{n-2k}{t-j}\right)\binom{n}{t}^{-1} & \textit{if } n-2k > 0 \end{cases}$$

and $k = \lfloor r/2 \rfloor + 1$.

Proof. We design a simulator Sim to act as adversary in the ideal-world execution. We consider in turn the four cases of the real-world where: both parties are honest, only the verifier is honest, only the prover is honest, and both parties are corrupt.

1. Both parties are honest: Upon receiving the query from $\mathcal{F}_{\mathrm{ZK}}$, the simulator Sim sends $(\mathsf{sid}, \mathsf{ready})$ to $\mathcal{A}$ on behalf of $\mathcal{F}_{\mathrm{OT}}$. If $\mathcal{A}$ responds with abort, then Sim responds abort to $\mathcal{F}_{\mathrm{ZK}}$, otherwise it responds with $\mathsf{continue}$.

2. Only the prover is corrupt: Upon receiving $\mathsf{view}^* = (\mathsf{view}_1^*, \ldots, \mathsf{view}_n^*)$ from the corrupt prover $\mathcal{P}^*$, the simulator reconstructs a witness $w^* = w_1^* + \cdots + w_n^*$ and sends $(\mathsf{sid}, \mathsf{prove}, \mathcal{P}, \mathcal{V}, C, w^*)$ to $\mathcal{F}_{\mathrm{ZK}}$. It also sends $(\mathsf{sid}, \mathsf{ready})$ to $\mathcal{A}$.

When $\mathcal{F}_{\mathrm{ZK}}$ queries Sim, the simulator first checks $\mathcal{A}$'s response. If $\mathcal{A}$ replied $(\mathsf{sid}, \mathsf{abort})$ to $\mathcal{F}_{\mathrm{OT}}$, then Sim also sends abort to $\mathcal{F}_{\mathrm{ZK}}$. Otherwise, Sim responds with $\mathsf{continue}$.

3. Only the verifier is corrupt: Upon receiving (prove, C) from $\mathcal{F}_{\mathrm{ZK}}$, the simulator sends $(\mathsf{sid}, \mathsf{ready})$ to $\mathcal{A}$ on behalf of $\mathcal{F}_{\mathrm{OT}}$. When $\mathcal{A}$ sends $(\mathsf{sid}, \mathsf{receive}, \mathcal{S}, \mathcal{R}, I)$ to $\mathcal{F}_{\mathrm{OT}}$, Sim sends $(\mathsf{sid}, \mathsf{verify}, C)$ to $\mathcal{F}_{\mathrm{ZK}}$. If $\mathcal{A}$ responded with $(\mathsf{sid}, \mathsf{abort})$ to $\mathcal{F}_{\mathrm{OT}}$, then Sim responds abort to $\mathcal{F}_{\mathrm{ZK}}$ when queried, otherwise it responds $\mathsf{continue}$, and receives (sid, C, y).

The simulator invokes the t-privacy simulator $\mathsf{Sim}_{\mathsf{MPC}}$ of the MPC protocol on corruption set I by sampling $\{w_i\}_{i\in I}$ uniformly at random as in the protocol and inputting $(I, x, \{w_i\}_{i\in I}, y)$. From $\mathsf{Sim}_{\mathsf{MPC}}$ it then receives a set of consistent views $\{\mathsf{view}_i\}_{i\in I}$ which will agree with the required outcome, y. Finally, Sim sends these views to $\mathcal{A}$ as the response from $\mathcal{F}_{\mathrm{OT}}$.

4. Both parties are corrupt: Just like in case 2, the corrupt prover, $\mathcal{P}^*$, submits $\mathsf{view}^* = (\mathsf{view}_1^*, \ldots, \mathsf{view}_n^*)$ to Sim. Upon receiving these views Sim sends $\mathsf{sid}, \mathsf{ready}$ to $\mathcal{A}$ and processes any abort instructions coming from $\mathcal{A}$ if necessary. Unless it receives an abort instruction from $\mathcal{A}$, like in case 3, the corrupt verifier, $\mathcal{V}^*$, submits a set I to Sim. Sim then sends $(\mathsf{view}_i)_{i\in I}$ to $\mathcal{V}^*$. No further simulation is necessary as both the prover and the verifier are corrupt.

Completeness [27, proof of Theorem 3.1]: If $(x, w) \in R$ and the prover is honest, then, since $\sum_{i=1}^{n} w_i = w$ and Π_{MPC} is perfectly correct, the views $\mathsf{view}_1, \ldots, \mathsf{view}_n$ always have output 1. Since these views are honestly produced, they are always consistent with each other.

Soundness: Note that in the real world, the prover uses an MPC-in-the-Head protocol to produce a set of n views, $(\mathsf{view}_1, \ldots, \mathsf{view}_n)$, from which the verifier then gets to select a t-sized set of views to open. However, in the ideal world, the prover submits the n views to Sim who then extracts the witnesses $w_1^*, \ldots, w_n^*$ and recombines them to obtain $w^* = w_1^* + \ldots + w_n^*$. Sim then sends the recombined witness to $\mathcal{F}_{\mathrm{ZK}}$. $\mathcal{F}_{\mathrm{ZK}}$ then evaluates if the witness received is correct and returns the outcome, abort or accept, to the verifier. Clearly there is a discrepancy here between the real world and the ideal world if, and only if, the t sized set of views opened to the verifier is consistent while there are views in the remaining $(n - t)$ views that would cause an inconsistency; this would cause the ideal world verifier to abort while the real world verifier would accept. Since we are opening t views of a t-private and perfectly r-robust MPC protocol, the soundness analysis follows exactly that of the protocol of Ishai et al. for MPC-in-the-head with MPC in the malicious model [27, Theorem 4.1]. Here we make use of the explicit probability formulae given by Giacomelli et al. [19] following the analysis of Ishai et al. We therefore have that the soundness error is equal to the value $\epsilon(n, t, r) = \max\{p_1(n, t, r), p_2(n, t, r)\}$, where

$$p_1(n, t, r) = \binom{r}{t}\binom{n}{t}^{-1}, \quad \text{and}$$

$$p_2(n, t, r) = \begin{cases} 0 & \text{otherwise} \\ \left(\sum_{j=0}^{k} \binom{k}{j}\binom{n-2k}{t-j}\right)\binom{n}{t}^{-1} & \text{if } n - 2k > 0 \end{cases},$$

where $k = \lfloor r/2 \rfloor + 1$. Here p_1 illustrates the case in which $\mathcal{A}$ has corrupted a set of views which do not pass the robustness threshold and therefore $t \leq r$. This means that the soundness error, which is the probability that the ideal world aborts while the real world accepts, is dictated by the probability that a set is chosen in which the t views are consistent while there is an inconsistency within the remaining $r - t$ views out of all the possible size t sets. Similarly, p_2 illustrates the case in which $\mathcal{A}$ manages to corrupt a set that breaks the r-robustness of the protocol. Note that in this case r-robustness can not be broken if $2k \geq n$. This concludes that the soundness error can be described as

$$|\Pr[\mathsf{Exec}_{Z,\Pi,\mathcal{P}^*} = 1] - \Pr[\mathsf{Exec}_{Z,\mathcal{F},\mathsf{Sim}_{\mathcal{P}^*}} = 1]| = \epsilon(n, t, r)$$

Zero-knowledge: If Π_{MPC} is *perfectly* t-private, then the simulation returned by $\mathsf{Sim}_{\mathsf{MPC}}$ is case 3 is distributed identically to an honest execution of the protocol. Similarly, if Π_{MPC} is *statistically or computationally* t-private, then the distribution of the views returned by $\mathsf{Sim}_{\mathsf{MPC}}$ is statistically or computationally close to that of the views produced by an honest prover.

$$|\Pr[\mathsf{Exec}_{Z,\Pi,\mathcal{A}} = 1] - \Pr[\mathsf{Exec}_{Z,\mathcal{F},\mathsf{Sim}_{\mathcal{V}^*}} = 1]| = |D_{\Pi_{\mathsf{MPC}}} - D_{\mathsf{Sim}_{\mathsf{MPC}}}|$$

Reference	Format	Rounds	UC Secure	Security Level	Post Quantum	OT type
[4]	1/2	2	ROM	Statistical	Multiple	String
[35]	1/2	2	CRS	Statistical	LWE	Bit
[31]	1/2	4	ROM	Computational	Isogenies	String
[17]	1/2	2	CRS	Statistical	LPN	String
[3] Protocol-1	1/2	2	ROM	Computational	Isogenies	String
[3] Protocol-2	1/2	4	Standard	Computational	Isogenies	String
[33]	$1/n$	5	Standard	Statistical	NTRU	String
[26]	$1/n$	2	CRS	Computational	×	String
[9]	$1/n$	3	ROM	Statistical	LWE	String
[25]	k/n	3	CRS	Computational	×	String

Table 1. Non-exhaustive list of UC-secure OT protocols

□

4 Suitable Oblivious Transfer Protocols

The characteristics of the MPCitH proof system in the OT-hybrid model that we propose in Section 3 are strongly tied to those of the chosen OT protocol. Namely, the proof system will have as many rounds as the OT protocol does, will be secure against either unbounded or computationally-bounded[4] provers depending on the OT protocol's security against malicious senders, will be honest-verifier zero-knowledge if the OT protocol is only secure against passive malicious receivers, and so on. In this section, we therefore discuss the suitability of a non-exhaustive list of UC-secure OT protocols from the literature, summarized in Table 1, to instantiate the OT functionality used by our protocol.

While the MPCitH proof system from Section 3 uses an arbitrary k-out-of-n OT protocol, in practice the values for k and n are fixed by the choice of the MPC protocol. As can be seen from Table 1, in fact k-out-of-n OT protocols are the least common in the literature as they are often not the initial goal of OT protocol designers.

4.1 Generic MPCitH and 1-out-of-2 Oblivious Transfer

The initial proposal for MPCitH by Ishai et al. [27] can straightforwardly be instantiated with a 2-party MPC protocol, implying $t = 1$; this enables the use of 1-out-of-2 OT to realize $\mathcal{F}_{\mathrm{OT}}$. This is advantageous because this is the type of OT that is most often first constructed, and is the most present in the post-quantum OT literature (see Table 1).

This type of OT also tends to be the most efficient, with several constructions requiring only two rounds of communication; this yields a two-round MPCitH

[4] In this case our protocol would formally be an MPCitH argument system.

zero-knowledge argument system, since the security against a malicious OT sender holds with computational assumptions.

However, a drawback of this approach is that each execution of the MPCitH protocol has a soundness error of 1/2 because a new sharing of the witness is created each time. To achieve soundness errors of $O(2^{-\lambda})$ therefore requires $O(\lambda)$ independent repetitions of the MPC protocol, which is computationally expensive for the prover.

Furthermore, two-round OT protocols that are simulation-secure (let alone UC-secure) are impossible in the plain model [23] which therefore implies that any zero-knowledge proof systems based on efficient two-round oblivious transfer must necessarily rely on setup assumptions such as the random oracle model or a common reference string.

4.2 Broadcast MPCitH and 1-out-of-n Oblivious Transfer

A drawback of the previous instantiation is that creating independent 2-party secret sharings of the witness leads to computational inefficiency for the prover, since it has to simulate $O(\lambda)$ repetitions of the MPC protocol; this is also not efficient for the communication efficiency of the proof system, since each repetition of the MPC protocol needs to open the view of one party to the verifier.

To reduce the number of repetitions, and thus the amount of communication that is sent, it can be interesting to increase the value of n, and also vary the value of t. However, as Table 1 shows, t-out-of-n UC-secure OT protocols are not common—the only one we found is furthermore not post-quantum secure. Therefore it is more interesting to look at specific values for t.

When the MPC protocol used for the MPCitH construction is $(n-1)$-private, we say that is it "full-threshold" to mean that the threshold of tolerated privacy corruptions is as high as it can possibly be. In this setting, all but one of the MPC parties' views can be opened to the verifier which means that $\mathcal{F}_{\mathrm{OT}}$ of Figure 1 can be realized by an $(n-1)$-out-of-n OT protocol, also known as "all-but-one OT". However, efficient constructions for this type of OT have only recently been proposed [5] and their design space is not as well understood. Independently of the chosen OT protocol, such an instantiation would still require opening $n-1$ views for each repetition of the MPC protocol, which would not improve the communication efficiency.

This can be remedied by choosing an MPC protocol which exclusively uses a broadcast communication channel; that is, whenever a party sends a message, it is received identically by all other parties in the protocol. With such a communication model, much less data needs to be included in the views of each MPC party since all incomming messages from party P_i are identical for all other parties, and equal to all outgoing messages of party P_i.

Therefore, when combined with a full-threshold MPC protocol (see Section 4.2), when all parties except for P_i are requested by the MPCitH verifier, the prover needs to send only the list of outgoing messages of P_i, rather than the $n-1$ lists of incomming messages for the other parties.

For the $\mathcal{F}_{\mathrm{OT}}$-hybrid version that we propose in Section 3, this implies that $\mathcal{F}_{\mathrm{OT}}$ can be realized with 1-out-of-n OT protocols for the part of the views that contain the MPC protocol messages; since that is usually the biggest part of the view, this results in a factor n reduction in the amount of communication. Furthermore, 1-out-of-n OT protocols are more commonly built than $(n-1)$-out-of-n ones (see Table 1) which gives more choices for the instantiation.

4.3 Hypercube MPCitH and 1-out-of-2 Oblivious Transfer

While most recent MPCitH constructions are based on broadcast MPC [6,15], the computational cost of $(n-1)$-out-of-n and 1-out-of-n OT protocols required to instantiate our construction may be too high, and reverse the advantages gained from the use of broadcast-based MPC protocols.

The recent technique of "Hypercube MPCitH" [1] can enable the return to 2-party MPC by secret-sharing a high number of MPC parties, say $n = N^d = 32$, into $d = 5$ parallel executions of $N = 2$-party MPC computations which use a single N^d-sharing of the witness.

The advantage of this technique is to reduce the opening of the message part of the views of the MPC protocol to 1-out-of-2 OTs, instead of 1-out-of-n, while grouping the opening of $n-1$ witness share parts of the views into a single $(n-1)$-out-of-n OT.

Acknowledgments Cyprien Delpech de Saint Guilhem is a Junior FWO Postdoctoral Fellow under project 1266123N. Ehsan Ebrahimi is supported by the Luxembourg National Research Fund under the Junior CORE project QSP (C22/IS/17272217/QSP/Ebrahimi). This work has been supported in part by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001120C0085, by CyberSecurity Research Flanders with reference number VR20192203 and by the FWO under Odysseus project GOH9718N. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA, Cyber Security Research Flanders, or the FWO.

References

1. Aguilar Melchor, C., Gama, N., Howe, J., Hülsing, A., Joseph, D., Yue, D.: The return of the SDitH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 564–596. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_20
2. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems: The hardness of quantum rewinding. In: 55th FOCS. pp. 474–483. IEEE Computer Society Press (Oct 2014). https://doi.org/10.1109/FOCS.2014.57
3. Badrinarayanan, S., Masny, D., Mukherjee, P., Patranabis, S., Raghuraman, S., Sarkar, P.: Round-optimal oblivious transfer and MPC from computational CSIDH. In: Boldyreva, A., Kolesnikov, V. (eds.) PKC 2023, Part I. LNCS, vol. 13940, pp. 376–405. Springer, Heidelberg (May 2023). https://doi.org/10.1007/978-3-031-31368-4_14

4. Barreto, P.S.L.M., David, B., Dowsley, R., Morozov, K., Nascimento, A.C.A.: A framework for efficient adaptively secure composable oblivious transfer in the ROM. Cryptology ePrint Archive, Report 2017/993 (2017), https://eprint.iacr.org/2017/993
5. Baum, C., Braun, L., Delpech de Saint Guilhem, C., Klooß, M., Orsini, E., Roy, L., Scholl, P.: Publicly verifiable zero-knowledge and post-quantum signatures from vole-in-the-head. In: Handschuh, H., Lysyanskaya, A. (eds.) Advances in Cryptology – CRYPTO 2023. pp. 581–615. Springer Nature Switzerland, Cham (2023). https://doi.org/10.1007/978-3-031-38554-4_19
6. Baum, C., Delpech de Saint Guilhem, C., Kales, D., Orsini, E., Scholl, P., Zaverucha, G.: Banquet: Short and fast signatures from AES. In: Garay, J. (ed.) PKC 2021, Part I. LNCS, vol. 12710, pp. 266–297. Springer, Heidelberg (May 2021). https://doi.org/10.1007/978-3-030-75245-3_11
7. Bellare, M., Jakobsson, M., Yung, M.: Round-optimal zero-knowledge arguments based on any one-way function. In: Fumy, W. (ed.) EUROCRYPT'97. LNCS, vol. 1233, pp. 280–305. Springer, Heidelberg (May 1997). https://doi.org/10.1007/3-540-69053-0_20
8. Bitansky, N., Kalai, Y.T., Paneth, O.: Multi-collision resistance: a paradigm for keyless hash functions. 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. 671–684. ACM (2018). https://doi.org/10.1145/3188745.3188870
9. Blazy, O., Chevalier, C., Vu, Q.H.: Post-Quantum UC-Secure Oblivious Transfer in the Standard Model with Adaptive Corruptions. In: Proceedings of the 14th International Conference on Availability, Reliability and Security, ARES 2019. pp. 28:1–28:6. ACM (2019). https://doi.org/10.1145/3339252.3339280
10. Brassard, G., Chaum, D., Crépeau, C.: Minimum disclosure proofs of knowledge. J. Comput. Syst. Sci. **37**(2), 156–189 (1988). https://doi.org/10.1016/0022-0000(88)90005-0
11. Canetti, R.: Security and composition of multi-party cryptographic protocols. Cryptology ePrint Archive, Report 1998/018 (1998), https://eprint.iacr.org/1998/018
12. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: 42nd FOCS. pp. 136–145. IEEE Computer Society Press (Oct 2001). https://doi.org/10.1109/SFCS.2001.959888
13. Chia, N.H., Chung, K.M., Liu, Q., Yamakawa, T.: On the impossibility of post-quantum black-box zero-knowledge in constant round. In: 62nd FOCS. pp. 59–67. IEEE Computer Society Press (Feb 2022). https://doi.org/10.1109/FOCS52979.2021.00015
14. David, B., Dowsley, R., Nascimento, A.C.A.: Universally composable oblivious transfer based on a variant of LPN. In: Gritzalis, D., Kiayias, A., Askoxylakis, I.G. (eds.) CANS 14. LNCS, vol. 8813, pp. 143–158. Springer, Heidelberg (Oct 2014). https://doi.org/10.1007/978-3-319-12280-9_10
15. Delpech de Saint Guilhem, C., Orsini, E., Tanguy, T.: Limbo: Efficient zero-knowledge MPCitH-based arguments. In: Vigna, G., Shi, E. (eds.) ACM CCS 2021. pp. 3022–3036. ACM Press (Nov 2021). https://doi.org/10.1145/3460120.3484595
16. Delpech de Saint Guilhem, C., Orsini, E., Tanguy, T., Verbauwhede, M.: Efficient proof of RAM programs from any public-coin zero-knowledge system. Cryptology ePrint Archive, Report 2022/313 (2022), https://eprint.iacr.org/2022/313

17. Döttling, N., Garg, S., Hajiabadi, M., Masny, D., Wichs, D.: Two-round oblivious transfer from CDH or LPN. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 768–797. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45724-2_26
18. Fleischhacker, N., Goyal, V., Jain, A.: On the existence of three round zero-knowledge proofs. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part III. LNCS, vol. 10822, pp. 3–33. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78372-7_1
19. Giacomelli, I., Madsen, J., Orlandi, C.: ZKBoo: Faster zero-knowledge for Boolean circuits. In: Holz, T., Savage, S. (eds.) USENIX Security 2016. pp. 1069–1083. USENIX Association (Aug 2016)
20. Goldreich, O.: Foundations of Cryptography: Basic Tools, vol. 1. Cambridge University Press, Cambridge, UK (2001)
21. Goldreich, O., Kahan, A.: How to construct constant-round zero-knowledge proof systems for NP. Journal of Cryptology **9**(3), 167–190 (Jun 1996)
22. Goldreich, O., Krawczyk, H.: On the composition of zero-knowledge proof systems. In: Paterson, M. (ed.) Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, UK, July 16-20, 1990, Proceedings. Lecture Notes in Computer Science, vol. 443, pp. 268–282. Springer (1990). https://doi.org/10.1007/BFb0032038
23. Goldreich, O., Oren, Y.: Definitions and properties of zero-knowledge proof systems. Journal of Cryptology **7**(1), 1–32 (Dec 1994). https://doi.org/10.1007/BF00195207
24. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM Journal on Computing **18**(1), 186–208 (1989)
25. Green, M., Hohenberger, S.: Practical adaptive oblivious transfer from simple assumptions. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 347–363. Springer, Heidelberg (Mar 2011). https://doi.org/10.1007/978-3-642-19571-6_21
26. Hauck, E., Loss, J.: Efficient and universally composable protocols for oblivious transfer from the CDH assumption. Cryptology ePrint Archive, Report 2017/1011 (2017), https://eprint.iacr.org/2017/1011
27. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Zero-knowledge proofs from secure multiparty computation. SIAM J. Comput. **39**(3), 1121–1152 (2009). https://doi.org/10.1137/080725398
28. Ishai, Y., Prabhakaran, M., Sahai, A.: Founding cryptography on oblivious transfer - efficiently. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 572–591. Springer, Heidelberg (Aug 2008). https://doi.org/10.1007/978-3-540-85174-5_32
29. Kalai, Y.T., Rothblum, G.N., Rothblum, R.D.: From obfuscation to the security of Fiat-Shamir for proofs. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 224–251. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63715-0_8
30. Katz, J.: Which languages have 4-round zero-knowledge proofs? In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 73–88. Springer, Heidelberg (Mar 2008). https://doi.org/10.1007/978-3-540-78524-8_5
31. Lai, Y.F., Galbraith, S.D., Delpech de Saint Guilhem, C.: Compact, efficient and UC-secure isogeny-based oblivious transfer. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 213–241. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_8
32. Lombardi, A., Ma, F., Spooner, N.: Post-quantum zero knowledge, revisited or: How to do quantum rewinding undetectably. In: 63rd FOCS. pp. 851–859. IEEE

Computer Society Press (Oct / Nov 2022). https://doi.org/10.1109/FOCS54457.2022.00086

33. Mi, B., Huang, D., Wan, S., Hu, Y., Choo, K.K.R.: A post-quantum light weight 1-out-n oblivious transfer protocol. Computers & Electrical Engineering **75**, 90–100 (2019). https://doi.org/10.1016/j.compeleceng.2019.01.021
34. Ong, S.J., Vadhan, S.P.: An equivalence between zero knowledge and commitments. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 482–500. Springer, Heidelberg (Mar 2008). https://doi.org/10.1007/978-3-540-78524-8_27
35. Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 554–571. Springer, Heidelberg (Aug 2008). https://doi.org/10.1007/978-3-540-85174-5_31
36. Rabin, M.O.: How to exchange secrets with oblivious transfer. Cryptology ePrint Archive, Report 2005/187 (2005), https://eprint.iacr.org/2005/187

Ref	System	Verifier coins	Black-box Sim	Round	Achievability	Assumption
[22]	Proof	Public	✓	constant	×	
[29]	Proof	Public	×	constant	×	iO
[21]	Proof	Private	✓	5	✓	Claw-free
[18]	Proof	Private	×	3	×	iO
[8]	Proof	Private	×	4	✓	certain HF[3] and LWE
[22]	Arg.	Public	✓	3	×	
[8]	Arg.	Public	×	5	✓	certain HF[3] and LWE
[22]	Arg.	Private	✓	3	×	
[7]	Arg.	Private	✓	4	✓	one-way function
[8]	Arg.	Private	×	3	✓	certain HF[3] and LWE

Table 2. Constant-round Zero-knowledge Protocols.

A Constant-Round Zero-Knowledge

Table 2 surveys the round complexity of computational zero-knowledge protocols. Katz [30] shows that if a language L has a 4-round, black-box, computational zero-knowledge proof system with negligible soundness error, then $\bar{L} \in \mathbf{MA}$. Particularly, assuming the polynomial hierarchy does not collapse, the five rounds computational zero-knowledge proof systems [21] is optimal.

[3] Keyless multi-collision-resistant hash functions.