Untelegraphable Encryption and its Applications

Jeffrey Champion*, Fuyuki Kitagawa†, Ryo Nishimaki†, Takashi Yamakawa†

*UT Austin
jchampion@utexas.edu
†NTT Social Informatics Laboratories
{fuyuki.kitagawa, ryo.nishimaki, takashi.yamakawa}@ntt.com

Abstract

We initiate the study of untelegraphable encryption (UTE), founded on the no-telegraphing principle, which allows an encryptor to encrypt a message such that a binary string representation of the ciphertext cannot be decrypted by a user with the secret key, a task that is classically impossible. This is a natural relaxation of unclonable encryption (UE), inspired by the recent work of Nehoran and Zhandry (ITCS 2024), who showed a computational separation between the no-cloning and no-telegraphing principles.

In this work, we define and construct UTE information-theoretically in the plain model. Building off this, we give several applications of UTE and study the interplay of UTE with UE and well-studied tasks in quantum state learning, yielding the following contributions:

- A construction of collusion-resistant UTE from standard secret-key encryption (SKE). We additionally show that hyper-efficient shadow tomography (HEST) is impossible assuming collusion-resistant UTE exists. By considering a relaxation of collusion-resistant UTE, we are able to show the impossibility of HEST assuming only pseudorandom state generators (which may not imply one-way functions). This almost unconditionally answers an open inquiry of Aaronson (STOC 2018).
- A construction of UTE from a quasi-polynomially secure one-shot message authentication code (OSMAC) in the classical oracle model, such that there is an explicit attack that breaks UE security for an *unbounded* polynomial number of decryptors.
- A construction of *everlasting secure* collusion-resistant UTE, where the decryptor adversary can run in unbounded time, in the quantum random oracle model (QROM), and formal evidence that a construction in the plain model is a challenging task. We additionally show that HEST with unbounded post-processing time (which we call *weakly-efficient* shadow tomography) is impossible assuming everlasting secure collusion-resistant UTE exists.
- A construction of secret sharing for all polynomial-size policies that is resilient to *joint* and *unbounded* classical leakage from collusion-resistant UTE and classical secret sharing for all policies.
- A construction (and definition) of collusion-resistant untelegraphable secret-key functional encryption (UTSKFE) from single-decryptor functional encryption and plain secret-key functional encryption, and a construction of collusion-resistant untelegraphable public-key functional encryption from UTSKFE, plain SKE, and plain public-key functional encryption.

1 Introduction

Unclonable cryptography uses the no-cloning principle of quantum mechanics [Par70, WZ82, Die82] to enable cryptographic properties that are impossible to achieve classically. Constructing primitives with such properties has been an active area of research for the past four decades, starting with the well-studied notion of quantum money [Wie83].

Unclonable encryption. In recent years, there has been particular interest in unclonable encryption (UE), first introduced by Gottesman [Got03], and later refined by Broadbent and Lord [BL20]. Unclonable encryption is a one-time encryption scheme with the following guarantee: any adversary $\mathcal{A}$ that is given a ciphertext of a message m in the form of a quantum state cannot output two (possibly-entangled) states to adversaries $\mathcal{B}$ and $\mathcal{C}$ such that $\mathcal{B}$ and $\mathcal{C}$ can recover information about m given the classical secret key sk. When m is a uniformly chosen message, and $\mathcal{B}$ and $\mathcal{C}$ must recover the entire message, we call this *one-way security*. In the more natural notion of security, called *indistinguishability security*, $\mathcal{A}$ chooses two messages m_0, m_1, receives an encryption of m_b, and $\mathcal{B}$ and $\mathcal{C}$ guess b.

One-way security was achieved information theoretically in [BL20], but indistinguishability security has yet to be achieved from any standard assumption in the plain model. In particular, positive results on unclonable indistinguishability security have come at either the cost of oracles [BL20, AKL+22], relaxed definitions [KN23, AKY24, CG24c], or new conjectures [AB24]. Moreover, there are negative results that bring into question the plausibility of proving unclonable indistinguishability security from standard assumptions in the plain model [MST21, AKL+22].

This work: untelegraphable encryption. Partly motivated by the challenges of constructing UE, the main focus of this work is a natural relaxation of unclonable encryption, which we call untelegraphable encryption (UTE), a name that comes from its relation to the no-telegraphing principle [Wer98]. Like unclonable encryption, the basic primitive we consider is a one-time encryption scheme. The security guarantee is as follows: any adversary $\mathcal{A}$ that is given a ciphertext of a message m in the form of a quantum state cannot produce a classical string st that would allow an adversary $\mathcal{B}$ to recover information about m given st and the classical secret key sk. The one-way and indistinguishability security notions are analogous to the ones in UE.

The core challenges in constructing unclonable encryption from standard assumptions are: (1) there are two (possibly entangled) second stage adversaries, which make most computational reduction and search-to-decision techniques fail (2) the second stage adversaries get the secret key of the encryption scheme along with some function of the ciphertext, which is incompatible with many standard cryptographic primitives and techniques. Untelegraphable encryption by definition removes the first challenge, so there is reason to believe that it is more feasible than unclonable encryption. Furthermore, the study of untelegraphable encryption builds on the rich literature examining the relationship between classical and quantum information.

Quantum and classical information. The "quantum vs classical information" paradigm broadly spans the field of quantum computing. In quantum complexity theory, the fundamental question of whether quantum states are stronger than classical strings as proofs to an efficient verifier (also known as the QMA vs QCMA problem [AN02]) is a very well-studied problem, with numerous works [AK07, FK18, NN23, NZ24, LLPY24] attempting to get closer to an answer via oracle separations. A similar question has also been studied in terms of efficient algorithms interacting with classical and quantum advice [AK07, LLPY24]. In quantum information, two of the fundamental no-go theorems are the no-cloning and no-telegraphing principles. Previously believed to be equivalent, the work of Nehoran and Zhandry [NZ24] shows that when it comes to computationally bounded algorithms, there is indeed a difference between telegraphing and cloning. We believe that we can enrich our understanding of unclonability by studying untelegraphability, even if the focus is on their differences.

Learning quantum states. Furthermore, the fundamental task of describing an unknown quantum state, called quantum state tomography [VR89], aims to classically learn an unknown quantum state given many copies of the state. Due to the destructive nature of measurement, full quantum state tomography provably requires exponentially many copies of the state. This barrier lead Aaronson [Aar19] to introduce the notion of *shadow tomography*. Shadow tomography focuses on learning how a particular family of two-outcome measurements acts on an unknown state. Such a task can be done more efficiently, and may still extract the majority of "useful" information from the state.

In the same work, Aaronson suggested the possibility of so-called *hyper-efficient* shadow tomography (HEST), hoping to achieve a truly computationally feasible variant of shadow tomography. In the subsequent work of Huang, Kueng, and Preskill [HKP20], the notion of *classical shadows* was introduced, which aims for efficiency in the deconstruction of the quantum state into a classical string, such that the classical output can be post-processed (inefficiently) to learn interesting properties of a state. As we will show, HEST and classical shadows are strongly connected to untelegraphable encryption.

On provable guarantees. We further motivate untelegraphable encryption by remarking on the importance of having provable guarantees from well-founded assumptions in cryptography. Such guarantees can give us confidence in constructions that would otherwise derive their security entirely through cryptanalysis. This is the motivation for the notion of non-adaptive security in cryptography, which in general involves an adversary committing to how they will attack a protocol's security. Even though non-adaptive security fails to model the real world, it allows us to gain confidence in constructions, and can later lead to provable adaptive security. We believe that in a similar way, showing a provable guarantee like untelegraphable security for a scheme can be an avenue to gain confidence in the scheme's unclonable security, assuming no explicit cloning attack is known.

Cryptographic uses of UTE. Lastly, we justify UTE as a standalone cryptographic primitive by providing several natural use cases of UTE where UE provides no advantage. Straightforwardly, UTE allows one to prevent the classical exfiltration of important quantumly-stored information, which is an especially useful guarantee in the (possibly near-term) world where we have a small number of large scale quantum computers and many classical users interacting with them. In such a world, there are many potential UTE adversaries, but hardly any UE adversaries.

Alternatively, in any world (even one with many large scale quantum computers) where quantum teleportation is implementable, UTE enables a "perfect" encryption scheme so long as entanglement is shared by the sender and receiver. Specifically, by teleporting a UTE ciphertext, one can guarantee that an eavesdropper who obtains the classical communication needed for this teleportation cannot learn about the message, even given the secret key. Notably, this application does not have a well-defined unclonable analog since there is no quantum state to intercept.

1.1 Our Results

In this section, we give a summary of our contributions:

1. We formally define untelegraphable variants of several cryptographic primitives: untelegraphable encryption and variants on the basic security such as collusion-resistant and everlasting security (Section 2.3), and untelegraphable functional encryption (UTFE)(Section 7.1).

2. We construct untelegraphable encryption (UTE) with indistinguishability security information theoretically (Section 3.2), collusion-resistant UTE from standard secret-key encryption (SKE) (Section 3.3), and show that hyper-efficient shadow tomography (HEST) for general mixed quantum states is impossible assuming collusion-resistant UTE (Section 3.4). By instantiating SKE with succinct query-bounded SKE from a pseudorandom state (PRS) generator (shown in Section 3.5), we are able to show HEST is impossible assuming just PRS generators.

3. We give a separation between indistinguishability secure UTE and one-way secure UE from a quasi-polynomially secure one-shot message authentication code (OSMAC) in the classical oracle model(Section 4.2). The OSMAC can be instantiated with the learning with errors (LWE) assumption.

4. We construct everlasting secure collusion-resistant UTE in the quantum random oracle model (QROM)(Section 5.1)and show that weakly-efficient shadow tomography (WEST) for general mixed quantum states is impossible assuming everlasting secure collusion-resistant UTE(Section 5.2). We also show that public-key everlasting one-way secure UTE is separated from all falsifiable assumptions assuming the sub-exponential hardness of one-way functions(Section 5.3).

5. We construct unbounded joint-leakage-resilient secret sharing (UJLRSS) for polynomial-size policies from collusion-resistant UTE and classical secret sharing for polynomial-size policies(Section 6.2), both of which can be instantiated from standard SKE.

6. We construct collusion-resistant untelegraphable secret-key functional encryption (UTSKFE) from single-decryptor functional encryption and plain secret-key functional encryption (SDFE)(Section 7.2). We also construct collusion-resistant untelegraphable public-key functional encryption from UTSKFE, plain SKE, and plain public-key functional encryption(Section 7.3). The SDFE scheme can be instantiated from sub-exponentially secure indistinguishability obfuscation (iO) and one-way functions, along with LWE.

We omit many details in this paper due to space limitations. Please refer to the full version for further details. We now discuss how some of our results compare with related works.

Impossibility of hyper-efficient shadow tomography. The initial impossibility results shown for HEST by Aaronson [Aar19] and Kretschmer [Kre21] use oracles, so they are incomparable to our result. In the work of Çakan and Goyal [ÇG24b], impossibility of HEST is shown assuming sub-exponential post-quantum iO and LWE. We improve the assumption needed to just PRS generators [JLS18], which may be weaker than one-way functions [Kre21]. Moreover, the WEST impossibility result is unique to our work. Interestingly, the impossibility of WEST somewhat complements a positive quantum state learning result in the work of Huang, Kueng, and Preskill [HKP20], which essentially says that WEST is possible when the family of measurements are all projections to pure quantum states, while we show WEST is impossible in an instance where the measurements are not projections to pure states.

Separating cloning and telegraphing. The work of Nehoran and Zhandry [NZ24] showed the first computational separation of cloning and telegraphing. They show that there exists a quantum oracle and set of states such that the states can be efficiently cloned with respect to the oracle but not efficiently telegraphed. Moreover, their set of untelegraphable states is hard to deconstruct (make classical) even in unbounded time in a way that reconstruction (recovering the state) is efficient. Our separation between UE and UTE roughly shows the existence of a classical oracle and family of states where it is easy to clone many bits of information about the states, but it is hard to efficiently telegraph even a single bit. The results and assumptions needed for each result are incomparable. In the opposite direction, Çakan, Goyal, Liu-Zhang, and Ribeiro [ÇGLR24] show that unclonability likely does not imply "interactive" untelegraphability, where the classical string can come from multiple rounds of classical interaction.

Unbounded leakage-resilient secret sharing. The work of [ÇGLR24] introduces the notion of unbounded leakage-resilient secret sharing (ULRSS). In this variant of secret sharing, security holds even with respect to unbounded *local* classical leakage on each share, meaning the shares must be quantum. Our construction in this work extends security to the setting of *joint* leakage, where the leakage function can depend on unqualified sets of shares that are disjoint from one another. The distinction between these definitions is strict, as the construction in [ÇGLR24] is insecure under our definition. However, for Boolean formula policies, the construction in [ÇGLR24] is statistically secure, while our construction relies on PRS generators.

The work of Ananth, Goyal, Liu, and Liu [AGLL24] defines and constructs the notion of unclonable secret sharing (USS). Intuitively, USS allows for collusion among many or even all parties, where collusion between two parties means they are entangled and cannot communicate otherwise. In our definition of UJLRSS, we consider collusion between groups of parties that cannot reconstruct the secret, but allow for full sharing of information between parties in a group. Furthermore, we construct a secret sharing scheme for all polynomial-size policies, whereas [AGLL24] only consider n-out-of-n secret sharing. We compare the definitions more in Section 6.1.

Untelegraphable keys. Çakan, Goyal, Liu-Zhang, and Ribeiro [ÇGLR24] also show how to construct quantum keys resilient to unbounded classical (interactively generated) leakage. This can be thought of as cryptography with untelegraphable keys in our terminology. Specifically, they show how to protect keys for public-key encryption, digital signatures, and pseudorandom functions. Additionally, the work of [NZ24] shows how to construct an encryption scheme with keys that can be cloned but not exfiltrated.

Independent work. Independent of our work, Çakan and Goyal [ÇG24a] study LOCC leakage-resilient encryption, signatures, and NIZKs. In our terminology, they construct an extension of untelegraphable encryption where the first and second stage adversaries can interact when generating the classical state (before the second stage adversary gets the decryption key). They construct SKE and PKE with this security property, improving our results in Sections 3.2 and 3.3. However, apart from constructing this extension of UTE, the work of [ÇG24a] studies entirely different facets of untelegraphability from our work.

1.2 Technical Overview

In this section, we provide a high-level overview of our main constructions.

UTE with indistinguishability security. We begin with our construction of one-time untelegraphable encryption with statistical indistinguishability security. The construction uses a one-way secure UTE scheme (which we have from one-way secure UE) and a universal hash family as its main components, and can be described as follows:

- The secret key is a secret key $\mathsf{sk}_{\mathsf{OW}}$ for the one-way secure UTE, a random hash function h from the hash family, and a uniformly random string r.
- A ciphertext for a message m is an encryption $|\mathsf{ct}_{\mathsf{OW}}\rangle$ of a random message x for the one-way scheme, along with the string $r' = r \oplus h(x) \oplus m$, where $\oplus$ is the bitwise XOR operation.

To decrypt, simply decrypt the random message x and compute $m = r' \oplus r \oplus h(x)$. The key observation for proving security is that when the first stage adversary $\mathcal{A}$ outputs its two messages (m_0, m_1), the string r' in the ciphertext can be sampled uniformly and given to $\mathcal{A}$ along with the honestly generated $|\mathsf{ct}_{\mathsf{OW}}\rangle$. When $\mathcal{A}$ outputs classical string st for the second stage adversary $\mathcal{B}$, the challenger can then sample the hash function h and set $r = r' \oplus h(x) \oplus m_b$. Such a trick can be thought of like a one-time version of "non-committing encryption", a notion that will be discussed more later in this overview.

Thus, in the second stage, we can now treat the components $(\mathsf{sk}_{\mathsf{OW}}, \mathsf{st}, r')$ as "leakage" on the randomly chosen x. In particular, by the one-way security of the underlying UTE scheme, we can argue that x has high entropy from the perspective of $\mathcal{B}$, and furthermore, the hash h is independent of x and its leakage. This allows us to treat $h(x)$ as a uniform string by a variant of the well-known leftover hash lemma. We refer to Section 3.2 for the full details. It is unknown whether such a scheme satisfies UE security, but the main issue in proving security stems from the fact that one of the second stage adversaries $\mathcal{B}$ and $\mathcal{C}$ can obtain the honest ciphertext in its entirety. This makes it impossible to reason about the traditional notion of entropy from the perspectives of $\mathcal{B}$ and $\mathcal{C}$.

Collusion-resistant UTE. Given UTE with indistinguishability security, we can now consider constructing collusion-resistant UTE. By collusion-resistant security, we mean that the first stage adversary $\mathcal{A}$ can now make numerous message queries of the form (m_0, m_1) and receive encryptions of m_b in return. The term collusion-resistant comes from the fact that in the UE setting, the goal of an adversary given q encryptions would be to transmit the bit b to $q + 1$ second stage adversaries.

Similarly to the one-time case, our construction of collusion-resistant UTE makes use of a hybrid encryption technique using non-committing encryption (NCE) [JL00, CHK05], as has been used previously to construct advanced encryption schemes with certified deletion [HMNY21]. Intuitively, NCE is an encryption scheme with an algorithm for making "fake" ciphertexts that contain no message information, and a corresponding algorithm that "reveals" the true message m via outputting a decryption key dk. The security guarantee says that a fake ciphertext and decryption key look indistinguishable from a real ciphertext and decryption key on some challenge message, given access to an encryption oracle. Due to the nature of its security guarantee, NCE requires encryption and decryption keys to be separately defined. We make use of such a scheme along with a one-time UTE scheme to construct collusion-resistant UTE (with separate encryption and decryption keys) as follows:

- The encryption key ek and decryption key dk are directly sampled from the NCE scheme.
- A ciphertext for a message m consists of a one-time UTE encryption of the message $|\mathsf{ct}_{\mathsf{1UTE}}\rangle$ along with a ciphertext $\mathsf{ct}_{\mathsf{NCE}}$ of the secret key $\mathsf{sk}_{\mathsf{1UTE}}$ for the one-time UTE scheme.

To decrypt given dk, simply recover $\mathsf{sk}_{\mathsf{1UTE}}$ and decrypt the one-time UTE ciphertext. The intuition for security is that for each query (m_0, m_1) made by the first stage adversary $\mathcal{A}$, we can switch the ciphertext $\mathsf{ct}_{\mathsf{NCE}}$ that encrypts $\mathsf{sk}_{\mathsf{1UTE}}$ to a fake one, and have the decryption key dk be produced by the revealing algorithm for the message $\mathsf{sk}_{\mathsf{1UTE}}$. We can then appeal to one-time UTE security to switch the message m_b to 0, say. We refer to Section 3.3 for the full details. Note that such a construction does not yield collusion-resistant security in the UE case, since by the formulation of UE, there has not been a way to appeal to its security multiple times in a single proof. This prevents one from using a query-by-query security proof.

Impossibility of hyper-efficient shadow tomography. We now illustrate the connection between hyper-efficient shadow tomography (HEST) and collusion-resistant UTE. To start, we formalize the notion of hyper-efficient shadow tomography. Let E be a quantum circuit that takes as input some index $i \in [M]$ and an n-qubit quantum state ρ, and outputs a single classical bit. A shadow tomography procedure is an algorithm $\mathcal{T}$ that takes E and k copies of ρ and outputs (a classical description of) a quantum circuit C such that $C(i)$ estimates the probability that $E(i, \rho) = 1$ up to some error ε with failure probability δ. The algorithm $\mathcal{T}$ is *hyper-efficient* if its runtime and the required number of copies k are both $\mathrm{poly}(n, \log M, 1/\varepsilon, \log 1/\delta)$.

The key observation is to map the state ρ to a UTE ciphertext, map M to be the enumeration of UTE decryption keys dk, and map the circuit E to be the UTE decryption algorithm (assuming 1-bit messages). Since UTE algorithms must be efficient, the parameters n and $\log M$ are polynomial in the security parameter. This implies that running the procedure $\mathcal{T}(E, |\mathsf{ct}\rangle^{\otimes k})$, will efficiently generate a classical description of a circuit C, such that $C(\mathsf{dk})$ estimates $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle)$ up to error ε. By setting $\varepsilon < 1/2$ and δ to be negligible, this immediately gives an attack on collusion-resistant UTE with *overwhelming advantage*, so long as the UTE scheme is correct. Since the ciphertext and key sizes in the above scheme are fixed for any polynomial number of queries, we can rule out the existence of all HEST procedures assuming NCE exists. NCE can be constructed from CPA-secure SKE [KNTY19] and thus one-way functions (OWFs), so this immediately gives the impossibility of HEST from OWFs. We provide the full details in Section 3.4.

Weakening the assumption to PRS generators. A notable feature of the above impossibility result is that it hinges on the the ciphertext and key sizes of the underlying UTE scheme being fixed for any polynomial number of queries. Additionally, the copy threshold k is a fixed value, so it is fine for the UTE scheme to have its setup depend on k, so long as the aforementioned succinctness properties hold. Thus, the result would still hold if we were to construct "bounded-query" NCE with succinct ciphertexts and decryption keys, where the initial key generation can depend on the a query bound q along with the security parameter λ, analogous to notions like bounded-collusion functional encryption. Thus, in hopes of strengthening our separation even further, we aim to construct succinct bounded-query NCE from pseudorandom state (PRS) generators, which may not imply OWFs.

We do this by constructing succinct bounded-query CPA secure SKE from a pseudorandom function-like state (PRFS) generator with a polynomial-sized domain, which is implied by a PRS generator. Recall that a PRFS generator is a QPT algorithm G such that for a uniform random classical key k, tuples $(x, G(k, x))$ are indistinguishable from tuples $(x, |\psi\rangle)$, where each $|\psi\rangle$ is an independent Haar-random state and inputs x must be chosen non-adaptively. Reminiscent of bounded-collusion attribute-based encryption constructions (see [ISV+17]), our construction is as follows:

- The secret key consists of λ random keys for the PRFS generator $(k_1, \ldots, k_\lambda)$. The dependence on q comes from the fact that our PRFS generator G will have domain $\{0,1\}^d$ where d depends on q.
- To encrypt a message $m \in \{0,1\}$, sample shares $s_1, \cdots, s_\lambda$ uniformly such that $s_1 \oplus \cdots \oplus s_\lambda = m$ and sample $r_1, \ldots, r_\lambda$ uniformly from $\{0,1\}^{d-1}$. The ciphertext consists of pairs $(r_i, G(k_i, (r_i, s_i)))$ for all $i \in [\lambda]$.

Decryption involves testing the PRFS generator outputs: for each i, a decryptor checks whether the output is $G(k_i, (r_i, 0))$. This can be done for PRFS generators with negligible error [AQY22].

To invoke the PRFS generator security, it is crucial that the queried inputs x are distinct from each other. Since for each i the values r_i in the q ciphertexts queried are uniformly random, we can ensure they do not collide by setting the input size d to be large enough relative to q. Namely, by setting d such that the r_i are sampled from a set of size q^2, we can ensure that for a given i the probability of the q strings r_i colliding is some constant less than 1. Therefore, the chance of a collision happening for all $i \in [\lambda]$ is exponentially small in λ, since the r_i, r_j are sampled independently for $i \neq j$. With this in mind, we can switch the output of G to Haar-random states on the q inputs (r_{i^*}, s_{i^*}), where i^* is the first index such that the q strings r_{i^*} do not collide, and the q bits s_{i^*} are sampled randomly (which matches their marginal distribution). Since the PRFS outputs on index i^* are switched for Haar-random states, they no longer depend on the share s_{i^*}, meaning the remaining shares s_i for $i \neq i^*$ are uniform strings independent of the message, yielding security.

Since the size of $\{0,1\}^{d-1}$ needs to be at least q^2 for security, setting $d = 2\log q + 1$ suffices. Additionally, the output and key length of G need not depend on q, which implies ciphertexts and keys are succinct. By inspecting the transformations from CPA-secure SKE to NCE and NCE to collusion-resistant UTE, it is easy to see that succinctness is

retained. Due to the $\mathrm{poly}(\lambda, \log q)$-size ciphertexts and keys, for every HEST scheme needing k copies, we can find $q = \mathrm{poly}(\lambda)$ such that $q > k$, which means HEST contradicts q-bounded collusion resistance of UTE. Since q-bounded collusion resistance can be achieved for any $q = \mathrm{poly}(\lambda)$, the impossibility follows. Note that k circularly depends on q here, so if parameters are not succinct the number of copies k may always be larger than q, breaking the impossibility argument. We provide the full details in Section 3.5.

Separating UE and UTE. Our next result is a construction of a clonable UTE scheme: that is, a secure UTE scheme such that the message information in the ciphertext can be cloned and thus violate UE security. Note that if the number of second stage adversaries is bounded, the question is trivial, since one could just define encryption to encrypt the message many times with a collusion-resistant UTE scheme. Thus, we aim to break UE security for an unbounded polynomial number of second stage adversaries.

To do this, we will want to make use of the fact that we can repeatedly rerun a given second stage adversary $\mathcal{B}$ for UTE on a state st. Specifically, imagine we have a classical oracle $\mathcal{O}$ (think of it as an ideal obfuscation) that checks some condition and outputs the message m if the condition holds. From the perspective of attacking UE, we would like the property that $\mathcal{A}$ can easily sample many inputs that pass the check in the first stage if its output is allowed to be quantum. From a UTE security perspective, if $\mathcal{B}$ wins the game, then it must query $\mathcal{O}$ at an input which passes the check, and we want the property that we can sample prohibitively many inputs that pass the check by running $\mathcal{B}$ repeatedly, enough to break security of some primitive.

We instantiate this framework using a one-shot message authentication code (OSMAC). An OSMAC is a relaxation of one-shot signatures [AGKZ20], which support public verification, and two-tier one-shot signatures [MPY24], which support partial public verification. An OSMAC consists of a setup algorithm that outputs the public parameters pp and master verification key mvk, a key generation algorithm that uses pp to sample a verification key vk along with a quantum signing key $|\mathsf{sk}\rangle$, a signing algorithm that uses a signing key $|\mathsf{sk}\rangle$ to sign a message m to output a MAC σ, and finally a verification algorithm that checks σ with respect to $\mathsf{mvk}, \mathsf{pp}, \mathsf{vk}, m$. Security says that an adversary cannot output valid MACs on two different messages with respect to the same vk given pp. Our approach will be to use an "OSMAC chain" to realize the desired functionality. We illustrate our approach for a chain of length 2, since it is sufficient to elucidate the core idea. Our clonable UTE scheme has separate encryption and decryption keys, and is defined as follows:

- The encryption key ek is a tuple $(\mathsf{mvk}, \mathsf{pp})$ from the OSMAC scheme, along with a random string s, while the decryption key dk is just s.
- A ciphertext for a message m is a signing key $|\mathsf{sk}_\varepsilon\rangle$, the public parameters pp, and a classical oracle $\mathcal{O}$ that has ek, m, and vk_ε hardwired inside it, where vk_ε is the verification key generated with $|\mathsf{sk}_\varepsilon\rangle$. Given strings s' and $r = (r_1, r_2) \in \{0,1\}^2$ along with tuples

$$(\mathsf{vk}_0, \mathsf{vk}_1, \sigma_\varepsilon), (\mathsf{vk}_{r_1}, \mathsf{vk}_{r_1\|0}, \mathsf{vk}_{r_1\|1}, \sigma_{r_1}), (\mathsf{vk}_r, \sigma_r),$$

  the function $\mathcal{O}$ checks that $s' = s$, the MAC σ_v is valid on message $\mathsf{vk}_{v\|0}\|\mathsf{vk}_{v\|1}$ with respect to vk_v for $v \in \{\varepsilon, r_1\}$, and finally that σ_r is a valid MAC on s' with respect to vk_r. If all these checks pass, $\mathcal{O}$ outputs m. Otherwise, $\mathcal{O}$ outputs $\perp$.

To decrypt given $(|\mathsf{sk}_\varepsilon\rangle, \mathsf{pp}, \mathcal{O})$ and s, compute the MAC chain on 00: sample pairs $(|\mathsf{sk}_v\rangle, \mathsf{vk}_v)$ for $v \in \{0, 1, 00, 01\}$, compute the MACs σ_ε on $\mathsf{vk}_0\|\mathsf{vk}_1$ using $|\mathsf{sk}_\varepsilon\rangle$, σ_0 on $\mathsf{vk}_{00}\|\mathsf{vk}_{01}$ using $|\mathsf{sk}_0\rangle$, and σ_{00} on s using $|\mathsf{sk}_{00}\rangle$. Submit the tuples along with s and 00 to $\mathcal{O}$ to recover m. The UE attack for 4 second stage adversaries is simply to compute the MAC tree without the leaves:

$$(\mathsf{vk}_0, \mathsf{vk}_1, \sigma_\varepsilon), (\mathsf{vk}_0, \mathsf{vk}_{00}, \mathsf{vk}_{01}, \sigma_0), (\mathsf{vk}_1, \mathsf{vk}_{10}, \mathsf{vk}_{11}, \sigma_1).$$

Additionally send one of $(\mathsf{vk}_r, |\mathsf{sk}_r\rangle)$ for $r \in \{0,1\}^2$ to each second stage adversary. When a second stage adversary is given s, it can sign s with $|\mathsf{sk}_r\rangle$ to complete the MAC chain and recover m. Thus, this attack breaks one-way UE security given a single ciphertext in the first stage and 4 second stage adversaries. By increasing the length of r to $\omega(\log \lambda)$, the attack works for an unbounded polynomial number of second stage adversaries.

To see UTE security, first note that the oracle $\mathcal{O}$ is useless to the first stage adversary $\mathcal{A}$ since s is uniform and independent of $\mathcal{A}$'s view. Furthermore, for the second stage adversary $\mathcal{B}$ to win the security game, it must query the oracle $\mathcal{O}$ with a valid MAC chain with noticeable probability. Therefore, we can query $\mathcal{B}$ with randomly sampled strings s and measure a random query of $\mathcal{B}$, check that the value of s matches, and add that query to a list L. By doing this enough times, we can argue that with high probability there are at least 5 valid MAC chains contained in L, which means by the pigeonhole principle, there must exist two chains in L such that the values of r are equal and the values of s are not (which follows by setting s to have sufficient length). By picking two elements of L at random, we can find these two MAC chains with roughly $1/|L|^2$ probability. These MAC chains must contradict OSMAC security, because either we have two signatures that verify with vk_r or the two vk_r values are different, which means the messages signed with respect to vk_{r_1} are different. Then, we either have two signatures that verify with vk_{r_1} or the two vk_{r_1} values are different, implying there are definitely two signatures that verify with vk_ε, since vk_ε is always fixed. If the length of r is set to $\log^2 \lambda$ for the attack, this strategy works by assuming quasi-polynomial OSMAC security.

Everlasting secure UTE. Numerous works in cryptography have studied the natural notion of "everlasting security" [MQU07], which roughly says that after some notable event occurs in a security game, a security property is upheld even given unbounded time after the event. Given the two-stage nature of UTE (and UE), it is natural to consider whether everlasting untelegraphability (or unclonability) is possible. Concretely, this means that in the UTE security game, the second stage adversary can run in *unbounded* time. We show that the techniques used for certified everlasting hiding commitments in [HMNY22] adapted to the UTE setting indeed yields everlasting secure UTE. Given a one-time secure UTE scheme, CPA-secure SKE, and a hash function H modeled as a random oracle, the construction is as follows:

- The secret key is the key k for the SKE scheme along with the hash function H.
- A ciphertext for a message m consists of a one-time UTE encryption $|\mathsf{ct}_{\mathsf{1UTE}}\rangle$ of m, an encryption $\mathsf{ct}_{\mathsf{SKE}}$ of a random string r, and a string $h = H(r) \oplus \mathsf{sk}_{\mathsf{1UTE}}$, that masks the fresh one-time UTE key $\mathsf{sk}_{\mathsf{1UTE}}$.

To decrypt, simply recover r to get $H(r)$ and decrypt the one-time UTE ciphertext. The basic intuition for security is that by SKE security, the oracle given to $\mathcal{A}$ in the first stage is useless, and thus $\mathsf{sk}_{\mathsf{1UTE}}$ is statistically hidden by $H(r)$ until $\mathcal{B}$ learns it in the second stage. Security then follows by security of the one-time UTE scheme, since it is statistically secure. Extending to the collusion-resistant setting follows readily by a hybrid argument. We give the full details in Section 5.1. This technique can be thought of as leveraging the random oracle to get a stronger variant of NCE. A notable aspect of this construction is that in the single-ciphertext setting, it can be adapted to get everlasting indistinguishability secure public-key UE by using public-key encryption instead of SKE, since indistinguishability secure UE exists in the QROM [AKL+22].

Impossibility of weakly-efficient shadow tomography. We define weakly-efficient shadow tomography in a similar way to HEST. The main difference is that we explicitly split the procedure into two stages: the first efficient stage outputs any classical string st, and the second stage can use unbounded time to estimate $E(i, \rho)$ up to error ε given st and i. Thus, the attack on collusion-resistant everlasting secure UTE follows the same outline as the HEST attack on collusion-resistant UTE. The main notable difference is that our instantiation of collusion-resistant everlasting secure UTE is secure in the QROM. Thus, we need to slightly adapt the attack to make oracle access to the hash H instead of having E generically compute the decryption circuit, which would need to include a concrete description of H (in the case where the tomography algorithm $\mathcal{T}$ needs the circuit described in terms of atomic gates only). This can be done by modifying E to be the decryption circuit starting from after $H(r)$ is recovered (we can put $\mathsf{ct}_{\mathsf{SKE}}$ and h as part of $\mathcal{A}$'s output in the UTE game). We refer to Section 5.2 for the details.

Impossiblity of public-key everlasting secure UTE. A natural question is whether a construction of everlasting secure UTE (or UE) is possible in the plain model, as our positive result heavily relies on the QROM. However, we observe that this is not likely to be the case, at least using standard proof strategies which make black-box use of the adversary.

To start, notice that any efficient reduction $\mathcal{R}$ that makes black-box use of a UTE adversary $(\mathcal{A}, \mathcal{B})$ in the everlasting one-way security game to break some candidate falsifiable assumption A cannot query adversary $\mathcal{B}$, since $\mathcal{B}$ is potentially unbounded. Furthermore, there is no requirement on the structure of the string st output by $\mathcal{A}$, other than it being classical. Now, suppose that given a ciphertext $|\mathsf{ct}\rangle$ of a random message m, adversary $\mathcal{A}$ could recover m. While $\mathcal{A}$ could just output m in the clear and win the game, what if it were to output some encoding $\mathsf{Enc}(m)$ of the message? If different messages under Enc are statistically far apart, the unbounded adversary $\mathcal{B}$ could still recover m and win the game. Moreover, if encodings of different messages are computationally close, the reduction $\mathcal{R}$ would not be able to tell the difference between the encodings $\mathsf{Enc}(m)$ and $\mathsf{Enc}(0)$, meaning $\mathcal{R}$ cannot even determine whether $(\mathcal{A}, \mathcal{B})$ wins!

Thus, if we consider an inefficient adversary $\mathcal{A}$ that decrypts the ciphertext $|\mathsf{ct}\rangle$ it is given to get a message m and outputs $\mathsf{Enc}(m)$, the reduction $\mathcal{R}^{\mathcal{A}}$ that has oracle access to $\mathcal{A}$ must break the candidate assumption A, since adversary $\mathcal{B}$ is unbounded and can distinguish encodings of different messages. Since the ciphertext is quantum and there are no guarantees on its structure, we consider public-key UTE (PKUTE), which allows adversary $\mathcal{A}$ to recover m by finding a corresponding secret key to the public key it is given (e.g. by rejection sampling key pairs). Now, given that $\mathcal{R}^{\mathcal{A}}$ must break the candidate assumption A, we indistinguishably switch oracle access to $\mathcal{A}$ with oracle access to an efficient simulator $\mathcal{S}$, that will essentially ignore its input and output $\mathsf{Enc}(0)$. Since encodings of different messages are computationally close, $\mathcal{R}^{\mathcal{S}}$ must break the assumption A. But $\mathcal{R}^{\mathcal{S}}$ is efficient, so A must be false. This means there is either no black-box reduction to a falsifiable assumption A that shows everlasting one-way PKUTE security, or A is false. We supply the formal details in Section 5.3.

Unbounded joint-leakage-resilient secret sharing. In Section 6.2, we show that collusion-resistant UTE and classical secret sharing for all policies can be used to construct unbounded joint-leakage-resilient secret sharing (UJLRSS) for all polynomial-size monotone Boolean circuits. In this overview, we will show how to construct UJLRSS for all polynomial-size monotone Boolean formulas, since it is simpler and still captures the basic intuition.

We start by defining UJLRSS more formally. The syntax of UJLRSS is the same as regular n-party secret sharing, except that the shares of a classical message m output by the sharing algorithm are quantum states, and thus the input to the reconstruction algorithm contains quantum states. Security has two stages:

- In the first stage, an adversary $\mathcal{A}_0$ outputs messages m_0, m_1 and a "collusion partition" of disjoint sets $V^*, V_1, \ldots, V_\ell$ such that their union is the set of all indices $[n]$, and each V does not satisfy the policy. The challenger then shares m_b, and for each $i \in [\ell]$ gives the shares $|s_j\rangle$ for $j \in V_i$ to adversary $\mathcal{A}_i$. Each adversary $\mathcal{A}_i$ outputs a state st_i.
- In the second stage, adversary $\mathcal{B}$ is given st_i for each $i \in [\ell]$, along with shares $|s_j\rangle$ for $j \in V^*$, and tries to guess the bit b.

This definition captures the scenario where an adversarial group that does not satisfy the policy is able to exfiltrate the remaining shares classically. In such a case, our security notion guarantees that the adversarial group learns nothing about the secret. Alternatively, this definition can be thought of as ensuring that each individual share, along with each possible set of unqualified shares, is untelegraphable.

We now present a construction of UJLRSS for a monotone Boolean formula P from collusion-resistant UTE and a classical secret sharing scheme for P:

- To share a message m with respect to a policy P, sample classical shares $(s'_1, \ldots, s'_n)$ for m and $(\mathsf{sk}_1, \ldots, \mathsf{sk}_n)$ for a fresh UTE secret key $\mathsf{sk}_{\mathsf{UTE}}$. A share $|s_i\rangle$ for party $i \in [n]$ is sk_i along with an encryption $|\mathsf{ct}_i\rangle$ of s'_i.
- To reconstruct given a satisfying group of parties, reconstruct the key $\mathsf{sk}_{\mathsf{UTE}}$, decrypt the ciphertexts to get shares of m, and reconstruct m from those shares.

The intuition for security is that for each set V_i, we can appeal to collusion-resistant UTE security to switch the ciphertexts to encryptions of 0 instead of shares of m. To simulate the secret key shares sk_j for $\mathcal{A}_i$, sample them uniformly, since classical secret sharing for formulas is information-theoretically simulation secure. After $\mathsf{sk}_{\mathsf{UTE}}$ is revealed in the UTE security game, derive the remaining shares and run $\mathcal{A}_j$ for $j \neq i$ to get the remaining states, which can then be given to the second stage UJLRSS adversary to complete the reduction. After switching ciphertexts for all V_i, security follows by information theoretic security of the classical scheme, since the only classical shares left are the

ones in V^*. Since classical secret sharing for all monotone Boolean circuits is computationally secure, we can no longer easily simulate the key shares for an unsatisfying group, so the construction and proof strategy must be changed to adapt to this. We provide the full details in Section 6.2.

Untelegraphable functional encryption. Finally, we give a sketch of our untelegraphable secret-key and public-key functional encryption constructions. To start, we formalize the security definition of untelegraphable functional encryption. The first stage adversary $\mathcal{A}$ can query function keys and still must choose messages m_0, m_1 such that $f(m_0) = f(m_1)$ for all functions f queried, much like regular functional encryption, except $\mathcal{A}$ now outputs a state st. The second stage adversary $\mathcal{B}$ can query function keys as well, but with no constraints on the function f, even functions such that $f(m_0) \neq f(m_1)$. One can also consider the stronger definition where $\mathcal{B}$ is given the master secret key in the second stage, but in this work we only consider the unrestricted key query setting.

We start by constructing untelegraphable secret-key functional encryption (UTSKFE) from challenge-only single-decryptor functional encryption (SDFE)[1] [KN22] and plain secret-key functional encryption (SKFE). This construction essentially switches the roles of the SDFE function keys and ciphertext. We describe it as follows:

- The master key is an SDFE key pair $(\mathsf{pk}_{\mathsf{SDFE}}, \mathsf{msk}_{\mathsf{SDFE}})$ along with a plain SKFE master key $\mathsf{msk}_{\mathsf{SKFE}}$.
- A function key for a function f is an SDFE encryption ct_f of a freshly sampled function key sk_f from the SKFE scheme.
- A ciphertext for a message m is a function key $|\mathsf{sk}_G\rangle$ for the SDFE scheme, where G has a SKFE ciphertext $\mathsf{ct}_{\mathsf{SKFE}}$ of m hardwired, and on input an SKFE function key sk_f, decrypts $\mathsf{ct}_{\mathsf{SKFE}}$ with sk_f.

To decrypt, simply run the decryption algorithm of the SDFE scheme. Intuitively, security follows by appealing to SDFE security to switch each second stage key query to encrypt 0 instead of sk_f. Since the first stage is essentially the plain SKFE security game, security now follows by SKFE security. Since the SDFE scheme is challenge-only secure, this yields single-ciphertext secure untelegraphable SKFE. See Section 7.2for further details.

For our final construction, we show how to build untelegraphable public-key functional encryption from single-ciphertext secure UTSKFE, plain public-key functional encryption (PKFE), plain SKE, and a secure PRF. This construction follows the hybrid encryption blueprint from [ABSV15]. We sketch the construction as follows:

- The public key is just the public-key $\mathsf{pk}_{\mathsf{PKFE}}$ for the PKFE scheme. The master key is the master key $\mathsf{msk}_{\mathsf{PKFE}}$ for the PKFE scheme along with a secret key $\mathsf{sk}_{\mathsf{SKE}}$ for the SKE scheme.
- A function key for a function f is a PKFE function key sk_G, where $G[\mathsf{ct}_{\mathsf{SKE}}, f]$ is a function with f and a "dummy ciphertext" $\mathsf{ct}_{\mathsf{SKE}}$ of 0 under $\mathsf{sk}_{\mathsf{SKE}}$ hard-wired. The function G takes as input an indicator bit b along with three keys $\mathsf{msk}_{\mathsf{SKFE}}$, k, and $\mathsf{sk}_{\mathsf{SKE}}$, where k is a PRF key and $\mathsf{msk}_{\mathsf{SKFE}}$ is a master key for the UTSKFE scheme. If $b = 0$, G outputs sk_f using $\mathsf{msk}_{\mathsf{SKFE}}$ along with randomness from the PRF on f, and if $b = 1$, G outputs the decryption of $\mathsf{ct}_{\mathsf{SKE}}$ using $\mathsf{sk}_{\mathsf{SKE}}$.
- A ciphertext for a message m is a ciphertext $|\mathsf{ct}_{\mathsf{SKFE}}\rangle$ of m for freshly sampled $\mathsf{msk}_{\mathsf{SKFE}}$ along with a PKFE ciphertext $\mathsf{ct}_{\mathsf{PKFE}}$ of the message $(0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)$, where k is a fresh PRF key.

To decrypt, run PKFE decryption to get a key sk_f for the UTSKFE scheme and use sk_f to decrypt $|\mathsf{ct}_{\mathsf{SKFE}}\rangle$ for the message. Security follows the approach from [ABSV15]. First, we switch the dummy ciphertext $\mathsf{ct}_{\mathsf{SKE}}$ to encrypt sk_f of the UTSKFE scheme with randomness derived from the PRF on f. Then we switch the PKFE message from $(0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)$ to $(1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})$, which crucially relies on the fact that

$$G[\mathsf{ct}_{\mathsf{SKE}}, f](0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot) = G[\mathsf{ct}_{\mathsf{SKE}}, f](1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})$$

for all f, even when $f(m_0) \neq f(m_1)$, a property that was not utilized in [ABSV15]. To complete the proof, we switch the key generation randomness for generating sk_f from the UTSKFE scheme from the PRF on f to a random string, and finally appeal to UTSKFE security. See Section 7.3for details.

[1]We technically need to modify the definition to the untelegraphable setting such that it is stated quite differently than SDFE in [KN22], but for ease of exposition, we call it SDFE here.

2 Preliminaries

Let λ denote the security parameter unless otherwise specified. We write $\mathrm{poly}(\cdot)$ to denote an arbitrary polynomial and $\mathsf{negl}(\cdot)$ to denote an arbitrary negligible function. We say an event happens with overwhelming probability if the probability is at least $1 - \mathsf{negl}(\lambda)$.

For strings x and y, we use $x\|y$ and (x, y) to denote the concatenation of x and y. Let $[\ell]$ denote the set of integers $\{1, \cdots, \ell\}$. In this paper, for a finite set X and a distribution D, $x \xleftarrow{R} X$ denotes selecting an element from X uniformly at random, $x \leftarrow D$ denotes sampling an element x according to D. Let $y \leftarrow \mathcal{A}(x)$ denote assigning to y the output of a probabilistic or quantum algorithm $\mathcal{A}$. When we explicitly show that $\mathcal{A}$ uses randomness r, we write $y \leftarrow \mathcal{A}(x; r)$. We say an algorithm runs in quantum polynomial time (QPT) if it runs in quantum polynomial time in the size of its input, and we say an algorithm runs in probabalistic polynomial time (PPT) if it runs in classical polynomial time in the size of its input.

2.1 Quantum Information

Let $\mathcal{H}$ be a finite-dimensional complex Hilbert space. A (pure) quantum state is a vector $|\psi\rangle \in \mathcal{H}$. Let $\mathcal{S}(\mathcal{H})$ be the space of Hermitian operators on $\mathcal{H}$. A density matrix is a Hermitian operator $\rho \in \mathcal{S}(\mathcal{H})$ with $\mathrm{Tr}(\mathcal{X}) = 1$, which is a probabilistic mixture of pure states. A quantum state over $\mathcal{H} = \mathbb{C}^2$ is called qubit, which can be represented by the linear combination of the standard basis $\{|0\rangle, |1\rangle\}$. More generally, a quantum system over $(\mathbb{C}^2)^{\otimes n}$ is called an n-qubit quantum system for $n \in \mathbb{N} \setminus \{0\}$.

We write $\mathcal{H}_{\mathsf{R}}$ to denote that the Hilbert space $\mathcal{H}$ is tied to the register R. Also, we sometimes write $\mathcal{X}_{\mathsf{R}}$ to emphasize that the operator $\mathcal{X}$ acts on the register R. When we apply $\mathcal{X}_{\mathsf{R}_1}$ to registers R_1 and R_2, $\mathcal{X}_{\mathsf{R}_1}$ is identified with $\mathcal{X}_{\mathsf{R}_1} \otimes \boldsymbol{I}_{\mathsf{R}_2}$.

A unitary operation is represented by a complex matrix $\boldsymbol{U}$ such that $\boldsymbol{U}\boldsymbol{U}^\dagger = \boldsymbol{I}$. The operation $\boldsymbol{U}$ transforms $|\psi\rangle \in \mathcal{H}$ and $\rho \in \mathcal{S}(\mathcal{H})$ into $\boldsymbol{U}|\psi\rangle$ and $\boldsymbol{U}\rho\boldsymbol{U}^\dagger$, respectively. A projector $\boldsymbol{P}$ is a Hermitian operator ($\boldsymbol{P}^\dagger = \boldsymbol{P}$) such that $\boldsymbol{P}^2 = \boldsymbol{P}$.

For a quantum state ρ over two registers R_1 and R_2, we denote the state in R_1 as $\rho[\mathsf{R}_1]$, where $\rho[\mathsf{R}_1] = \mathrm{Tr}_2[\rho]$ is a partial trace of ρ (trace out R_2). For quantum states ρ_1 and ρ_2, $\mathsf{TD}(\rho_1, \rho_2)$ denotes the trace distance between ρ_1 and ρ_2.

2.2 Quantum Accessible Oracle and Useful Lemma

Given a function $F : X \to Y$, a quantum-accessible oracle O of F is modeled by a unitary transformation $\boldsymbol{U}_F$ operating on two registers $\mathsf{in} \otimes \mathsf{out}$, in which $|x\rangle|y\rangle$ is mapped to $|x\rangle|y \oplus F(x)\rangle$, where $\oplus$ denotes XOR group operation on Y. We write $\mathcal{A}^{|O\rangle}$ to denote that the algorithm $\mathcal{A}$'s oracle O is a quantum-accessible oracle.

Simulation of quantum random oracles. In this paper, following many previous works in the QROM, we give quantum-accessible random oracles to reduction algorithms if needed. This is just a convention. We can efficiently simulate quantum-accessible random oracles perfectly by using $2q$-wise independent hash function [Zha12], where q is the number of queries to the quantum-accessible random oracles by an adversary.

One-Way to Hiding (O2H) Lemma. We now recall the one-way to hiding lemma [AHU19].

Lemma 2.1 (O2H Lemma [AHU19])**.** *Let $G, H : X \to Y$ be any functions, z be a random value, and $S \subseteq X$ be a random set such that $G(x) = H(x)$ holds for every $x \notin S$. The tuple (G, H, S, z) may have arbitrary joint distribution. Furthermore, let $\mathcal{A}$ be a quantum oracle algorithm that makes at most q quantum queries. Let $\mathcal{B}$ be an algorithm that on input z, choose $i \leftarrow [q]$, runs $\mathcal{A}^H(z)$, measures $\mathcal{A}$'s i-th query, and outputs the measurement outcome. Then, we have*

$$\left|\Pr\left[\mathcal{A}^H(z) = 1\right] - \Pr\left[\mathcal{A}^G(z) = 1\right]\right| \leq 2q \cdot \sqrt{\Pr[\mathcal{B}^H(z) \in S]} \ .$$

2.3 Cryptographic Primitives

In this section, we introduce the core primitive of this work, which we call untelegraphable encryption. In doing so, we recall the notion of unclonable encryption, which will be needed at points throughout this work as well. Additionally, we provide the definition of CPA-secure secret-key encryption (SKE) for reference in **??**.

Unclonable encryption. We now recall the definition of unclonable secret key encryption (USKE) [BL20, AK21], which is often referred to as just unclonable encryption (UE).

Definition 2.2 (Unclonable Secret Key Encryption)**.** An unclonable secret key encryption scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{UE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with the following syntax:

- $\mathsf{Gen}(1^\lambda) \to \mathsf{sk}$: On input the security parameter λ, the key generation algorithm outputs a (classical) secret key sk.
- $\mathsf{Enc}(\mathsf{sk}, m) \to |\mathsf{ct}\rangle$: On input the secret key sk and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm outputs a quantum ciphertext $|\mathsf{ct}\rangle$.
- $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle) \to m$: On input the secret key sk and a quantum ciphertext $|\mathsf{ct}\rangle$, the decryption algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{UE} satisfy the following correctness property, along with at least one of the following security properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$ and any message $m \in \mathcal{M}_\lambda$, we have

$$\Pr\left[m' = m : \begin{array}{c} \mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda) \\ |\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{sk}, m) \\ m' \leftarrow \mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda)$$

- **One-way Security:** For a security parameter λ and a two-stage adversary $(\mathcal{A}, \mathcal{B}, \mathcal{C})$, we define the one-way security game as follows:

  1. At the beginning of the game, the challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$, $m \xleftarrow{R} \mathcal{M}_\lambda$, and $|\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{sk}, m)$. It gives $|\mathsf{ct}\rangle$ to $\mathcal{A}$.
  2. Adversary $\mathcal{A}$ outputs a quantum state ρ_{BC} in register B and C, and sends the corresponding registers to $\mathcal{B}$ and $\mathcal{C}$.
  3. The challenger gives sk to $\mathcal{B}$ and $\mathcal{C}$. Adversaries $\mathcal{B}$ and $\mathcal{C}$ output strings $m_\mathcal{B}$ and $m_\mathcal{C}$ respectively. The output of the experiment is a bit b which is 1 when $m = m_\mathcal{B} = m_\mathcal{C}$ and 0 otherwise.

  We say a USKE scheme satisfies (computational) one-way security if for all (efficient) adversaries $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, $\Pr[b = 1] = \mathsf{negl}(\lambda)$ in the one-way security game.

- **Indistinguishability Security:** For a security parameter λ and a two-stage adversary $(\mathcal{A}, \mathcal{B}, \mathcal{C})$, we define the indistinguishability security game as follows:

  1. Adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$.
  2. The challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$, $b \xleftarrow{R} \{0, 1\}$, and $|\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{sk}, m_b)$. It gives $|\mathsf{ct}\rangle$ to $\mathcal{A}$.
  3. Adversary $\mathcal{A}$ outputs a quantum state ρ_{BC} in register B and C, and sends the corresponding registers to $\mathcal{B}$ and $\mathcal{C}$.
  4. The challenger gives sk to $\mathcal{B}$ and $\mathcal{C}$. Adversaries $\mathcal{B}$ and $\mathcal{C}$ output bits $b_\mathcal{B}$ and $b_\mathcal{C}$ respectively. The output of the experiment is a bit b which is 1 when $b = b_\mathcal{B} = b_\mathcal{C}$ and 0 otherwise.

We say a USKE scheme satisfies statistical (computational) indistinguishability security if for all (efficient) adversaries $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, $\Pr[b = 1] \leq 1/2 + \mathsf{negl}(\lambda)$ in the above indistinguishability security game.

Definition 2.3 (UE with k second stage adversaries)**.** We say Π_{UE} satisfies *k-adversary security* if it is secure for first stage adversaries $\mathcal{A}$ in Definition 2.2 that output a quantum state ρ to k registers $B_1, \ldots, B_k$ that are given to corresponding second stage adversaries $(\mathcal{B}_1, \ldots, \mathcal{B}_k)$. All adversaries $B_1, \ldots, B_k$ must guess m in the one-way security game or the challenge bit b in the indistinguishability security game.

Theorem 2.4 (Statistical One-Way UE [BL20, AKL+22])**.** *There exists statistically one-way secure USKE where the maximum advantage of the adversary is* $2^{-\Omega(\lambda)}$.

Untelegraphable encryption. We now define the core primitive considered in this work. Untelegraphable encryption has the same syntax and correctness requirements as unclonable encryption above. The main security difference is that the output of the first stage adversary must be purely classical. In particular, this means there need not be two second stage adversaries as in unclonable encryption, since the classical string can be copied.

Definition 2.5 (Untelegraphable Secret Key Encryption)**.** An untelegraphable secret key encryption (UTSKE) scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda \in \mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with the same syntax and correctness requirement as Definition 2.2. We require that Π_{UTE} satisfy at least one of the following security properties:

- **One-way Security:** For a security parameter λ and a two-stage adversary $(\mathcal{A}, \mathcal{B})$, we define the one-way security game as follows:

  1. At the beginning of the game, the challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$, $m \xleftarrow{R} \mathcal{M}_\lambda$, and $|\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{sk}, m)$. It gives $|\mathsf{ct}\rangle$ to $\mathcal{A}$.
  2. Adversary $\mathcal{A}$ outputs a classical string st which is given to adversary $\mathcal{B}$.
  3. The challenger gives sk to $\mathcal{B}$. Adversary $\mathcal{B}$ outputs a string $m_{\mathcal{B}}$. The output of the experiment is a bit b which is 1 when $m = m_{\mathcal{B}}$ and 0 otherwise.

  We say a UTSKE scheme satisfies (computational) one-way security if for all (efficient) adversaries $(\mathcal{A}, \mathcal{B})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, $\Pr[b = 1] = \mathsf{negl}(\lambda)$ in the one-way security game.

- **Indistinguishability Security:** For a security parameter λ, a bit $b \in \{0, 1\}$, and a two-stage adversary $(\mathcal{A}, \mathcal{B})$, we define the indistinguishability security game as follows:

  1. Adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$.
  2. The challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$ and $|\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{sk}, m_b)$. It gives $|\mathsf{ct}\rangle$ to $\mathcal{A}$.
  3. Adversary $\mathcal{A}$ outputs a classical string st which is given to adversary $\mathcal{B}$.
  4. The challenger gives sk to $\mathcal{B}$. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.

  We say Π_{UTE} satisfies statistical (computational) indistinguishability security if for all (efficient) adversaries $(\mathcal{A}, \mathcal{B})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

  $$|\Pr[b_{\mathcal{B}} = 1 | b = 0] - \Pr[b_{\mathcal{B}} = 1 | b = 1]| = \mathsf{negl}(\lambda)$$

  in the above indistinguishability security game.

Definition 2.6 (Collusion-Resistant Security)**.** We say that an untelegraphable (unclonable) SKE scheme Π_{UTE} (Π_{UE}) satisfies *collusion-resistant* security if the first-stage adversary $\mathcal{A}$ in Definition 2.5 (Definition 2.2) can make many message queries $\{m_0^{(i)}, m_1^{(i)}\}_{i \in [Q]}$ adaptively and get encryptions of $\{m_b^{(i)}\}_{i \in [Q]}$ in return. We say the scheme is *t-copy secure* if the first-stage adversary specifies a single pair of messages (m_0, m_1) and gets t encryptions of m_b in return.

Definition 2.7 (Everlasting Security)**.** We say that an untelegraphable (unclonable) SKE scheme Π_{UTE} (Π_{UE}) satisfies *everlasting* security if the first stage adversary $\mathcal{A}$ in Definition 2.5 (Definition 2.2) is efficient while the second stage adversary is allowed to run in unbounded time.

Corollary 2.8 (Statistical One-Way UTSKE)**.** *There exists statistically one-way secure UTSKE where the maximum advantage of the adversary is* $2^{O(\lambda)}$.

Proof. Follows immediately by Theorem 2.4. In particular, any telegraphing adversary defines a cloning adversary which simply gives st to both $\mathcal{B}$ and $\mathcal{C}$. □

CPA-secure encryption. We finally recall the basic notion of CPA-secure secret-key encryption, which will be used at various points throughout this work. Classical CPA-secure encryption can be constructed from any one-way function.

Definition 2.9 (Secret-Key Encryption)**.** A secret key encryption scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of PPT algorithms $\Pi_{\mathsf{SKE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with the following syntax:

- $\mathsf{Gen}(1^\lambda) \to \mathsf{sk}$: On input the security parameter λ, the key generation algorithm outputs a secret key sk.
- $\mathsf{Enc}(\mathsf{sk}, m) \to \mathsf{ct}$: On input the secret key sk and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm outputs a ciphertext ct.
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to m$: On input the secret key sk and a ciphertext ct, the decryption algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{SKE} satisfy the following correctness property, along with at least one of the following security properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$ and any message $m \in \mathcal{M}_\lambda$, we have

$$\Pr\left[m' = m : \begin{array}{c} \mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{sk}, m) \\ m' \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda)$$

- **One-way CPA Security:** For a security parameter λ and an adversary $\mathcal{A}$, we define the one-way security game as follows:

  1. At the beginning of the game, the challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$, $m^* \xleftarrow{R} \mathcal{M}_\lambda$, and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{sk}, m^*)$. It gives ct to $\mathcal{A}$.
  2. Adversary $\mathcal{A}$ has access to an encryption oracle that on input $m \in \mathcal{M}_\lambda$ outputs $\mathsf{Enc}(\mathsf{sk}, m)$.
  3. At the end of the game, adversary $\mathcal{A}$ outputs $m' \in \mathcal{M}_\lambda$. The output of the experiment is a bit b which is 1 when $m' = m^*$ and 0 otherwise.

  We say an SKE scheme satisfies one-way CPA security if for all efficient adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, $\Pr[b = 1] = \mathsf{negl}(\lambda)$ in the one-way security game.

- **CPA Security:** For a security parameter λ, a bit $b \in \{0, 1\}$, and an adversary $\mathcal{A}$, we define the indistinguishability security game as follows:

  1. At the beginning of the game, the challenger samples $\mathsf{sk} \leftarrow \mathsf{Gen}(1^\lambda)$.
  2. Adversary $\mathcal{A}$ can make encryption queries on pairs of messages $m_0, m_1 \in \mathcal{M}_\lambda$. The challenger replies with $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{sk}, m_b)$
  3. At the end of the game, adversary $\mathcal{A}$ outputs $b' \in \{0, 1\}$, which is the output of the experiment.

We say an SKE scheme satisfies CPA-security if for all efficient adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

$$|\Pr[b' = 1|b = 0] - \Pr[b' = 1|b = 1]| = \mathsf{negl}(\lambda)$$

in the above indistinguishability security game. We say Π_{SKE} is *query-bounded* if the adversary can make at most q total queries, where q is additionally given to Gen.

Remark 2.10 (Quantum Ciphertexts). We can easily generalize Definition 2.9 to the setting where ciphertexts are quantum, which we abbreviate by QSKE.

3 Untelegraphable Encryption and the Impossibility of Hyper-Efficient Shadow Tomography

In this section, we construct untelegraphable SKE with indistiguishability security from one-way secure untelegraphable SKE. Next, we show how to upgrade the UTSKE to satisfy collusion-resistant security by additionally assuming the existence of non-committing encryption for receiver (NCER), which can be constructed from CPA-secure encryption. We then recall the notion of hyper-efficient shadow tomography (HEST), and use its existence to attack any collusion-resistant UTSKE scheme. This implies that HEST cannot exist for general states under the assumption that CPA-secure encryption exists. Finally, we show pseudorandom state (PRS) generators are sufficient for the impossibility of HEST, by constructing bounded-query CPA-secure encryption with succinct quantum ciphertexts and classical keys from PRS generators, where succinctness is with respect to the bound parameter.

3.1 Building Blocks

In this section, we describe preliminaries and notions that are needed for our constructions.

Min-entropy. We recall some basic definitions on min-entropy. Our definitions are adapted from those in [DRS04]. For a (discrete) random variable X, we write $\mathbf{H}_\infty(X) = -\log(\max_x \Pr[X = x])$ to denote its min-entropy. For two (possibly correlated) discrete random variables X and Y, we define the average min-entropy of X given Y to be $\mathbf{H}_\infty(X \mid Y) = -\log(\mathbb{E}_{y\leftarrow Y} \max_x \Pr[X = x \mid Y = y])$. The optimal probability of an unbounded adversary guessing X given the correlated value Y is $2^{-\mathbf{H}_\infty(X|Y)}$.

Leftover hash lemma. Our construction will also rely on the generalized leftover hash lemma (LHL) from [BDK+11]:

Theorem 3.1 (LHL with Conditional Min-Entropy [BDK+11, Theorem 3.2, adapted])**.** *Let (X, Z) be random variables sampled from some joint distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Z}$. Let $\mathcal{H} = \{h\colon \mathcal{X} \to \{0,1\}^v\}$ be a family of universal hash functions, and let $L = \mathbf{H}_\infty(X \mid Z) - v$ be the entropy loss. Let $\mathcal{A}(r, h, z)$ be a (possibly probabilistic) distinguisher where*

$$\Pr[\mathcal{A}(r, h, z) = 1 : r \xleftarrow{R} \{0,1\}^v, h \xleftarrow{R} \mathcal{H}, (x, z) \leftarrow \mathcal{D}] \leq \varepsilon.$$

Then, the distinguishing advantage of $\mathcal{A}$ on the following distributions is at most $\sqrt{\varepsilon/2^L}$:

$$\left\{(h(x), h, z) : \begin{matrix}(x,z)\leftarrow\mathcal{D}\\ h\xleftarrow{R}\mathcal{H}\end{matrix}\right\} \text{ and } \left\{(r, h, z) : \begin{matrix}(x,z)\leftarrow\mathcal{D}\\ r\xleftarrow{R}\{0,1\}^v, h\xleftarrow{R}\mathcal{H}\end{matrix}\right\}$$

Corollary 3.2 (LHL with Conditional Min-Entropy)**.** *Let (X, Z) be random variables sampled from some joint distribution $\mathcal{D}$ over $\mathcal{X}\times\mathcal{Z}$. Let $\mathcal{H} = \{h\colon \mathcal{X} \to \{0,1\}^v\}$ be a family of universal hash functions. Let $L = \mathbf{H}_\infty(X \mid Z) - v$ be the entropy loss. Then the statistical distance between the following distributions is at most $2^{-L/2}$:*

$$\left\{(h(x), h, z) : \begin{matrix}(x,z)\leftarrow\mathcal{D}\\ h\xleftarrow{R}\mathcal{H}\end{matrix}\right\} \text{ and } \left\{(r, h, z) : \begin{matrix}(x,z)\leftarrow\mathcal{D}\\ r\xleftarrow{R}\{0,1\}^v, h\xleftarrow{R}\mathcal{H}\end{matrix}\right\}$$

Proof. Follows by setting $\varepsilon = 1$ in Theorem 3.1 (which captures *all* distinguishers). □

Pseudorandom function-like state generators. We recall the notion of a pseudorandom function-like state (PRFS) generator [AQY22]. Intuitively, this is the quantum analog of a pseudorandom function (PRF). For polynomial sized domains, PRFS generators can be constructed from pseudorandom state (PRS) generators [JLS18], which may not imply one-way functions [Kre21]. We use the definition of a PRFS generator since it will be more convenient.

Definition 3.3 (PRFS Generator [AQY22])**.** Let λ be a security parameter. We say that a QPT algorithm G is a selectively secure pseudorandom function-like state generator if for all polynomials $s = s(\lambda), t = t(\lambda)$, QPT adversaries $\mathcal{A}$ and a family of distinct indices $(\{x_1, \ldots, x_s\} \subseteq \{0,1\}^{d(\lambda)})_\lambda$, there exists a negligible function $\mathsf{negl}(\lambda)$ such that for all $\lambda \in \mathbb{N}$,

$$\left| \Pr_{k \xleftarrow{R} \{0,1\}^\lambda} [\mathcal{A}(\{x_i, G_\lambda(k, x_i)^{\otimes t}\}_{i\in[s]}) = 1] \right.$$
$$\left. - \Pr_{|\psi_1\rangle,\ldots,|\psi_s\rangle \leftarrow \mu_n} [\mathcal{A}(\{x_i, |\psi_i\rangle^{\otimes t}\}_{i\in[s]}) = 1] \right| = \mathsf{negl}(\lambda),$$

where μ_n is the Haar measure on n-qubit states. We note that there is an efficient algorithm $\mathsf{Test}(k, x, G(k, x'))$ that determines if $x' = x$ with some $\mathsf{negl}(\lambda)$ error so long as $n = \omega(\log \lambda)$ [AQY22].

Non-committing encryption. We also recall the notion of non-committing encryption (NCE). Specifically, we define secret-key non-committing encryption for receiver (SK-NCER), which is the secret-key variant of NCER [JL00, CHK05]. Roughly speaking, SK-NCER is SKE which allows one to generate a "fake" ciphertext so that it can be later revealed to any message along with a "fake" decryption key. The formal definition is given below.

Definition 3.4 (Secret-Key Non-Committing Encryption for Receiver)**.** Let λ be a security parameter. A secret key non-committing encryption for receiver (SK-NCER) scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of efficient algorithms $\Pi_{\mathsf{NCE}} = (\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Fake}, \mathsf{Reveal})$ with the following syntax:

- $\mathsf{KG}(1^\lambda) \to (\mathsf{ek}, \mathsf{dk})$: On input the security parameter 1^λ, the key generation algorithm outputs an encryption key ek and decryption key dk.[2]
- $\mathsf{Enc}(\mathsf{ek}, m) \to \mathsf{ct}$: On input an encryption key ek and message $m \in \mathcal{M}_\lambda$, the encryption algorithm outputs a ciphertext ct.
- $\mathsf{Dec}(\mathsf{dk}, \mathsf{ct}) \to m$: On input a decryption key dk and ciphertext ct, the decryption algorithm outputs a message $m \in \mathcal{M}_\lambda$.
- $\mathsf{Fake}(\mathsf{ek}) \to (\mathsf{ct}, \mathsf{st})$: On input an encryption key ek, the ciphertext faking algorithm outputs a fake ciphertext ct and state st.
- $\mathsf{Reveal}(\mathsf{st}, m^*)$: On input a state st and message $m^* \in \mathcal{M}_\lambda$, the reveal algorithm outputs a fake decryption key dk^*.

We require that Π_{NCE} satisfy the following properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}_\lambda$,
$$\Pr\left[\mathsf{Dec}(\mathsf{dk}, \mathsf{Enc}(\mathsf{ek}, m)) = m : (\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{KG}(1^\lambda)\right] \geq 1 - \mathsf{negl}(\lambda).$$
- **Non-committing Security:** For a security parameter λ, a bit $b \in \{0,1\}$, and an adversary $\mathcal{A}$, we define the non-committing security game as follows:
  1. At the beginning of the game, the challenger samples $(\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{KG}(1^\lambda)$.
  2. Adversary $\mathcal{A}$ makes arbitrarily many encryption queries and a single challenge query in any order:

[2]One may wonder why encryption and decryption keys are separately defined though we consider a secret-key primitive. The reason is that the security of SK-NCER involves an adversary that obtains a decryption key, and we cannot include all secret information in a decryption key.

- **Encryption Query:** When $\mathcal{A}$ makes an encryption query $m \in \mathcal{M}_\lambda$, the challenger computes $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{ek}, m)$ and returns ct to $\mathcal{A}$.
- **Challenge Query:** When $\mathcal{A}$ makes a challenge query $m^* \in \mathcal{M}_\lambda$, the challenger returns a tuple $(\mathsf{ct}^*, \mathsf{dk}^*)$. If $b = 0$, the challenger computes $\mathsf{ct}^* \leftarrow \mathsf{Enc}(\mathsf{ek}, m^*)$ and sets $\mathsf{dk}^* = \mathsf{dk}$. If $b = 1$, the challenger computes $(\mathsf{ct}^*, \mathsf{st}) \leftarrow \mathsf{Fake}(\mathsf{ek})$ and $\mathsf{dk}^* \leftarrow \mathsf{Reveal}(\mathsf{st}, m^*)$.

3. Adversary $\mathcal{A}$ outputs a bit $b' \in \{0, 1\}$, which is the output of the experiment.

We say Π_{NCE} satisfies non-committing security if for all efficient adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

$$|\Pr[b' = 1|b = 0] - \Pr[b' = 1|b = 1]| = \mathsf{negl}(\lambda)$$

in the above security game. We say Π_{NCE} is query-bounded if the adversary can make at most q total queries, where q is additionally given to KG.

Theorem 3.5 (NCE from CPA-secure SKE [KNTY19])**.** *If CPA-secure secret-key encryption exists, then there exists secret-key NCE for receiver.*

Corollary 3.6 (Bounded Query NCE from Bounded Query CPA-secure SKE)**.** *If bounded query CPA-secure secret-key encryption exists, then there exists bounded query secret-key NCE for receiver, where the ciphertext and decryption key sizes are $2\ell|\mathsf{ct}|$ and $\ell|\mathsf{sk}|$, where $|\mathsf{ct}|, |\mathsf{sk}|$ are the sizes of the SKE ciphertext and key, and ℓ is the message length.*

Proof. Follows by Theorem 3.5 and inspection of the construction in [KNTY19]. □

Generalizing to quantum ciphertexts. We can generalize Definition 3.4 to support quantum ciphertexts with a small change in the syntax. This will allow us to construct query-bounded NCE from pseudorandom state generators in Section 3.5.

Hyper-efficient shadow tomography. We finally recall the notion of hyper-efficient shadow tomography (HEST), introduced in [Aar19]. Intuitively, shadow tomography is a quantum state learning task where the objective is to learn a how a specific quantum circuit acts on a state given many copies of the state.

Definition 3.7 (Hyper-Efficient Shadow Tomography [Aar19])**.** Let E denote a uniform quantum circuit family with classical binary output that takes as input $i \in [M]$ and an n-qubit quantum state ρ. Then, a shadow tomography procedure $\mathcal{T}$ takes as input E and k copies of ρ, and outputs a quantum circuit C such that $\Pr[|C(i) - \Pr[E(i, \rho) = 1]| \leq \varepsilon] \geq 1-\delta$. The procedure is said to be *hyper-efficient* if the number of copies k and the runtime are both $\mathrm{poly}(n, \log M, 1/\varepsilon, \log 1/\delta)$.

3.2 Untelegraphable Encryption with Indistinguishability Security

In this section, we construct a one-time secure untelegraphable SKE scheme with indistinguishability security from one-way secure UTSKE and a universal hash function.

Construction 3.8 (One-Time UTE)**.** Let λ be a security parameter. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{OWUTE}} = (\mathsf{OWUTE.Gen}, \mathsf{OWUTE.Enc}, \mathsf{OWUTE.Dec})$ be a one-way secure UTSKE scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$.
- Let $\mathcal{H}$ be a universal hashing family of size at most $2^{\mathrm{poly}(\lambda)}$ where each function $h : \mathcal{M}_\lambda \to \{0, 1\}$ has domain $\mathcal{M}_\lambda$ and range $\{0, 1\}$.

We now construct our UTSKE scheme $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Gen}(1^\lambda)$: On input the security parameter λ, the key generation algorithm samples $\mathsf{sk}_{\mathsf{OW}} \leftarrow \mathsf{OWUTE.Gen}(1^\lambda)$, $h \xleftarrow{R} \mathcal{H}, r \xleftarrow{R} \{0, 1\}$ and outputs $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r)$.

- $\mathsf{Enc}(\mathsf{sk}, m)$: On input the secret key $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r)$ and a message $m \in \{0,1\}$, the encryption algorithm samples $x \xleftarrow{R} \mathcal{M}_\lambda$, computes $|\mathsf{ct}_{\mathsf{OW}}\rangle \leftarrow \mathsf{OWUTE.Enc}(\mathsf{sk}_{\mathsf{OW}}, x)$, and outputs the quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{OW}}\rangle, r \oplus h(x) \oplus m)$.
- $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$: On input the secret key $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r)$ and a quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{OW}}\rangle, r')$, the decryption algorithm outputs $m = r' \oplus r \oplus h(\mathsf{OWUTE.Dec}(\mathsf{sk}_{\mathsf{OW}}, |\mathsf{ct}_{\mathsf{OW}}\rangle))$.

Theorem 3.9 (Correctness)**.** *Suppose Π_{OWUTE} is correct. Then, Construction 3.8 is correct.*

Proof. Fix any $\lambda \in \mathbb{N}$ and $m \in \{0,1\}$. Let $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r) \leftarrow \mathsf{Gen}(1^\lambda)$ and $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{OW}}\rangle, r \oplus h(x) \oplus m) \leftarrow \mathsf{Enc}(\mathsf{sk}, m)$. We consider the output of the decryption algorithm $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$. By correctness of Π_{OWUTE}, we have $x \leftarrow \mathsf{OWUTE.Dec}(\mathsf{sk}_{\mathsf{OW}}, |\mathsf{ct}_{\mathsf{OW}}\rangle)$ with overwhelming probability. It is then immediate that $r \oplus h(x) \oplus m \oplus r \oplus h(x) = m$, so m is the output of $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$ with overwhelming probability. □

Theorem 3.10 (Indistinguishability Security)**.** *Suppose Π_{OWUTE} is statistically one-way secure, where the maximum advantage of the adversary is $2^{-\Omega(\lambda)}$. Then, Construction 3.8 satisfies statistical indistinguishability security.*

Proof. We prove the theorem by showing encryptions of 0 and 1 are statistically indistinguishable. Suppose there exists a two-stage adversary $(\mathcal{A}, \mathcal{B})$ that has non-negligible advantage δ in the indistinguishability security game. We define a sequence of hybrid experiments:

- $\mathsf{Hyb}_0^{(b)}$: This is the real indistinguishability game with bit $b \in \{0,1\}$. In particular:
  - At the beginning of the game, the challenger samples $\mathsf{sk}_{\mathsf{OW}} \leftarrow \mathsf{OWUTE.Gen}(1^\lambda), h \xleftarrow{R} \mathcal{H}, r \xleftarrow{R} \{0,1\}$ and sets $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r)$. The challenger then samples $x \xleftarrow{R} \mathcal{M}_\lambda$, computes $|\mathsf{ct}_{\mathsf{OW}}\rangle \leftarrow \mathsf{OWUTE.Enc}(\mathsf{sk}_{\mathsf{OW}}, x)$, and gives the quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{OW}}\rangle, r' = r \oplus h(x) \oplus b)$ to $\mathcal{A}$.
  - Adversary $\mathcal{A}$ then outputs a classical string st, which is given to $\mathcal{B}$ along with sk from the challenger. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.
- $\mathsf{Hyb}_1^{(b)}$: Same as $\mathsf{Hyb}_0^{(b)}$ except the challenger samples the r' component of the challenge ciphertext as $r' \xleftarrow{R} \{0,1\}$, and sets the r component of the secret key as $r = r' \oplus h(x) \oplus b$.
- $\mathsf{Hyb}_2^{(b)}$: Same as $\mathsf{Hyb}_1^{(b)}$ except the challenger samples $u \xleftarrow{R} \{0,1\}$ and sets $r = r' \oplus u$.

We write $\mathsf{Hyb}_i^{(b)}(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of $\mathsf{Hyb}_i^{(b)}$ with adversary $(\mathcal{A}, \mathcal{B})$. We now argue that each adjacent pair of distributions are indistinguishable.

Lemma 3.11. *The experiments $\mathsf{Hyb}_0^{(b)}(\mathcal{A}, \mathcal{B})$ and $\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B})$ are identically distributed.*

Proof. Since $r \xleftarrow{R} \{0,1\}$ in $\mathsf{Hyb}_0^{(b)}(\mathcal{A}, \mathcal{B})$, the marginal distribution of r' is uniformly random. Moreover, by setting $r = r' \oplus h(x) \oplus b$, the relation $r' = r \oplus h(x) \oplus b$ still holds in $\mathsf{Hyb}_1^{(b)}$. Thus, the view of $\mathcal{A}$ and $\mathcal{B}$ is identical in the two experiments. □

Lemma 3.12. *Suppose Π_{OWUTE} is statistically one-way secure, where the maximum advantage of the adversary is $2^{-\Omega(\lambda)}$. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left| \Pr\left[\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] \right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes $\mathsf{Hyb}_1^{(b)}$ and $\mathsf{Hyb}_2^{(b)}$ with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to contradict Corollary 3.2. Define the joint distribution $\mathcal{D}_{\mathcal{A}}$ over $\mathcal{X} \times \mathcal{Z}$ as follows:

- $\mathcal{X}$: Sample $x \xleftarrow{R} \mathcal{M}_\lambda$. Output x.
- $\mathcal{Z}$: Sample $\mathsf{sk}_{\mathsf{OW}} \leftarrow \mathsf{OWUTE.Gen}(1^\lambda), r' \xleftarrow{R} \{0,1\}$, and compute $|\mathsf{ct}_{\mathsf{OW}}\rangle \leftarrow \mathsf{OWUTE.Enc}(\mathsf{sk}_{\mathsf{OW}}, x)$. Run $\mathcal{A}$ on input $(|\mathsf{ct}_{\mathsf{OW}}\rangle, r')$ to get st. Output $(\mathsf{sk}_{\mathsf{OW}}, \mathsf{st}, r')$.

Since Π_{OWUTE} is statistically one-way secure, where the maximum advantage of the adversary is $2^{-\Omega(\lambda)}$, and r' is sampled independently of x, it must be that $\mathbf{H}_\infty(\mathcal{X} \mid \mathcal{Z}) \geq \Omega(\lambda)$ holds. We now construct a distinguisher $\mathcal{A}'$ that contradicts Corollary 3.2:

1. Algorithm $\mathcal{A}'$ gets an LHL challenge (u, h, z) with joint distribution $\mathcal{D}_{\mathcal{A}}$ as described above.
2. Algorithm $\mathcal{A}'$ parses $z = (\mathsf{sk}_{\mathsf{OW}}, \mathsf{st}, r')$, sets $r = r' \oplus u \oplus b$, sets $\mathsf{sk} = (\mathsf{sk}_{\mathsf{OW}}, h, r)$, runs adversary $\mathcal{B}$ on input $(\mathsf{st}, \mathsf{sk})$, and outputs whatever $\mathcal{B}$ outputs.

If $u \leftarrow h(x)$, then $\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B})$ is perfectly simulated by definition of $\mathcal{D}_{\mathcal{A}}$. If $u \xleftarrow{R} \{0,1\}$, then $\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B})$ is perfectly simulated, since the bit b is fixed. Thus, $\mathcal{A}'$ is able to distinguish the distributions in Corollary 3.2 with advantage $\delta' = \omega(2^{-\Omega(\lambda)}) = \omega(2^{-L/2})$, a contradiction. □

Lemma 3.13. *The experiments* $\mathsf{Hyb}_2^{(0)}(\mathcal{A}, \mathcal{B})$ *and* $\mathsf{Hyb}_2^{(1)}(\mathcal{A}, \mathcal{B})$ *are identically distributed.*

Proof. By construction, the challenger's behavior in Hyb_2 is *independent* of the challenge bit $b \in \{0,1\}$, so the adversary's view in the two distributions is identical. □

Combining Lemmata 3.11 to 3.13, statistical indistinguishability security follows by a standard hybrid argument. □

Remark 3.14 (Longer Messages). Since the only limitation to message length is the entropy loss L in Corollary 3.2, by setting the security parameter for the one-way scheme to be λ^2, we can achieve indistinguishability security for λ-bit messages, which extends to $\mathrm{poly}(\lambda)$-bit messages (assuming one-way functions) via key encapsulation techniques.

3.3 Collusion-Resistant Untelegraphable Encryption

In this section, we give a construction of collusion-resistant untelegraphable encryption from one-time UTE with indistinguishability security and SK-NCER. Note that due to the definition of SK-NCER (Definition 3.4), we inherit the need for separate encryption and decryption keys, where only the decryption key gets revealed to the second stage adversary.

Construction 3.15. Let λ be a security parameter and $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda \in \mathbb{N}}$ be a message space. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{1UTE}} = (\mathsf{1UTE.Gen}, \mathsf{1UTE.Enc}, \mathsf{1UTE.Dec})$ be a one-time secure UTSKE scheme with message space $\mathcal{M}$ and key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda \in \mathbb{N}}$.
- Let $\Pi_{\mathsf{NCE}} = (\mathsf{NCE.KG}, \mathsf{NCE.Enc}, \mathsf{NCE.Dec}, \mathsf{NCE.Fake}, \mathsf{NCE.Reveal})$ be an SK-NCER scheme with message space $\mathcal{K}$.

We construct our collusion-resistant UTE scheme $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Gen}(1^\lambda)$: On input the security parameter λ, the key generation algorithm outputs a pair of keys $(\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{NCE.KG}(1^\lambda)$.
- $\mathsf{Enc}(\mathsf{ek}, m)$: On input the encryption key ek and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm samples $\mathsf{sk}_{\mathsf{1UTE}} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$, computes $|\mathsf{ct}_{\mathsf{1UTE}}\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}, m), \mathsf{ct}_{\mathsf{NCE}} \leftarrow \mathsf{NCE.Enc}(\mathsf{ek}, \mathsf{sk}_{\mathsf{1UTE}})$ and outputs $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{1UTE}}\rangle, \mathsf{ct}_{\mathsf{NCE}})$.
- $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle)$: On input the decryption key dk and a quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_1\rangle, \mathsf{ct}_2)$, the decryption algorithm outputs $\mathsf{1UTE.Dec}(\mathsf{NCE.Dec}(\mathsf{dk}, \mathsf{ct}_2), |\mathsf{ct}_1\rangle)$.

Theorem 3.16 (Correctness)**.** *Suppose* Π_{1UTE} *and* Π_{NCE} *are correct. Then, Construction 3.15 is correct.*

Proof. Fix any $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}_\lambda$. Let $(\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{Gen}(1^\lambda), |\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{1UTE}}\rangle, \mathsf{ct}_{\mathsf{NCE}}) \leftarrow \mathsf{Enc}(\mathsf{ek}, m)$. We consider the output of $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle)$. By correctness of Π_{NCE} and definition of Enc, we have $\mathsf{sk}_{\mathsf{1UTE}} \leftarrow \mathsf{NCE.Dec}(\mathsf{dk}, \mathsf{ct}_{\mathsf{NCE}})$ with overwhelming probability. Similarly, by Π_{1UTE} correctness and definition of Enc, we have $m \leftarrow \mathsf{1UTE.Dec}(\mathsf{sk}_{\mathsf{1UTE}}, |\mathsf{ct}_{\mathsf{1UTE}}\rangle)$ with overwhelming probability, which completes the proof. □

Theorem 3.17 (Collusion-Resistant Security)**.** *Suppose* Π_{1UTE} *satisfies indistinguishability security (Definition 2.5) and* Π_{NCE} *satisfies non-committing security for QPT adversaries. Then, Construction 3.15 satisfies collusion-resistant security.*

Proof. Suppose there exists a QPT two-stage adversary $(\mathcal{A}, \mathcal{B})$ that has non-negligible advantage δ in the collusion-resistant security game and makes at most Q message queries. We define a sequence of hybrid experiments:

- Hyb_0: This is the collusion-resistant game with bit $b = 0$. In particular:
  - At the beginning of the game, the challenger samples $(\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{NCE.KG}(1^\lambda)$.
  - When adversary $\mathcal{A}$ makes its ith query $(m_0^{(i)}, m_1^{(i)})$, the challenger samples $\mathsf{sk}_{\mathsf{1UTE}}^{(i)} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$ and computes
  $$\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_0^{(i)}) \quad \text{and} \quad \mathsf{ct}_{\mathsf{NCE}}^{(i)} \leftarrow \mathsf{NCE.Enc}(\mathsf{ek}, \mathsf{sk}_{\mathsf{1UTE}}^{(i)}).$$
  The challenger gives the ciphertext $\left|\mathsf{ct}^{(i)}\right\rangle = \left(\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle, \mathsf{ct}_{\mathsf{NCE}}^{(i)}\right)$ to $\mathcal{A}$.
  - Adversary $\mathcal{A}$ then outputs a classical string st, which is given to $\mathcal{B}$ along with dk from the challenger. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.
- Hyb_i: Same as Hyb_{i-1} except the challenger now computes $\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_1^{(i)})$.

Note that Hyb_Q is the collusion-resistant game with bit $b = 1$. We write $\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of Hyb_i with adversary $(\mathcal{A}, \mathcal{B})$. We now show that for each $i \in [Q]$, experiments $\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B})$ and $\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B})$ are indistinguishable.

Lemma 3.18. *Suppose* Π_{1UTE} *satisfies indistinguishability security and* Π_{NCE} *satisfies non-committing security for QPT adversaries. Then, for all* $i \in [Q]$*, there exists a negligible function* $\mathsf{negl}(\cdot)$ *such that for all* $\lambda \in \mathbb{N}$*, we have*

$$\left|\Pr\left[\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr[\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B}) = 1]\right| = \mathsf{negl}(\lambda).$$

Proof. We define the following intermediate hybrid experiments:

- Hyb'_{i-1}: Same as Hyb_{i-1} except the challenger now computes $(\mathsf{ct}_{\mathsf{NCE}}^{(i)}, \mathsf{st}_{\mathsf{NCE}}) \leftarrow \mathsf{NCE.Fake}(\mathsf{ek})$ and gives $\mathsf{dk}^* \leftarrow \mathsf{NCE.Reveal}(\mathsf{st}_{\mathsf{NCE}}, \mathsf{sk}_{\mathsf{1UTE}}^{(i)})$ to $\mathcal{B}$.
- Hyb''_{i-1}: Same as Hyb'_{i-1} except the challenger now computes $\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_1^{(i)})$.

We show that each adjacent pair of experiments from the list $(\mathsf{Hyb}_{i-1}, \mathsf{Hyb}'_{i-1}, \mathsf{Hyb}''_{i-1}, \mathsf{Hyb}_i)$ are indistinguishable.

Claim 3.19. Suppose Π_{NCE} satisfies non-committing security for QPT adversaries. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks non-committing security as follows:

1. For each $j \in [Q]$, adversary $\mathcal{A}'$ samples $\mathsf{sk}_{\mathsf{1UTE}}^{(j)} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$.
2. When adversary $\mathcal{A}$ makes the jth query $(m_0^{(j)}, m_1^{(j)})$, adversary $\mathcal{A}'$ does the following:
   - For all $j \in [Q]$, compute
   $$\left|\mathsf{ct}_{\mathsf{1UTE}}^{(j)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(j)}, m_{b_j}^{(j)}),$$
   where $b_j = 0$ for $j \geq i$ and $b_j = 1$ for $j < i$.

- For $j \neq i$, compute $\mathsf{ct}_{\mathsf{NCE}}^{(j)}$ by making an encryption query on message $\mathsf{sk}_{\mathsf{1UTE}}^{(j)}$.
- For $j = i$, make a challenge query on $\mathsf{sk}_{\mathsf{1UTE}}^{(i)}$ to get $(\mathsf{ct}_{\mathsf{NCE}}^{(i)}, \mathsf{dk}^*)$.

Adversary $\mathcal{A}'$ replies with $\left|\mathsf{ct}^{(j)}\right\rangle = \left(\left|\mathsf{ct}_{\mathsf{1UTE}}^{(j)}\right\rangle, \mathsf{ct}_{\mathsf{NCE}}^{(j)}\right)$.

3. When adversary $\mathcal{A}$ outputs st, adversary $\mathcal{A}'$ gives $(\mathsf{st}, \mathsf{dk}^*)$ to $\mathcal{B}$, and outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is QPT since $(\mathcal{A}, \mathcal{B})$ is. If the ciphertext and decryption key are computed normally, $\mathcal{A}'$ perfectly simulates $\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B})$. If the ciphertext and decryption key are faked, $\mathcal{A}'$ perfectly simulates $\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B})$. Thus, $\mathcal{A}'$ has advantage δ' in the non-committing security game, a contradiction. □

Claim 3.20. Suppose Π_{1UTE} satisfies indistinguishability security. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}''_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $(\mathcal{A}', \mathcal{B}')$ that breaks indistinguishability security as follows:

1. Adversary $\mathcal{A}'$ samples $(\mathsf{ek}, \mathsf{dk}) \leftarrow \mathsf{NCE.KG}(1^\lambda)$.
2. When adversary $\mathcal{A}$ makes the jth query $(m_0^{(j)}, m_1^{(j)})$, adversary $\mathcal{A}'$ does the following:
   - If $j \neq i$, sample $\mathsf{sk}_{\mathsf{1UTE}}^{(j)} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$ and compute
   $$\left|\mathsf{ct}_{\mathsf{1UTE}}^{(j)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(j)}, m_{b_j}^{(j)}) \quad \text{and} \quad \mathsf{ct}_{\mathsf{NCE}}^{(j)} \leftarrow \mathsf{NCE.Enc}(\mathsf{ek}, \mathsf{sk}_{\mathsf{1UTE}}^{(j)}),$$
   where $b_j = 0$ for $j > i$ and $b_j = 1$ for $j < i$.
   - If $j = i$, submit challenge messages $m_0^{(i)}, m_1^{(i)}$ to the UTE challenger to get $\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle$ and compute $(\mathsf{ct}_{\mathsf{NCE}}^{(i)}, \mathsf{st}_{\mathsf{NCE}}) \leftarrow \mathsf{NCE.Fake}(\mathsf{ek})$.

   Adversary $\mathcal{A}'$ replies with $\left|\mathsf{ct}^{(j)}\right\rangle = \left(\left|\mathsf{ct}_{\mathsf{1UTE}}^{(j)}\right\rangle, \mathsf{ct}_{\mathsf{NCE}}^{(j)}\right)$.
3. When adversary $\mathcal{A}$ outputs st, adversary $\mathcal{A}'$ submits $(\mathsf{st}, \mathsf{st}_{\mathsf{NCE}})$ to its challenger to be given to $\mathcal{B}'$.
4. On input $(\mathsf{st}, \mathsf{st}_{\mathsf{NCE}})$ and $\mathsf{sk}_{\mathsf{1UTE}}^{(i)}$, adversary $\mathcal{B}'$ computes $\mathsf{dk}^* \leftarrow \mathsf{NCE.Reveal}(\mathsf{st}_{\mathsf{NCE}}, \mathsf{sk}_{\mathsf{1UTE}}^{(i)})$, gives $(\mathsf{st}, \mathsf{dk}^*)$ to $\mathcal{B}$, and outputs whatever $\mathcal{B}$ outputs.

Clearly $(\mathcal{A}', \mathcal{B}')$ is QPT since $(\mathcal{A}, \mathcal{B})$ is. If the UTE challenger encrypts $m_0^{(i)}$, adversary $(\mathcal{A}', \mathcal{B}')$ perfectly simulates $\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B})$. If the UTE challenger encrypts $m_1^{(i)}$, adversary $(\mathcal{A}', \mathcal{B}')$ perfectly simulates $\mathsf{Hyb}''_{i-1}(\mathcal{A}, \mathcal{B})$. Thus, $(\mathcal{A}', \mathcal{B}')$ has advantage δ' in the indistinguishability security game, a contradiction. □

Claim 3.21. Suppose Π_{NCE} satisfies non-committing security for QPT adversaries. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}''_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr[\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B}) = 1]\right| = \mathsf{negl}(\lambda).$$

Proof. Analogous to the proof of Claim 3.19. □

Combining Claims 3.19 to 3.21 proves the lemma by a hybrid argument. □

The theorem follows by a hybrid argument via Lemma 3.18. □

Remark 3.22 (NCE with Quantum Ciphertexts). The proof of security is analogous even if the SK-NCER scheme has quantum ciphertexts, so long as the state produced by Fake is classical.

Remark 3.23 (Second-Stage Encryption Queries). With separate encryption and decryption keys, an adversary obtaining the decryption key in the second-stage cannot run encryption. Thus, it is natural to consider a notion where the second-stage adversary has access to an encryption oracle. The above proof can be easily generalized to handle this, since SK-NCER allows the adversary to make encryption queries.

Remark 3.24 (Generalizing to PKE and ABE). The above result generalizes to the setting of public-key encryption (PKE) and attribute-based encryption (ABE) by replacing SK-NCER with non-committing PKE and non-committing ABE, which can be constructed from PKE and indistinguishability obfuscation, respectively (see [HMNY21]).

3.4 Impossibility of Hyper-Efficient Shadow Tomography

In this section, we show an efficient attack on any collusion-resistant UTE that succeeds with overwhelming probability from hyper-efficient shadow tomography (HEST). In particular, this implies the impossibility of HEST for general mixed states assuming only pseudorandom states.

Theorem 3.25 (HEST Breaks t-Copy Secure UTE)**.** *Suppose there exists an algorithm $\mathcal{T}$ that satisfies Definition 3.7 for all mixed quantum states with number of copies $k = k(n, \log M, 1/\varepsilon, \log(1/\delta))$ for polynomial k. Then, there does not exist k-copy secure untelegraphable encryption that also satisfies correctness.*

Proof. Let $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with $\mathrm{poly}(\lambda)$ be a candidate correct UTE scheme with message space $\{0,1\}$ (without loss of generality) and decryption key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda\in\mathbb{N}}$. Let $n = n(\lambda)$ be the maximum size of a ciphertext, $M = |\mathcal{K}_\lambda|$, $\varepsilon < 1/2$ be a constant, and $\delta = 2^{-\lambda}$. We define the quantum circuit $E_\lambda(i,\rho)$ to output $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle) \in \{0,1\}$ where the input is parsed as $i = \mathsf{dk}$ and $\rho = |\mathsf{ct}\rangle$. We now construct an adversary $(\mathcal{A}, \mathcal{B})$ that breaks k-copy security of Π_{UTE}:

1. On input $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$, adversary $\mathcal{A}$ runs $\mathcal{T}(E_\lambda, \{|\mathsf{ct}_i\rangle\}_{i\in[k]})$ to get a classical description of a quantum circuit C, which $\mathcal{A}$ outputs as its classical state st.
2. On input dk and $\mathsf{st} = C$, adversary $\mathcal{B}$ outputs 1 if $C(\mathsf{dk}) > 1/2$ and 0 otherwise.

By definition of UTE, $n = \mathrm{poly}(\lambda)$ and $\log M = \mathrm{poly}(\lambda)$. By the choice of δ, $\log(1/\delta) = \lambda$ and $1/\varepsilon = 3$. Thus, $\mathcal{T}$ runs in QPT, which implies C is polynomial size, and thus $(\mathcal{A}, \mathcal{B})$ runs in QPT. Since $\Pr[|C(i) - \Pr[E(i,\rho) = 1]| \le \varepsilon] \ge 1 - \delta$, by the setting of ε, δ, we have

$$\Pr[|C(i) - \Pr[E(i,\rho) = 1]| < 1/2] \ge 1 - 2^{-\lambda}.$$

By correctness of UTE, we have $\Pr[E(i,\rho) = 1] = \mathsf{negl}(\lambda)$ when $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$ are encryptions of 0 and $\Pr[E(i,\rho) = 1] \ge 1 - \mathsf{negl}(\lambda)$ when $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$ are encryptions of 1. This implies $C(\mathsf{dk}) > 1/2$ for encryptions of 1 and $C(\mathsf{dk}) < 1/2$ for encryptions of 0 with probability $1 - 2^{-\lambda}$. Thus, $(\mathcal{A}, \mathcal{B})$ wins the k-copy security game with overwhelming probability. □

Corollary 3.26 (Impossibility of HEST from SK-NCER)**.** *Assuming the existence of SK-NCER (Definition 3.4), hyper-efficient shadow tomography for all mixed quantum states does not exist.*

Proof. Follows immediately by Theorems 3.16, 3.17 and 3.25 and the fact that collusion-resistant security implies t-copy security (see Definition 2.6). □

Corollary 3.27 (Impossiblity of HEST from OWFs)**.** *Assuming the existence of one-way functions, hyper-efficient shadow tomography for all mixed quantum states does not exist.*

Proof. Follows immediately by Theorem 3.5 and Corollary 3.26. □

3.5 Impossibility of HEST from Pseudorandom State Generators

In this section, we show how to weaken the assumption needed to assert the impossibility of hyper-efficient shadow tomography to just pseudorandom state generators, which are believed to be possibly weaker than one-way functions [Kre21]. Specifically, we show that the attack in Theorem 3.25 will still work with similar parameters on a bounded collusion-resistant UTE scheme, so long as the sizes of the ciphertext and decryption key is *polylogarithmic* in the collusion bound q. To achieve a bounded collusion-resistant UTE scheme with the required succinctness property, we first construct query-bounded CPA-secure SKE with quantum ciphertexts (QSKE) from a PRFS generator with a polynomial sized domain (which is implied by a PRS generator). Then, as a corollary, we get query-bounded SK-NCER with quantum ciphertexts and bounded collusion-resistant UTE from PRS generators, where the succinctness properties will be retained.

Query-Bounded CPA-Secure QSKE. We start by formally constructing a query-bounded CPA-secure secret-key encryption with quantum ciphertexts from pseudo-random function-like state (PRFS) generators (Definition 3.3). We show correctness, security, and analyze the succinctness of the ciphertext and decryption key.

Construction 3.28 (Query-Bounded CPA-Secure QSKE)**.** Let λ be a security parameter and q be a query bound. Let $G = G_{\lambda,q}$ be a PRFS generator with input length $d = d(\lambda, q)$ and output length $n = n(\lambda)$. We construct our QSKE scheme $\Pi_{\mathsf{SKE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Gen}(1^\lambda, 1^q)$: On input the security parameter λ, the key generation algorithm samples $k_1, \ldots, k_\lambda \xleftarrow{R} \{0,1\}^\lambda$ and outputs $\mathsf{sk} = (k_1, \ldots, k_\lambda)$.
- $\mathsf{Enc}(\mathsf{sk}, m)$: On input the secret key $\mathsf{sk} = (k_1, \ldots, k_\lambda)$ and a message $m \in \{0,1\}$, the encryption algorithm samples $s_1, \cdots, s_\lambda$ uniformly from $\{0,1\}$ such that $s_1 \oplus \cdots \oplus s_\lambda = m$ and samples $r_1, \ldots, r_\lambda \xleftarrow{R} \{0,1\}^{d-1}$. Compute $\rho_i \leftarrow G(k_i, (r_i, s_i))$ for $i \in [\lambda]$ and output $|\mathsf{ct}\rangle = (\{r_i, \rho_i\}_{i\in[\lambda]})$.
- $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$: On input the secret key $\mathsf{sk} = (k_1, \ldots, k_\lambda)$ and a quantum ciphertext $|\mathsf{ct}\rangle = (\{r_i, \rho_i\}_{i\in[\lambda]})$, the decryption algorithm runs $\mathsf{Test}(k_i, (r_i, 0), \rho_i)$ to get s_i for $i \in [\lambda]$. It outputs $m = s_1 \oplus \cdots \oplus s_\lambda$.

Theorem 3.29 (Correctness)**.** *If $n = \omega(\log \lambda)$, then Construction 3.28 is correct.*

Proof. Fix $\lambda \in \mathbb{N}$ and $m \in \{0,1\}$. Let $\mathsf{sk} = (k_1, \ldots, k_\lambda) \leftarrow \mathsf{Gen}(1^\lambda, 1^q)$ and

$$|\mathsf{ct}\rangle = (\{r_i, \rho_i\}_{i\in[\lambda]}) \leftarrow \mathsf{Enc}(\mathsf{sk}, m).$$

We consider the output of $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$. By definition of Test (see Definition 3.3), the output of $\mathsf{Test}(k_i, (r_i, 0), \rho_i)$ determines if $s_i = 0$ up to negligible error, meaning the derived s_i is correct with overwhelming probability. By the distribution of $s_1, \cdots, s_\lambda$, we have $m = s_1 \oplus \cdots \oplus s_\lambda$ if the s_i are correct. By a union bound, this holds with overwhelming probability, as desired. □

Theorem 3.30 (Query-Bounded CPA Security)**.** *Let λ be a security parameter and $q = q(\lambda)$ be any polynomial. Suppose $d \geq 2\log q + 1$ and G is a selectively secure PRFS generator. Then, Construction 3.28 is CPA-secure for up to q queries.*

Proof. We prove the theorem by showing encryptions of 0 and 1 are computationally indistinguishable. Suppose there exists a QPT adversary $\mathcal{A}$ that has non-negligble advantage δ in the query-bounded CPA security game. We define a sequence of hybrid experiments:

- $\mathsf{Hyb}_0^{(b)}$: This is the CPA security game with challenge bit b. In particular:
  - At the beginning of the game, the challenger samples $k_1, \ldots, k_\lambda \xleftarrow{R} \{0,1\}^\lambda$.
  - When adversary $\mathcal{A}$ makes the jth encryption query $m_0^{(j)}, m_1^{(j)} \in \{0,1\}$ for some $j \in [q]$, the challenger samples $s_1^{(j)}, \ldots, s_\lambda^{(j)}$ uniformly from $\{0,1\}$ such that $s_1^{(j)} \oplus \cdots \oplus s_\lambda^{(j)} = m_b^{(j)}$ and samples $r_1^{(j)}, \ldots, r_\lambda^{(j)} \xleftarrow{R} \{0,1\}^{d-1}$. The challenger computes $\rho_i^{(j)} \leftarrow G(k_i, (r_i^{(j)}, s_i^{(j)}))$ for $i \in [\lambda]$ and gives $\left|\mathsf{ct}^{(j)}\right\rangle = (\{r_i^{(j)}, \rho_i^{(j)}\}_{i\in[\lambda]})$ to $\mathcal{A}$.

– After q queries, adversary $\mathcal{A}$ outputs a bit $b' \in \{0,1\}$, which is the output of the experiment.

- $\mathsf{Hyb}_1^{(b)}$: Same as $\mathsf{Hyb}_0^{(b)}$ except the challenger now samples $r_i^{(j)} \xleftarrow{R} \{0,1\}^{d-1}$ for $i \in [\lambda], j \in [q]$ at the start of the game and outputs $\perp$ if for all $i \in [\lambda]$, there exists $j, j' \in [q]$ such that $j \neq j'$ and $r_i^{(j)} = r_i^{(j')}$.
- $\mathsf{Hyb}_2^{(b)}$: Same as $\mathsf{Hyb}_1^{(b)}$ except the challenger samples $\rho_{i^*}^{(j)} \leftarrow \mu_n$ for all $j \in [q]$, where $i^* \in [\lambda]$ is the first index such that $r_{i^*}^{(1)}, \ldots, r_{i^*}^{(q)}$ are distinct.

We write $\mathsf{Hyb}_i^{(b)}(\mathcal{A})$ to denote the output distribution of an execution of $\mathsf{Hyb}_i^{(b)}$ with adversary $\mathcal{A}$. We now argue that each adjacent pair of distributions are indistinguishable.

Lemma 3.31. *Suppose $d \geq 2\log q + 1$. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_0^{(b)}(\mathcal{A}) = 1\right] - \Pr\left[\mathsf{Hyb}_1^{(b)}(\mathcal{A}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. First, note that sampling $r_i^{(j)}$ at the start is purely syntactic, since $r_i^{(j)}$ is independently sampled from each $m_b^{(j)}$. We show the lemma by showing that the probability that the challenger outputs $\perp$ at the start of the game is negligible, as this is the only other difference between the experiments. Since $d \geq 2\log q + 1$, the set $\{0,1\}^{d-1}$ has size $2^{d-1} \geq 2^{2\log q} = q^2$. This implies that for each index $i \in [\lambda]$, the probability that there exists $j, j' \in [q]$ such that $j \neq j'$ and $r_i^{(j)} = r_i^{(j')}$ is at most

$$1 - \left(1 - \frac{q-1}{q^2}\right)^q \leq 1 - 1/e.$$

Since $r_i^{(1)}, \ldots, r_i^{(q)}$ are sampled independently for each $i \in [\lambda]$, the probability of a collision for all $i \in [\lambda]$ is at most $(1-1/e)^\lambda$, which is negligible. □

Lemma 3.32. *Suppose G is a selectively secure PRFS generator. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_1^{(b)}(\mathcal{A}) = 1\right] - \Pr\left[\mathsf{Hyb}_2^{(b)}(\mathcal{A}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $\mathcal{A}$ distinguishes the experiments with non-negligible probability δ'. We use $\mathcal{A}$ to construct an adversary $\mathcal{B}$ that breaks PRFS security as follows:

1. Adversary $\mathcal{B}$ starts by sampling $r_i^{(j)} \xleftarrow{R} \{0,1\}^{d-1}$ for $i \in [\lambda], j \in [q]$ and checking the abort condition. If it does not abort, it finds the first index $i^* \in [\lambda]$ such that $r_{i^*}^{(1)}, \ldots, r_{i^*}^{(q)}$ are distinct.
2. Adversary $\mathcal{B}$ samples $s_{i^*}^{(j)} \xleftarrow{R} \{0,1\}$ for $j \in [q]$ and submits the queries $\{(r_{i^*}^{(j)}, s_{i^*}^{(j)})\}_{j\in[q]}$ to the PRFS challenger to get back $\rho_{i^*}^{(j)}$ for $j \in [q]$.
3. Adversary $\mathcal{B}$ samples $k_i \xleftarrow{R} \{0,1\}^\lambda$ for $i \neq i^*$. When advervsary $\mathcal{A}$ makes its jth encryption query $m_0^{(j)}, m_1^{(j)} \in \{0,1\}$, the $\mathcal{B}$ samples $s_1^{(j)}, \ldots, s_{i^*-1}^{(j)}, s_{i^*+1}^{(j)}, \ldots, s_\lambda^{(j)}$ uniformly from $\{0,1\}$ such that

$$s_1^{(j)} \oplus \cdots s_{i^*-1}^{(j)} \oplus s_{i^*+1}^{(j)} \cdots \oplus s_\lambda^{(j)} = m_b^{(j)} \oplus s_{i^*}^{(j)}.$$

   Adversary $\mathcal{B}$ computes $\rho_i^{(j)} \leftarrow G(k_i, (r_i^{(j)}, s_i^{(j)}))$ for $i \in [\lambda] \setminus \{i^*\}$ and gives $\left|\mathsf{ct}^{(j)}\right\rangle = (\{r_i^{(j)}, \rho_i^{(j)}\}_{i\in[\lambda]})$ to $\mathcal{A}$.
4. At the end of the game, adversary $\mathcal{B}$ outputs whatever $\mathcal{A}$ outputs.

Clearly adversary $\mathcal{B}$ is QPT if $\mathcal{A}$ is. If $\rho_{i^*}^{(j)}$ are Haar-random states for $j \in [q]$, adversary $\mathcal{B}$ simulates $\mathsf{Hyb}_2^{(b)}(\mathcal{A})$. If $\rho_{i^*}^{(j)}$ are PRFS generator outputs for $j \in [q]$, adversary $\mathcal{B}$ simulates $\mathsf{Hyb}_1^{(b)}(\mathcal{A})$. Thus, adversary $\mathcal{B}$ has advantage δ' in the PRFS security game, a contradiction. □

Lemma 3.33. *The experiments* $\mathsf{Hyb}_2^{(0)}(\mathcal{A})$ *and* $\mathsf{Hyb}_2^{(1)}(\mathcal{A})$ *are identically distributed.*

Proof. Since for all $j \in [q]$ the state $\rho_{i^*}^{(j)}$ no longer depends on $s_{i^*}^{(j)}$, each $s_i^{(j)}$ for $i \neq i^*$ is uniform and independent of b, meaning the adversary's view in the two experiments is identical. □

Combining Lemmata 3.31 to 3.33, the theorem follows by a hybrid argument. □

Theorem 3.34 (Succinct Ciphertexts)**.** *Let* $\lambda \in \mathbb{N}$ *be a security parameter and* $q \in \mathbb{N}$ *be a query bound. Then, the size of a ciphertext in Construction 3.28 is* $\mathrm{poly}(\lambda, \log q)$ *and the size of the secret key is* $\mathrm{poly}(\lambda)$.

Proof. For Theorem 3.29, the parameter requirement is $n = \omega(\log \lambda)$ and for Theorem 3.30 the parameter requirement is $d \geq 2\log q + 1$. Thus, setting $n = \lambda$ and $d = 2\log q + 1$ suffices. A ciphertext consists of λ pairs of $r \in \{0,1\}^{d-1}$ and ρ which is the output of the PRFS. Thus, the total number of qubits in a ciphertext is $\lambda(2\log q + \lambda) = 2\lambda \log q + \lambda^2$, as desired. Moreover, the key size of a PRFS generator can be made independent of the domain size, so the size of a secret key is $\mathrm{poly}(\lambda)$. □

Corollary 3.35 (Succinct Bounded-Query SK-NCER)**.** *Let* $\lambda \in \mathbb{N}$ *be a security parameter and* $q(\lambda)$ *be any polynomial query bound. If PRS generators exist, then there is a* q*-bounded-query SK-NCER scheme with* $\mathrm{poly}(\lambda, \log q)$*-sized quantum ciphertexts and* $\mathrm{poly}(\lambda)$*-sized decryption keys.*

Proof. Follows by Theorems 3.29, 3.30 and 3.34 and Corollary 3.6. □

Corollary 3.36 (Succinct Bounded Collusion-Resistant UTE)**.** *Let* $\lambda \in \mathbb{N}$ *be a security parameter and* $q(\lambda)$ *be any polynomial query bound. If PRS generators exist, then there is a* q*-bounded collusion-resistant UTE scheme with* $\mathrm{poly}(\lambda, \log q)$*-sized ciphertexts and* $\mathrm{poly}(\lambda)$*-sized decryption keys.*

Proof. Follows by considering the analog of Construction 3.15 and the proofs of Theorems 3.16 and 3.17 in the bounded collusion-resistant setting, along with Corollary 3.35. The size of a ciphertext is the sum of the sizes of a one-time UTE ciphertext and the SK-NCER ciphertext. The size of the one-time UTE ciphertext is $\mathrm{poly}(\lambda)$ (since it does not have a bound parameter), and by Corollary 3.35 the size of the SK-NCER ciphertext is $\mathrm{poly}(\lambda, \log q)$. The size of a decryption key is the same as the underlying SK-NCER scheme, which is $\mathrm{poly}(\lambda)$. □

Impossibility of HEST. We now show an attack on a succinct bounded q-copy secure UTE from hyper-efficient shadow tomography with number of copies k.

Theorem 3.37 (HEST Breaks Bounded q-Copy Secure UTE)**.** *Let* λ *be a security parameter. Suppose there exists an algorithm* $\mathcal{T}$ *that satisfies Definition 3.7 for all mixed quantum states with number of copies that is upperbounded by* $k = k(n, \log M, 1/\varepsilon, \log(1/\delta))$ *for polynomial* k*. Then, there exists* $q = \mathrm{poly}(\lambda)$ *such that bounded* q*-copy secure untelegraphable encryption with succinct ciphertexts and decryption keys does not exist.*

Proof. The attack is analogous to the one in Theorem 3.25. Thus, we just show that that parameters can be satisfied. Recall that n corresponds to the number of qubits in a ciphertext. If the ciphertexts or keys are not succinct, there is a circularity issue: namely, k grows with $\mathrm{poly}(n, \log M) = \mathrm{poly}(q)$, so q may always be less than k. However, when ciphertexts and keys are succinct, we can set the parameters such that this circularity is not an issue. Specifically, fixing any $\lambda \in \mathbb{N}$ and letting $k_0 = \mathrm{poly}(\lambda, \log q)$ be the value of k when $q = \lambda$, we set $q = \lambda k_0^2$, which will correspondingly bound k by $k_0 \cdot \mathrm{polylog}(k_0)$. This means the UTE scheme can support k-copy security when $q \geq \lambda k_0^2$, which we know is $\mathrm{poly}(\lambda)$ by definition of k_0. □

Corollary 3.38 (Impossiblity of HEST from PRS Generators)**.** *Assuming the existence of pseudorandom state generators, hyper-efficient shadow tomography for all mixed quantum states does not exist.*

Proof. The corollary follows by Theorem 3.37 and Corollary 3.36. □

4 Separating Untelegraphable Encryption and Unclonable Encryption

In this section, we construct an untelegraphable encryption scheme that satisfies UTE security, but simultaneously has an adversary that wins the UE security game with non-negligible advantage for an *unbounded* polynomial number of second-stage adversaries. The construction is secure under the LWE assumption in the classical oracle model. In the classical oracle model, we assume algorithms can generate oracles to classical functions efficiently so long as the function has a polynomial-size description. This can be thought of as an extension of ideal obfuscation. Note that for any fixed bound n on the number of second-stage adversaries, one can construct n-clonable UTE trivially from collusion-resistant UTE, by defining encryption to generate n fresh UTE ciphertexts.

4.1 Building Blocks

In this section, we describe notions that are needed for our construction of unbounded clonable UTE.

One-shot MAC. We introduce the notion of a one-shot message authentication code (OSMAC). An OSMAC functions similar to a standard MAC, except users now generate a one-time quantum signing key, along with a classical verification key. The security notion states that it is computationally hard for an adversary to generate two signatures of two different messages under any single verification key that is generated by the adversary. One-shot MACs are a relaxation of one-shot signatures [AGKZ20], which are publicly verifiable, and two-tier one-shot signatures [MPY24], which have partial public verification. In [MPY24], two-tier one-shot signatures are constructed from the standard learning with errors (LWE) assumption [Reg05].

Definition 4.1 (One-shot MAC)**.** A one-shot message authentication code (OSMAC) scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{OSM}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Vrfy})$ with the following syntax:

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{mvk})$: On input the security parameter λ, the setup algorithm outputs the public parameters pp and a master verification key mvk.
- $\mathsf{KeyGen}(\mathsf{pp}) \to (|\mathsf{sk}\rangle, \mathsf{vk})$: On input the public parameters pp, the key generation algorithm outputs a quantum signing key $|\mathsf{sk}\rangle$ along with a verification key vk.
- $\mathsf{Sign}(|\mathsf{sk}\rangle, m) \to \sigma$: On input a quantum signing key $|\mathsf{sk}\rangle$ and a message $m \in \mathcal{M}_\lambda$, the signing algorithm outputs a signature σ.
- $\mathsf{Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}, m, \sigma) \to b$: On input the public parameters pp, the master verification key mvk, a verification key vk, a message $m \in \mathcal{M}_\lambda$, and a signature σ, the verification algorithm outputs a bit $b \in \{0,1\}$.

We require that Π_{OSM} satisfy the following properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}_\lambda$,
$$\Pr\left[\mathsf{Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}, m, \sigma) = 1 : \begin{array}{c}(\mathsf{pp}, \mathsf{mvk}) \leftarrow \mathsf{Setup}(1^\lambda)\\ (|\mathsf{sk}\rangle, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp})\\ \sigma \leftarrow \mathsf{Sign}(|\mathsf{sk}\rangle, m)\end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$
- **Two-Signature Security:** For all QPT adversaries $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,
$$\Pr\left[\begin{array}{c}m_0 \neq m_1 \ \wedge \mathsf{Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}, m_0, \sigma_0) = 1\\ \wedge \mathsf{Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}, m_1, \sigma_1) = 1\end{array} : \begin{array}{c}(\mathsf{pp}, \mathsf{mvk}) \leftarrow \mathsf{Setup}(1^\lambda)\\ (\mathsf{vk}, m_0, \sigma_0, m_1, \sigma_1) \leftarrow \mathcal{A}(\mathsf{pp})\end{array}\right] \leq \mathsf{negl}(\lambda).$$

Theorem 4.2 (Two-tier one-shot signature [MPY24, Theorem 3.2])**.** *Assuming the quantum hardness of LWE, there exists a two-tier one-shot signature scheme with message space* $\{0,1\}$.

Theorem 4.3 (One-shot MAC)**.** *Assuming the quantum hardness of LWE, there exists a one-shot MAC scheme with message space* $\{0,1\}^{\mathrm{poly}(\lambda)}$ *for any fixed polynomial* $\mathrm{poly}(\cdot)$.

Proof. A scheme with message space $\{0,1\}$ follows immediately from Theorem 4.2, and extending to multi-bit messages follows by running many single-bit OSMAC schemes in parallel, where each scheme corresponds to a message bit. □

4.2 Constructing Unbounded Clonable Untelegraphable Encryption

In this section, we describe our construction of an unbounded clonable UTE scheme assuming the existence of an OSMAC in the classical oracle model. In the classical oracle model, we will assume algorithms can efficiently generate functions as black-boxes, meaning an adversary will only have oracle access to said function in the security proof. Our untelegraphable encryption construction is again in the setting where there are separate encryption and decryption keys, where only the decryption key is revealed to the second stage adversary.

Construction 4.4 (Unbounded Clonable UTE)**.** Let λ be a security parameter, $n = n(\lambda)$ be length parameter, and $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ be a message space. Let $\Pi_{\mathsf{OSM}} = (\mathsf{OSM.Setup}, \mathsf{OSM.KeyGen}, \mathsf{OSM.Sign}, \mathsf{OSM.Vrfy})$ be an OSMAC scheme with message space $\{0,1\}^{\ell(\lambda)}$. We construct our unbounded clonable UTE scheme $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Gen}(1^\lambda)$: On input the security parameter λ, the key generation algorithm samples $s \xleftarrow{R} \{0,1\}^{n+\lambda}$ and $(\mathsf{pp}, \mathsf{mvk}) \leftarrow \mathsf{OSM.Setup}(1^\lambda)$. It outputs $(\mathsf{ek} = (\mathsf{pp}, \mathsf{mvk}, s), \mathsf{dk} = s)$.
- $\mathsf{Enc}(\mathsf{ek}, m)$: On input the encryption key $\mathsf{ek} = (\mathsf{pp}, \mathsf{mvk}, s)$ and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm samples $(|\mathsf{sk}_\varepsilon\rangle, \mathsf{vk}_\varepsilon) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp})$ and generates the following classical oracle $\mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}$:

  Hard-wired: master verification key mvk, public parameters pp, string $s \in \{0,1\}^{n+\lambda}$, message $m \in \mathcal{M}_\lambda$, verification key vk_ε
  Input: strings $s' \in \{0,1\}^{n+\lambda}, r \in \{0,1\}^n$, set of signature tuples $\{\mathsf{st}_v\}_{v\in V_r}$ where the set $V_r = \{\varepsilon, r_1, r[1,2], \ldots, r[1,n-1], r\}$ contains all prefixes of r including the empty string ε

  1. Parse $\mathsf{st}_\varepsilon = (\mathsf{vk}_0, \mathsf{vk}_1, \sigma_\varepsilon)$, $\mathsf{st}_v = (\mathsf{vk}_v, \mathsf{vk}_{v\|0}, \mathsf{vk}_{v\|1}, \sigma_v)$ for $v \in V_r \setminus \{\varepsilon, r\}$, $\mathsf{st}_r = (\mathsf{vk}_r, \sigma_r)$.
  2. Output m if $s = s'$, $\mathsf{OSM.Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_v, \mathsf{vk}_{v\|0}\|\mathsf{vk}_{v\|1}, \sigma_v) = 1$ for all $v \in V_r \setminus \{r\}$, and $\mathsf{OSM.Vrfy}(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_r, s', \sigma_r) = 1$. Otherwise, output $\perp$.

  Figure 1: Classical oracle $\mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}$

  The encryption algorithm outputs the ciphertext $|\mathsf{ct}\rangle = (|\mathsf{sk}_\varepsilon\rangle, \mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}, \mathsf{pp})$.
- $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle)$: On input the decryption key dk and a quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{sk}_\varepsilon\rangle, \mathcal{O}, \mathsf{pp})$, the decryption algorithm does the following:

  1. For all $v \in V_{0^n} \setminus \{\varepsilon\}$ sample

     $$(|\mathsf{sk}_v\rangle, \mathsf{vk}_v) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp}) \text{ and } (|\mathsf{sk}_{v\oplus 1}\rangle, \mathsf{vk}_{v\oplus 1}) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp}),$$

     where $v \oplus 1$ is v with its last bit flipped for a variable length bitstring v.
  2. Compute the signatures $\sigma_v \leftarrow \mathsf{OSM.Sign}(|\mathsf{sk}_v\rangle, \mathsf{vk}_{v\|0}\|\mathsf{vk}_{v\|1})$ for all $v \in V_{0^n} \setminus \{0^n\}$ and $\sigma_{0^n} \leftarrow \mathsf{OSM.Sign}(|\mathsf{sk}_{0^n}\rangle, \mathsf{dk})$.
  3. Set $\mathsf{st}_\varepsilon = (\mathsf{vk}_0, \mathsf{vk}_1, \sigma_\varepsilon)$, $\mathsf{st}_v = (\mathsf{vk}_v, \mathsf{vk}_{v\|0}, \mathsf{vk}_{v\|1}, \sigma_v)$ for $v \in V_{0^n} \setminus \{\varepsilon, 0^n\}$, $\mathsf{st}_{0^n} = (\mathsf{vk}_{0^n}, \sigma_{0^n})$, and output $\mathcal{O}(\mathsf{dk}, 0^n, \{\mathsf{st}_v\}_{v\in V_{0^n}})$.

Theorem 4.5 (Correctness)**.** *Suppose Π_{OSM} is correct. Then, Construction 4.4 is correct.*

Proof. Fix any $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}_\lambda$. Let $(\mathsf{ek} = (\mathsf{pp}, \mathsf{mvk}, s), \mathsf{dk} = s) \leftarrow \mathsf{Gen}(1^\lambda)$ and $|\mathsf{ct}\rangle = (|\mathsf{sk}_\varepsilon\rangle, \mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}, \mathsf{pp}) \leftarrow \mathsf{Enc}(\mathsf{ek}, m)$. We consider the output of $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle)$. By inspection, Dec honestly signs the same MAC chain checked by $\mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}$ (Figure 1) for $r = 0^n$. Thus, by correctness of Π_{OSM}, we have $m \leftarrow \mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}(\mathsf{dk}, 0^n, \{\mathsf{st}_v\}_{v\in V_{0^n}})$ with overwhelming probability, as desired. □

Theorem 4.6 (Indistinguishability Security)**.** *Let $\lambda \in \mathbb{N}$ be a security parameter and suppose $n = n(\lambda) = \Omega(\log \lambda)$, $n = o(\lambda)$, and Π_{OSM} satisfies two-signature security for adversaries that run in time $2^n \cdot \mathrm{poly}(\lambda)$ and have advantage at most 2^{-n^2}. Then, Construction 4.4 satisfies one-time indistinguishability security in the classical oracle model.*

Proof. Suppose there exists a QPT adversary $(\mathcal{A}, \mathcal{B})$ such that $\mathcal{A}$ and $\mathcal{B}$ make exactly q queries (without loss of generality) and wins the UTE indistinguishability security game with non-negligible advantage $\delta > 0$. Let $\mathcal{O}_\perp$ be an oracle which always outputs $\perp$. Now, consider an adversary $\hat{\mathcal{A}}$ which is given an oracle $\mathcal{O}$ that is either $\mathcal{O}_\perp$ or $\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}$ (Figure 1) for challenge bit $b \in \{0,1\}$, along with an input $(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon)$. Adversary $\hat{\mathcal{A}}$ runs the UTE security game except that it uses $\mathcal{O}$ to simulate $\mathcal{A}$'s oracle queries and the secret information $(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon)$ to simulate $\mathcal{B}$'s oracle queries. Note that $\hat{\mathcal{A}}^{\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}}(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon)$ perfectly simulates the UTE game with challenge bit b. Setting $\mathcal{B}_{\mathsf{O2H}}$ to the an algorithm that measures a uniformly random query of $\hat{\mathcal{A}}$, by Lemma 2.1, we have

$$\begin{aligned}&\left|\Pr\left[\hat{\mathcal{A}}^{\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}}(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon) = 1\right] - \Pr\left[\hat{\mathcal{A}}^{\mathcal{O}_\perp}(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon) = 1\right]\right| \\ &\leq 2q \cdot \sqrt{\Pr\left[\mathcal{B}^{\mathcal{O}_\perp}_{\mathsf{O2H}}(\mathsf{mvk}, s, m_b, \mathsf{vk}_\varepsilon) \in S\right]} = \nu(\lambda),\end{aligned}$$

where S is the set of valid tuples $(s, r, \{\mathsf{st}_v\}_{v \in V_r})$ that makes $\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}$ output m_b. Since s is uniform and independent of $\mathcal{A}$'s view, it must be that ν is negligible, since $q = \mathrm{poly}(\lambda)$. By definition of $\hat{\mathcal{A}}$, this implies

$$\Pr_{b,\mathsf{pp},\mathsf{mvk},s,\mathsf{vk}_\varepsilon,|\mathsf{sk}_\varepsilon\rangle}\left[\mathcal{B}^{\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}}(\mathsf{st}, s) = b : \mathsf{st} \leftarrow \mathcal{A}^{\mathcal{O}_\perp}(\mathsf{pp}, |\mathsf{sk}_\varepsilon\rangle)\right] \geq \frac{1}{2} + \delta - \nu(\lambda),$$

where $b \xleftarrow{R} \{0,1\}$, $s \xleftarrow{R} \{0,1\}^{n+\lambda}$, $(\mathsf{pp}, \mathsf{mvk}) \leftarrow \mathsf{OSM.Setup}(1^\lambda)$, $(|\mathsf{sk}_\varepsilon\rangle, \mathsf{vk}_\varepsilon) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp})$, as in the UTE security game with a random challenge bit b. We can rewrite the above expression as

$$\mathop{\mathbb{E}}_{\mathsf{pp},\mathsf{mvk},\mathsf{vk}_\varepsilon,\mathsf{st}}\left[\Pr_{b,s}\left[\mathcal{B}^{\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\epsilon}}(\mathsf{st}, s) = b\right]\right] \geq \frac{1}{2} + \delta - \nu(\lambda), \tag{1}$$

where $\mathsf{st} \leftarrow \mathcal{A}^{\mathcal{O}_\perp}(\mathsf{pp}, |\mathsf{sk}_\varepsilon\rangle)$. Let a tuple $(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_\varepsilon, \mathsf{st})$ be "good" if we have

$$\Pr_{b,s}\left[\mathcal{B}^{\mathcal{O}_{\mathsf{ek},m_b,\mathsf{vk}_\varepsilon}}(\mathsf{st}, s) = b\right] \geq \frac{1}{2} + \frac{\delta}{2} - \nu(\lambda).$$

By Equation (1), a tuple $(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_\varepsilon, \mathsf{st})$ is good with probability at least $\delta/2$. Furthermore, for any $(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_\varepsilon, \mathsf{st})$, we have

$$\Pr_{b,s}\left[\mathcal{B}^{\mathcal{O}_\perp}(\mathsf{st}, s) = b\right] = \frac{1}{2}. \tag{2}$$

Fix a good tuple $(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_\varepsilon, \mathsf{st})$, let S be the same set as above, and let $\mathcal{B}_{\mathsf{ext}}$ be an algorithm that measures a uniform query of $\mathcal{B}$. Then, by Lemma 2.1, the definition of a good tuple, and Equation (2), we have

$$\Pr_{b,s}\left[\mathcal{B}^{\mathcal{O}_\perp}_{\mathsf{ext}}(\mathsf{st}, s, b) \in S\right] \geq (\delta/4q - \nu(\lambda)/2q)^2 = \delta_{\mathsf{ext}}(\lambda). \tag{3}$$

We now construct an OSMAC adversary $\mathcal{A}_{\mathsf{OSM}}$ as follows:

1. On input pp, adversary $\mathcal{A}_{\mathsf{OSM}}$ samples $(|\mathsf{sk}_\varepsilon\rangle, \mathsf{vk}_\varepsilon) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp})$ and runs $\mathsf{st} \leftarrow \mathcal{A}^{\mathcal{O}_\perp}(\mathsf{pp}, |\mathsf{sk}_\varepsilon\rangle)$.
2. Adversary $\mathcal{A}_{\mathsf{OSM}}$ initializes a list L, and for each $i \in [(2^n + 1) \cdot \lambda/\delta_{\mathsf{ext}}(\lambda)]$, it does the following:
   (a) Sample $s_i \xleftarrow{R} \{0,1\}^{n+\lambda}$ and $b_i \xleftarrow{R} \{0,1\}$.
   (b) Run $(s', r_i, (\mathsf{st}_v)_{v \in V_{r_i}}) \leftarrow \mathcal{B}^{\mathcal{O}_\perp}_{\mathsf{ext}}(\mathsf{st}, s_i, b_i)$, and add $(s', r_i, (\mathsf{st}_v)_{v \in V_{r_i}})$ to L if $s' = s_i$.
3. Adversary $\mathcal{A}_{\mathsf{OSM}}$ randomly chooses two elements $(s_j, r_j, (\mathsf{st}_v)_{v \in V_{r_j}})$ and $(s_k, r_k, (\mathsf{st}_v)_{v \in V_{r_k}})$ of L. If $s_j = s_k$ or $r_j \neq r_k$, $\mathcal{A}_{\mathsf{OSM}}$ outputs $\perp$. Otherwise, $\mathcal{A}_{\mathsf{OSM}}$ outputs the first collision $(\mathsf{vk}_v, m, \sigma_v, m', \sigma'_v)$ it finds among the two tuples.

Adversary $\mathcal{A}_{\mathsf{OSM}}$ runs in $2^n \cdot \mathrm{poly}(\lambda)$ time for infinitely many $\lambda \in \mathbb{N}$, since δ is non-negligible and $(\mathcal{A}, \mathcal{B})$ runs in QPT. Furthermore, assuming $(\mathsf{pp}, \mathsf{mvk}, \mathsf{vk}_\varepsilon, \mathsf{st})$ is a good tuple, Equation (3) and the number of iterations ensure that L contains $2^n + 1$ valid MAC chains $(s_i, r_i, (\mathsf{st}_v)_{v \in V_{r_i}})$ with $1 - 2^{-O(\lambda)}$ probability. By the pigeonhole principle, there exists $(s_j, r_j, (\mathsf{st}_v)_{v \in V_{r_j}})$ and $(s_k, r_k, (\mathsf{st}_v)_{v \in V_{r_k}})$ among those $2^n + 1$ valid MAC chains such that $r_j = r_k$, and with overwhelming probability we have $s_j \neq s_k$ for all items in L since $s_i \xleftarrow{R} \{0,1\}^{n+\lambda}$ in each iteration. Thus, such a pair of MAC chains must have a collision where different signatures are given on the same verification key. Specifically, if vk_{r_j} is not the verification key for the collision, then $\mathsf{vk}_{r_j} \neq \mathsf{vk}_{r_k}$, implying the chain must differ for some other $v \in V_{r_j} \setminus \{r_j\}$, since vk_ε is the same for both. The advantage of $\mathcal{A}_{\mathsf{OSM}}$ is then at least

$$\frac{\delta}{2|L|^2} \geq \frac{1}{\mathrm{poly}(\lambda) \cdot 2^{2n}} > \frac{1}{2^{n^2}},$$

for infinitely many $\lambda \in \mathbb{N}$, as desired. □

Theorem 4.7 (Attack on One-Way Unclonable Security)**.** *Suppose Π_{OSM} is correct and $n = \omega(\log \lambda)$. Then, Construction 4.4 is not k-adversary one-way unclonable secure (Definition 2.3) for any $k = \mathrm{poly}(\lambda)$ in the classical oracle model.*

Proof. We give a two stage adversary $(\mathcal{A}, \mathcal{B})$ to break one-way unclonable security for any $k = \mathrm{poly}(\lambda)$ number of second stage adversaries (all second stage adversaries run the same algorithm $\mathcal{B}$):

1. At the beginning of the game, $\mathcal{A}$ gets a single ciphertext $|\mathsf{ct}\rangle = (|\mathsf{sk}_\varepsilon\rangle, \mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}, \mathsf{pp})$ from the challenger and decides on any $k = \mathrm{poly}(\lambda)$ number of second stage adversaries.
2. For all $r \in [k]$ (where $r \in \{0,1\}^n$ is appropriately padded), $v \in V_r \setminus \{\varepsilon\}$ adversary $\mathcal{A}$ samples
$$(|\mathsf{sk}_v\rangle, \mathsf{vk}_v) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp}) \text{ and } (|\mathsf{sk}_{v\oplus 1}\rangle, \mathsf{vk}_{v\oplus 1}) \leftarrow \mathsf{OSM.KeyGen}(\mathsf{pp}).$$
3. For all $r \in [k]$, $v \in V_r \setminus \{r\}$ adversary $\mathcal{A}$ computes $\sigma_v \leftarrow \mathsf{OSM.Sign}(|\mathsf{sk}_v\rangle, \mathsf{vk}_{v\|0} \| \mathsf{vk}_{v\|1})$.
4. Adversary $\mathcal{A}$ sets $\mathsf{st}_\varepsilon = (\mathsf{vk}_0, \mathsf{vk}_1, \sigma_\varepsilon)$ and for all $r \in [k]$ sets $\mathsf{st}_v = (\mathsf{vk}_v, \mathsf{vk}_{v\|0}, \mathsf{vk}_{v\|1}, \sigma_v)$ for $v \in V_r \setminus \{\varepsilon, r\}$. For the rth second stage adversary, $\mathcal{A}$ outputs $(\mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}, r, \{\mathsf{st}_v\}_{v \in V_r \setminus \{r\}}, |\mathsf{sk}_r\rangle, \mathsf{vk}_r)$ in its register.
5. On input $(r, \{\mathsf{st}_v\}_{v \in V_r \setminus \{r\}}, |\mathsf{sk}_r\rangle, \mathsf{vk}_r)$ and $\mathsf{dk} = s$, adversary $\mathcal{B}$ computes $\sigma_r \leftarrow \mathsf{OSM.Sign}(|\mathsf{sk}_r\rangle, s)$, sets $\mathsf{st}_r = (\mathsf{vk}_r, \sigma_r)$, and outputs $\mathcal{O}(s, r, \{\mathsf{st}_v\}_{v \in V_r})$.

Since k is polynomial, $(\mathcal{A}, \mathcal{B})$ is efficient. Since the MAC chain is generated honestly for each $r \in [k]$ and $\mathcal{B}$ is given the honest decryption key s, the checks in $\mathcal{O}_{\mathsf{ek},m,\mathsf{vk}_\varepsilon}$ pass with overwhelming probability by correctness of Π_{OSM}. Thus, all k second stage adversaries recover m with overwhelming probability. □

Parameter setting. Setting $n = \log^2 \lambda$, satisfies the constraint in Theorem 4.7 and correspondingly assumes quasi-polynomial security of Π_{OSM} for Theorem 4.6. This yields the following corollary:

Corollary 4.8 (Clonable UTE)**.** *Assuming the quasi-polynomial quantum hardness of LWE in the classical oracle model, there exists an indistinguishability secure UTE scheme that is not one-way unclonable secure for an unbounded polynomial number of second stage adversaries.*

5 Untelegraphable Encryption with Everlasting Security

In this section, we explore the notion of everlasting security (Definition 2.7) for untelegraphable encryption schemes. In particular, we construct collusion-resistant untelegraphable encryption with everlasting security in the quantum random oracle model (QROM). We also define the notion of weakly-efficient shadow tomography (WEST) and show that such a notion leads to an attack on the everlasting security of any correct collusion-resistant UTE scheme. Finally, we show that working in the QROM or assuming some other non-falsifiable assumption may be inherent when trying to construct UTE with everlasting security, by giving a black-box separation of public-key everlasting one-way secure UTE and any falsifiable assumption, assuming sub-exponential one-way functions exist.

5.1 Collusion-Resistant Untelegraphable Encryption with Everlasting Security

In this section, we describe our construction of collusion-resistant untelegraphable encryption with everlasting security in the quantum random oracle model (QROM), which parallels the construction of certified everlasting hiding commitments in [HMNY22]. A modification of the construction can also be used to build single-ciphertext public key unclonable encryption with everlasting security.

Construction 5.1 (Everlasting Secure UTE)**.** Let λ be a security parameter. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{SKE}} = (\mathsf{SKE.Gen}, \mathsf{SKE.Enc}, \mathsf{SKE.Dec})$ be an SKE scheme with message space $\{0,1\}^\lambda$.
- Let $\Pi_{\mathsf{1UTE}} = (\mathsf{1UTE.Gen}, \mathsf{1UTE.Enc}, \mathsf{1UTE.Dec})$ be a secret-key untelegraphable encryption scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and key space $\{0,1\}^{\ell(\lambda)}$.
- Let $\mathcal{H} = \{H_\lambda : \{0,1\}^\lambda \to \{0,1\}^\ell\}$ be a hash function family, which will be modeled as a random oracle in the security analysis.

We now construct our everlasting UE scheme $\Pi_{\mathsf{UE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Gen}(1^\lambda)$: On input the security parameter λ, the key generation algorithm samples $k \leftarrow \mathsf{SKE.Gen}(1^\lambda)$, $H = H_\lambda$ from $\mathcal{H}$, and outputs (k, H).
- $\mathsf{Enc}(\mathsf{sk}, m)$: On input the secret key $\mathsf{sk} = (k, H)$ and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm samples $\mathsf{sk}_{\mathsf{1UTE}} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$, $r \xleftarrow{R} \{0,1\}^\lambda$, computes $|\mathsf{ct}_{\mathsf{1UTE}}\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}, m)$, $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(k, r)$ and outputs
$$|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{1UTE}}\rangle, \mathsf{ct}_{\mathsf{SKE}}, H(r) \oplus \mathsf{sk}_{\mathsf{1UTE}}).$$
- $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$: On input the secret key $\mathsf{sk} = (k, H)$ and a quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_1\rangle, \mathsf{ct}_2, \mathsf{ct}_3)$, the decryption algorithm outputs $\mathsf{1UTE.Dec}(H(\mathsf{SKE.Dec}(k, \mathsf{ct}_2)) \oplus \mathsf{ct}_3, |\mathsf{ct}_1\rangle)$.

Theorem 5.2 (Correctness)**.** *Suppose Π_{1UTE} and Π_{SKE} are correct. Then, Construction 5.1 is correct.*

Proof. Fix any $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}_\lambda$. Let $\mathsf{sk} = (k, H) \leftarrow \mathsf{Gen}(1^\lambda)$ and
$$|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{1UTE}}\rangle, \mathsf{ct}_{\mathsf{SKE}}, H(r) \oplus \mathsf{sk}_{\mathsf{1UTE}}) \leftarrow \mathsf{Enc}(\mathsf{sk}, m).$$
We consider the output of $\mathsf{Dec}(\mathsf{sk}, |\mathsf{ct}\rangle)$. By Π_{SKE} correctness, $r \leftarrow \mathsf{SKE.Dec}(k, \mathsf{ct}_{\mathsf{SKE}})$ with overwhelming probability. By Π_{1UTE} correctness, $m \leftarrow \mathsf{1UTE.Dec}(H(r) \oplus H(r) \oplus \mathsf{sk}_{\mathsf{1UTE}}, |\mathsf{ct}_{\mathsf{1UTE}}\rangle)$ with overwhelming probability, as desired. □

Theorem 5.3 (Everlasting Collusion-Resistant Security)**.** *Suppose Π_{1UTE} satisfies statistical indistinguishability security (Definition 2.5) and Π_{SKE} satisfies CPA security. Then, Construction 5.1 satisfies everlasting collusion-resistant security in the quantumly-accessible random oracle model.*

Proof. Suppose there exists a QPT first stage adversary $\mathcal{A}$ and an unbounded time second stage adversary $\mathcal{B}$ that has non-negligible advantage δ in the collusion-resistant security game and makes at most Q message queries. We define a sequence of hybrid experiments:

- Hyb_0: This is the collusion-resistant game with bit $b = 0$. In particular:
  - At the beginning of the game, the challenger samples $k \leftarrow \mathsf{SKE.Gen}(1^\lambda)$, $H = H_\lambda$ from $\mathcal{H}$, and sets $\mathsf{sk} = (k, H)$.
  - When adversary $\mathcal{A}$ makes its ith query $(m_0^{(i)}, m_1^{(i)})$, the challenger samples $\mathsf{sk}_{\mathsf{1UTE}}^{(i)} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$, $r_i \xleftarrow{R} \{0,1\}^\lambda$, sets $h_i = H(r_i) \oplus \mathsf{sk}_{\mathsf{1UTE}}^{(i)}$, and computes
  $$\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_0^{(i)}) \quad \text{and} \quad \mathsf{ct}_{\mathsf{SKE}}^{(i)} \leftarrow \mathsf{SKE.Enc}(k, r_i).$$
  The challenger gives the ciphertext $\left|\mathsf{ct}^{(i)}\right\rangle = \left(\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle, \mathsf{ct}_{\mathsf{SKE}}^{(i)}, h_i\right)$ to $\mathcal{A}$.

- Adversary $\mathcal{A}$ then outputs a classical string st, which is given to $\mathcal{B}$ along with sk from the challenger. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.

- Hyb_i: Same as Hyb_{i-1} except the challenger now computes $\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_1^{(i)})$.

Note that Hyb_Q is the collusion-resistant game with bit $b = 1$. We write $\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of Hyb_i with adversary $(\mathcal{A}, \mathcal{B})$. We now show that for each $i \in [Q]$, experiments $\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B})$ and $\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B})$ are indistinguishable.

Lemma 5.4. *Suppose Π_{1UTE} satisfies statistical indistinguishability security and Π_{SKE} satisfies one-way CPA security for QPT adversaries. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr[\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B}) = 1]\right| = \mathsf{negl}(\lambda).$$

Proof. We define the following intermediate hybrid experiments:

- Hyb'_{i-1}: Same as Hyb_{i-1} except the oracle given to $\mathcal{A}$ is replaced with $H_{r_i \to H'}$, which is H reprogrammed according to H' on an input r where H' is another independent uniformly random function. In particular, $H_{r_i \to H'}(x) = H(x)$ when $x \neq r_i$ and $H_{r_i \to H'}(x) = H'(x)$ when $x = r_i$.

- Hyb''_{i-1}: Same as Hyb'_{i-1} except for the following points:
  - The challenger samples $h_i \xleftarrow{R} \{0,1\}^\ell$.
  - The oracle given to $\mathcal{A}$ is replaced with H', where H' is a uniformly random function that is independent of H.
  - The oracle given to $\mathcal{B}$ is replaced with $H'_{r_i \to t}$, which is H' reprogrammed to $t = h_i \oplus \mathsf{sk}_{\mathsf{1UTE}}^{(i)}$ on input r_i.

- Hyb'''_{i-1}: Same as Hyb''_{i-1} except the challenger now computes $\left|\mathsf{ct}_{\mathsf{1UTE}}^{(i)}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}_{\mathsf{1UTE}}^{(i)}, m_1^{(i)})$.

Claim 5.5. Suppose Π_{SKE} satisfies one-way CPA security for QPT adversaries. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We will use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{B}_{\mathsf{SKE}}$ that breaks one-way CPA security.

First, consider an adversary $\hat{\mathcal{A}}$ which is given an oracle $\mathcal{O}$ that is either H or $H_{r_i \to H'}$ and an input (r_i, H) where $r_i \xleftarrow{R} \{0,1\}^\lambda$ and H represents the entire truth table of H. Adversary $\hat{\mathcal{A}}$ runs $\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B})$ except that it uses $\mathcal{O}$ to simulate $\mathcal{A}$'s random oracle queries and the truth table of H to simulate $\mathcal{B}$'s random oracle queries. We assume without loss of generality that $\mathcal{A}$ makes exactly $Q_{\mathcal{A}}$ queries. Thus, $\Pr\left[\mathsf{Hyb}_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] = \Pr\left[\hat{\mathcal{A}}^H(r_i, H) = 1\right]$ and $\Pr\left[\mathsf{Hyb}'_{i-1}(\mathcal{A}, \mathcal{B})\right] = \Pr\left[\hat{\mathcal{A}}^{H_{r_i \to H'}}(r_i, H) = 1\right]$. Setting $\mathcal{B}_{\mathsf{O2H}}$ to be an algorithm that measures a uniformly random query of $\hat{\mathcal{A}}$, by Lemma 2.1, we have

$$\delta' \leq \left|\Pr\left[\hat{\mathcal{A}}^H(r_i, H) = 1\right] - \Pr\left[\hat{\mathcal{A}}^{H_{r_i \to H'}}(r_i, H) = 1\right]\right| \leq 2Q_{\mathcal{A}} \cdot \sqrt{\Pr\left[\mathcal{B}_{\mathsf{O2H}}^{H_{r_i \to H'}}(r_i, H) = r_i\right]}.$$

Let $\hat{\mathcal{B}}_{\mathsf{O2H}}$ be the same as $\mathcal{B}_{\mathsf{O2H}}$ except it does not take the truth table of H as input and samples $h_i \xleftarrow{R} \{0,1\}^\ell$. Then,

$$\Pr\left[\mathcal{B}_{\mathsf{O2H}}^{H_{r_i \to H'}}(r_i, H) = r_i\right] = \Pr\left[\hat{\mathcal{B}}_{\mathsf{O2H}}^{H_{r_i \to H'}}(r_i) = r_i\right]$$

since the truth table for H is only needed to generate $h_i = H(r_i) \oplus \mathsf{sk}_{\mathsf{1UTE}}^{(i)}$ (it halts before $\mathcal{B}$ is run) and $H_{r_i \to H'}$ reveals nothing about $H(r_i)$. Furthermore, for any r_i, the function $H_{r_i \to H'}$ is random when H, H' are random, meaning

$$\Pr\left[\hat{\mathcal{B}}_{\mathsf{O2H}}^{H_{r_i \to H'}}(r_i) = r_i\right] = \Pr\left[\hat{\mathcal{B}}_{\mathsf{O2H}}^{H}(r_i) = r_i\right] \geq \left(\frac{\delta'}{2Q_{\mathcal{A}}}\right)^2.$$

Note that $\hat{\mathcal{B}}_{\mathsf{O2H}}$ ignores its input and simulates $\mathsf{Hyb}_{i-1}(\mathcal{A},\mathcal{B})$ by sampling $h_i \xleftarrow{R} \{0,1\}^\ell$, measuring a random query $j \xleftarrow{R} [Q_{\mathcal{A}}]$ of adversary $\mathcal{A}$, and outputting the measurement. With this in mind, we can construct $\mathcal{B}_{\mathsf{SKE}}$ as follows:

1. At the beginning of the game, $\mathcal{B}_{\mathsf{SKE}}$ is given $\mathsf{ct}^{(i)}_{\mathsf{SKE}}$ from its challenger, samples $h_i \xleftarrow{R} \{0,1\}^\ell$, and samples $i^* \xleftarrow{R} [Q_{\mathcal{A}}]$.

2. When adversary $\mathcal{A}$ makes the jth query $(m_0^{(j)}, m_1^{(j)})$, adversary $\mathcal{B}_{\mathsf{SKE}}$ does the following:

   - For each $j \in [Q]$, sample $\mathsf{sk}^{(j)}_{\mathsf{1UTE}} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$ and compute
   $$\left|\mathsf{ct}^{(j)}_{\mathsf{1UTE}}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}^{(j)}_{\mathsf{1UTE}}, m_0^{(j)}).$$
   - For $j \in [Q] \setminus \{i\}$, sample $r_j \xleftarrow{R} \{0,1\}^\lambda$, set $h_j = H(r_j) \oplus \mathsf{sk}^{(j)}_{\mathsf{1UTE}}$, and submit r_j to the CPA challenger to get $\mathsf{ct}^{(j)}_{\mathsf{SKE}}$.

   Adversary $\mathcal{B}_{\mathsf{SKE}}$ replies with $\left|\mathsf{ct}^{(j)}\right\rangle = \left(\left|\mathsf{ct}^{(j)}_{\mathsf{1UTE}}\right\rangle, \mathsf{ct}^{(j)}_{\mathsf{SKE}}, h_j\right)$.

3. For random oracle queries, $\mathcal{B}_{\mathsf{SKE}}$ simulates them efficiently (see Section 2.2) and measures query i^* that $\mathcal{A}$ makes. Adversary $\mathcal{B}_{\mathsf{SKE}}$ outputs the measurement outcome.

By definition $\mathcal{B}_{\mathsf{SKE}}$ outputs r_i (which we denote as the random message used to construct $\mathsf{ct}^{(i)}_{\mathsf{SKE}}$) with probability $\Pr\left[\hat{\mathcal{B}}^H_{\mathsf{O2H}}(r_i) = r_i\right] \geq \left(\frac{\delta'}{2Q_{\mathcal{A}}}\right)^2$, which is non-negligible. Thus, $\mathcal{B}_{\mathsf{SKE}}$ breaks one-way CPA security, a contradiction. □

Claim 5.6. For all $i \in [Q]$ and $\lambda \in \mathbb{N}$, $\Pr\left[\mathsf{Hyb}'_{i-1}(\mathcal{A},\mathcal{B}) = 1\right] = \Pr\left[\mathsf{Hyb}''_{i-1}(\mathcal{A},\mathcal{B}) = 1\right]$.

Proof. First, $h_i = H(r_i) \oplus \mathsf{sk}^{(i)}_{\mathsf{1UTE}}$ in Hyb'_{i-1} and h_i is uniform in Hyb''_{i-1}, so the marginal of h_i is identical in both distributions. Next, $H_{r_i \to H'}$ is independent of h_i in Hyb'_{i-1} and is a uniformly random function, which is also the case in Hyb''_{i-1} with H', so the hybrids have the same joint distribution for the first stage oracle and h_i. Moreover, in Hyb'_{i-1}, we have $H(r_i) = h_i \oplus \mathsf{sk}^{(i)}_{\mathsf{1UTE}}$ and in Hyb''_{i-1} we have $H'_{r_i \to t}(r_i) = t = h_i \oplus \mathsf{sk}^{(i)}_{\mathsf{1UTE}}$. Lastly, the first and second stage oracles only differ at the point r_i in both hybrids, meaning the joint distributions for both oracles and h_i are identical. Since the oracles and h_i are the only modified components, this completes the proof. □

Claim 5.7. Suppose Π_{1UTE} satisfies statistical indistinguishability security. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}''_{i-1}(\mathcal{A},\mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}'''_{i-1}(\mathcal{A},\mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A},\mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We will use $(\mathcal{A},\mathcal{B})$ to construct an adversary $(\mathcal{A}',\mathcal{B}')$ that breaks indistinguishability security as follows:

1. Adversary $\mathcal{A}'$ first samples $r_j \xleftarrow{R} \{0,1\}^\lambda, s_j \xleftarrow{R} \{0,1\}^\ell$ for each $j \in [Q]$ and sets $H'(r_j) = s_j$ for $j \neq i$. It also samples $\mathsf{sk} = k \leftarrow \mathsf{SKE.Gen}(1^\lambda)$ and for all $j \in [Q]$ samples $\mathsf{ct}^{(j)}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(k, r_j)$.

2. Adversary $\mathcal{A}'$ answers random oracle queries at random or according to the values already set for H' if they are queried.

3. When adversary $\mathcal{A}$ makes the jth query $(m_0^{(j)}, m_1^{(j)})$, adversary $\mathcal{A}'$ does the following:

   - For $j \neq i$, sample $\mathsf{sk}^{(j)}_{\mathsf{1UTE}} \leftarrow \mathsf{1UTE.Gen}(1^\lambda)$ and compute $\left|\mathsf{ct}^{(j)}_{\mathsf{1UTE}}\right\rangle \leftarrow \mathsf{1UTE.Enc}(\mathsf{sk}^{(j)}_{\mathsf{1UTE}}, m^{(j)}_{b_j})$, where $b_j = 0$ for $j > i$ and $b_j = 1$ for $j < i$. Set $h_j = s_j \oplus \mathsf{sk}^{(j)}_{\mathsf{1UTE}}$.
   - For $j = i$, sample $h_i \xleftarrow{R} \{0,1\}^\ell$ and send $m_0^{(i)}, m_1^{(i)}$ to the UTE challenger to get $\left|\mathsf{ct}^{(i)}_{\mathsf{1UTE}}\right\rangle$.

Adversary $\mathcal{A}'$ replies with $\left|\mathsf{ct}^{(j)}\right\rangle = \left(\left|\mathsf{ct}_{\mathsf{1UTE}}^{(j)}\right\rangle, \mathsf{ct}_{\mathsf{SKE}}^{(j)}, h_j\right)$.

4. When adversary $\mathcal{A}$ outputs st, adversary $\mathcal{A}'$ passes the state (st, h_i, H') to $\mathcal{B}'$, where H' is the current oracle information from $\mathcal{A}$'s queries and the values set for H'.

5. On input (st, h_i, H') and $\mathsf{sk}_{\mathsf{1UTE}}^{(i)}$, adversary $\mathcal{B}'$ sets $t = h_i \oplus \mathsf{sk}_{\mathsf{1UTE}}^{(i)}$ and sets $H'(r_i) = t$. It runs $\mathcal{B}$ on input $(\mathsf{st}, \mathsf{sk})$ and answers random oracle queries at random or according to the values already set for H' if they are queried. Adversary $\mathcal{B}'$ outputs whatever $\mathcal{B}$ outputs.

If the UTE challenger encrypts $m_0^{(i)}$, adversary $(\mathcal{A}', \mathcal{B}')$ perfectly simulates $\mathsf{Hyb}''_{i-1}(\mathcal{A}, \mathcal{B})$. If the UTE challenger encrypts $m_1^{(i)}$, adversary $(\mathcal{A}', \mathcal{B}')$ perfectly simulates $\mathsf{Hyb}'''_{i-1}(\mathcal{A}, \mathcal{B})$. Thus, $(\mathcal{A}', \mathcal{B}')$ has advantage δ' in the indistinguishability security game, a contradiction. □

Claim 5.8. Suppose Π_{SKE} satisfies one-way CPA security for QPT adversaries. Then, for all $i \in [Q]$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have

$$\left|\Pr\left[\mathsf{Hyb}'''_{i-1}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr[\mathsf{Hyb}_i(\mathcal{A}, \mathcal{B}) = 1]\right| = \mathsf{negl}(\lambda).$$

Proof. Analogous to the proofs of Claims 5.5 and 5.6. □

Combining Claims 5.5 to 5.8 proves the lemma by a hybrid argument.

□

The theorem follows by a hybrid argument via Lemma 5.4.

□

Remark 5.9 (Plain Model Insecurity). We note that for any concrete hash function H, if Π_{SKE} is FHE for quantum circuits (QFHE) with classical ciphertexts for classical messages, then even one-way non-everlasting UTE security is broken. Namely, QFHE can be used to get a classical ciphertext that contains the message given $|\mathsf{ct}\rangle$ by computing the homomorphic evaluation of Dec. Thus, the second-stage adversary can efficiently recover m given the SKE key k.

5.2 Impossibility of Weakly-Efficient Shadow Tomography

In this section, we show an efficient attack on any collusion-resistant UTE with everlasting security from weakly-efficient shadow tomography (WEST). In particular, this implies the impossibility of WEST for general mixed states in the QROM (since random oracles imply one-way functions and thus CPA-secure SKE).

Weakly-efficient shadow tomography. We first define the notion of weakly-efficient shadow tomography (WEST). This notion captures shadow tomography where the post-processing can run in unbounded time, but the initial stage must be efficient. In particular, this generalizes the notion of classical shadows [HKP20], which has been shown to compute useful information about unknown quantum states when the family of measurements are projections to pure states.

Definition 5.10 (Weakly-Efficient Shadow Tomography)**.** Let E denote a uniform quantum circuit family with classical binary output that takes as input $i \in [M]$ and an n-qubit quantum state ρ. Let a shadow tomography procedure be slightly redefined to be a pair of algorithms $(\mathcal{T}_1, \mathcal{T}_2)$ where $\mathcal{T}_1$ takes as input E and k copies of ρ, and outputs a classical string st such that $\Pr[|\mathcal{T}_2(i, \mathsf{st}) - \Pr[E(i, \rho) = 1]| \leq \varepsilon] \geq 1 - \delta$. We call said procedure *weakly-efficient* if the number of copies k and the runtime of $\mathcal{T}_1$ are both $\mathrm{poly}(n, \log M, 1/\varepsilon, \log 1/\delta)$. Note that $\mathcal{T}_2$ can run in *unbounded* time.

Theorem 5.11 (WEST Breaks t-Copy Everlasting Secure UTE)**.** *Let λ be a security parameter. Suppose there exists an algorithm $\mathcal{T} = (\mathcal{T}_1, \mathcal{T}_2)$ that satisfies Definition 5.10 for all mixed quantum states with number of copies $k = k(n, \log M, 1/\varepsilon, \log(1/\delta))$ for polynomial k. Then, there does not exist k-copy everlasting secure untelegraphable encryption that also satisfies correctness.*

Proof. Let $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with $\mathrm{poly}(\lambda)$ be a candidate correct UTE scheme with message space $\{0,1\}$ (without loss of generality) and decryption key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda\in\mathbb{N}}$. Let $n = n(\lambda)$ be the maximum size of a ciphertext, $M = |\mathcal{K}_\lambda|$, $\varepsilon < 1/2$ be a constant, and $\delta = 2^{-\lambda}$. We define the quantum circuit $E_\lambda(i,\rho)$ to output $\mathsf{Dec}(\mathsf{dk}, |\mathsf{ct}\rangle) \in \{0,1\}$ where the input is parsed as $i = \mathsf{dk}$ and $\rho = |\mathsf{ct}\rangle$. We now construct an adversary $(\mathcal{A},\mathcal{B})$ that breaks k-copy everlasting security of Π_{UTE}:

1. On input $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$, adversary $\mathcal{A}$ runs $\mathcal{T}_1(E_\lambda, \{|\mathsf{ct}_i\rangle\}_{i\in[k]})$ to get a classical state st, which $\mathcal{A}$ outputs as its classical state.

2. On input dk and st, adversary $\mathcal{B}$ outputs 1 if $\mathcal{T}_2(\mathsf{dk},\mathsf{st}) > 1/2$ and 0 otherwise.

By definition of UTE, $n = \mathrm{poly}(\lambda)$ and $\log M = \mathrm{poly}(\lambda)$. By the choice of δ, $\log(1/\delta) = \lambda$ and $1/\varepsilon = 3$. Thus, $\mathcal{T}_1$ runs in QPT and $(\mathcal{A},\mathcal{B})$ meets the efficiency condition. Since $\Pr[|\mathcal{T}_2(i,\mathsf{st}) - \Pr[E(i,\rho)=1]| \le \varepsilon] \ge 1-\delta$, by the setting of ε,δ, we have

$$\Pr[|\mathcal{T}_2(i,\mathsf{st}) - \Pr[E(i,\rho)=1]| < 1/2] \ge 1 - 2^{-\lambda}.$$

By correctness of UTE, we have $\Pr[E(i,\rho)=1] = \mathsf{negl}(\lambda)$ when $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$ are encryptions of 0 and $\Pr[E(i,\rho)=1] \ge 1-\mathsf{negl}(\lambda)$ when $\{|\mathsf{ct}_i\rangle\}_{i\in[k]}$ are encryptions of 1. This implies $\mathcal{T}_2(\mathsf{dk},\mathsf{st}) > 1/2$ for encryptions of 1 and $\mathcal{T}_2(\mathsf{dk},\mathsf{st}) < 1/2$ for encryptions of 0 with probability $1-2^{-\lambda}$. Thus, $(\mathcal{A},\mathcal{B})$ wins the k-copy security game with overwhelming probability. □

Corollary 5.12. *Weakly-efficient shadow tomography for all mixed quantum states does not exist in the quantum random oracle model.*

Proof. Follows by Theorems 5.2, 5.3 and 5.11 and the fact that collusion-resistant security implies t-copy security. To adapt the above attack to Construction 5.1 with only query access to the hash H, we can define the circuit E to run the decryption algorithm of Construction 5.1 modified to take the value $H(r)$ as input. The classical part of the ciphertext can be appended to st, which will allow $H(r)$ to be computed after the key is revealed. Thus, even if the WEST procedure requires the circuit E to be composed of atomic gates only (no hash gate), the attack still works. □

5.3 Impossibility of Public-Key Untelegraphable Encryption with Everlasting Security

In this section, we give evidence that a non-falsifiable assumption is necessary to construct everlasting secure UTE (and thus UE as well). We do so by showing that any black-box reduction that proves everlasting one-way security of a public-key UTE (UTPKE) scheme with respect to an assumption A can be used to construct an efficient adversary that breaks A, assuming one-way functions exist. This complements the observation that Construction 5.1 is broken for any concrete choice of hash function (see Remark 5.9) if the classical encryption scheme is quantum fully homomorphic.

Falsifiable cryptographic assumptions. We start by recalling the notion of a falsifiable cryptographic assumption. Intuitively, this notion refers to any cryptographic assumption which can be broken in a way that can efficiently checked, such as one-way functions. We give the formal definition from [GW11].

Definition 5.13 (Falsifiable Cryptographic Assumption [GW11])**.** A falsifiable cryptographic assumption consists of an interactive challenger $\mathcal{C}$ and a constant $c \in [0,1)$. On security parameter λ, the challenger $\mathcal{C}(1^\lambda)$ interacts with an algorithm $\mathcal{A}(1^\lambda)$ and may output a special symbol `win`. If this occurs, we say that $\mathcal{A}(1^\lambda)$ wins $\mathcal{C}(1^\lambda)$. The assumption associated with the tuple $(\mathcal{C},c)$ says that for all efficient (possibly non-uniform) algorithms $\mathcal{A}$, there exists a negligible function negl such that for all $\lambda\in\mathbb{N}$, $\Pr\big[\mathcal{A}(1^\lambda) \text{ wins } \mathcal{C}(1^\lambda)\big] \le c + \mathsf{negl}(\lambda)$, where the probability is over the random coins of $\mathcal{C}$ and $\mathcal{A}$. We say that a (possibly inefficient) algorithm $\mathcal{A}$ breaks such an assumption if its probability of winning exceeds that of the assumption. When considering quantum assumptions, we refer to the purified version of algorithm $\mathcal{A}$ as a unitary $\boldsymbol{U}_{\mathcal{A}}$.

We also define what it means for a black-box reduction to show everlasting security of a UTE scheme:

Definition 5.14 (Black-Box Reduction for Everlasting Secure UTPKE)**.** A black-box reduction showing everlasting security of a UTE scheme based on a falsifiable assumption $(\mathcal{C}, c)$ is an efficient oracle access algorithm $\mathcal{R}^{(\cdot)}$ such that, for every (possibly inefficient) adversary $(\mathcal{A}, \mathcal{B})$ that wins the everlasting UTPKE security game with non-negligible advantage, algorithm $\mathcal{R}^{\boldsymbol{U}_{\mathcal{A}}}$ breaks the assumption. We model oracle query to $\boldsymbol{U}_{\mathcal{A}}$ as follows: given an input state $|\psi\rangle = (\mathsf{pk}, |\mathsf{ct}\rangle)$, the response is the output register $\mathsf{R}_{\mathsf{out}}$ of $\boldsymbol{U}_{\mathcal{A}}|\psi\rangle$. Furthermore, we do not give oracle access to the inverse unitary $\boldsymbol{U}_{\mathcal{A}}^{\dagger}$. Note that since $\mathcal{B}$ runs in unbounded time in the everlasting security game, oracle access to $\mathcal{B}$ is not given.

We now state and prove the main theorem of this section:

Theorem 5.15 (Impossibility of Everlasting One-Way Secure UTPKE)**.** *Let $\lambda \in \mathbb{N}$ be a security parameter and $\delta \in (0,1)$ be a constant. Let $\Pi_{\mathsf{UTE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be any candidate UTPKE scheme that satisfies correctness. Suppose post-quantum CPA-secure encryption exists for size $2^{\lambda^{\delta}}$ adversaries such that the ciphertexts of different messages are statistically far apart. Then, for any falsifiable assumption $(\mathcal{C}, c)$, one of the following holds:*

- *The assumption $(\mathcal{C}, c)$ is false.*
- *There is no black-box reduction (Definition 5.14) showing everlasting one-way security of Π_{UTE} based on the assumption $(\mathcal{C}, c)$.*

Proof. Let $\lambda \in \mathbb{N}$ be the security parameter, and suppose the length of the keys output from $\mathsf{Gen}(1^{\lambda})$ is $|\mathsf{pk}| + |\mathsf{sk}| \leq \lambda^c$ for constant $c \in \mathbb{N}$. Let $n = n(\lambda) = \lambda^{(c+2)/\delta}$ be a CPA-secure encryption security parameter and let $\Pi_{\mathsf{SKE}} = \mathsf{SKE}.(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an SKE scheme that is CPA-secure for size $2^{n^{\delta}} = 2^{\lambda^{c+2}}$ adversaries and has ciphertexts that are statistically far apart. Now, consider an inefficient adversary $\mathcal{A}$ that on input $(1^{\kappa}, \mathsf{pk}, |\mathsf{ct}\rangle)$ does the following:

1. Measures the pk register (since it is supposed to be classical).
2. Rejection samples $(\mathsf{pk}', \mathsf{sk}') \leftarrow \mathsf{Gen}(1^{\kappa})$ until $\mathsf{pk}' = \mathsf{pk}$ and computes $m \leftarrow \mathsf{Dec}(\mathsf{sk}', |\mathsf{ct}\rangle)$.
3. Samples $\mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE}.\mathsf{Gen}(1^{n(\kappa)})$ and outputs $\mathsf{st} = \mathsf{SKE}.\mathsf{Enc}(\mathsf{sk}_{\mathsf{SKE}}, m)$.

Define the second stage adversary $\mathcal{B}$ to find m by brute-force, which is possible since ciphertexts for Π_{SKE} must be statistically far from each other. Clearly $(\mathcal{A}, \mathcal{B})$ breaks everlasting one-way security with non-negligible advantage.

We now show that for every polynomial-size distinguisher $\mathcal{D}(1^{\lambda})$, there exists a simulator $\mathcal{S}(1^{\lambda})$ and negligible function negl such that for all $\lambda \in \mathbb{N}$,

$$\Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{A}}}(1^{\lambda}) = 1\right] - \Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{S}}(1^{\lambda})}(1^{\lambda}) = 1\right] = \mathsf{negl}(\lambda).$$

We define an efficient non-uniform simulator $\mathcal{S}(1^{\lambda})$ for $\mathcal{A}$ that on input a query $(1^{\kappa}, \mathsf{pk}, |\mathsf{ct}\rangle)$ does the following:

1. If $\kappa \leq \kappa^*(\lambda) = \left\lfloor \log^{1/(c+1)} \lambda \right\rfloor$, the simulator measures the pk register, uses its advice to get a valid key sk for pk, and runs the steps of $\mathcal{A}$ from that point. The advice consists of a table T_{λ} that maps public keys $\mathsf{pk} \in \{0,1\}^{\kappa^c}$ to a corresponding secret key sk for all $\kappa \leq \kappa^*(\lambda)$.
2. If $\kappa > \kappa^*(\lambda)$, the simulator measures the pk register, samples $\mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE}.\mathsf{Gen}(1^n)$ and outputs $\mathsf{st} = \mathsf{SKE}.\mathsf{Enc}(\mathsf{sk}_{\mathsf{SKE}}, 0)$.

Note that T_{λ} has size at most $\kappa^*(\lambda)2^{\kappa^*(\lambda)} = \mathrm{poly}(\lambda)$ by the bound on the key size, so $\mathcal{S}$ is efficient and has polynomial-sized advice. Moreover, by an averaging argument, there exists a choice of T_{λ} such that for the adversary $\mathcal{A}(T_{\lambda})$ that answers queries where $\kappa \leq \kappa^*(\lambda)$ using the table, we have

$$\Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{A}(T_{\lambda})}}(1^{\lambda}) = 1\right] \geq \Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{A}}}(1^{\lambda}) = 1\right],$$

so we pick this table as advice. We now appeal to CPA-security of Π_{SKE} to show

$$\Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{A}(T_{\lambda})}}(1^{\lambda}) = 1\right] - \Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{S}}(1^{\lambda})}(1^{\lambda}) = 1\right] = \mathsf{negl}(\lambda).$$

Suppose $\mathcal{D}(1^\lambda)$ makes at most $q(\lambda)$ queries. For $i \in [q(\lambda)]$, define $\mathcal{O}_i$ to be the oracle that answers the first i queries with $\mathcal{A}(T_\lambda)$ and the last $q(\lambda) - i$ queries with $\mathcal{S}(\lambda)$. Then, for $i \in [q(\lambda)]$,

$$\left|\Pr\left[\mathcal{D}^{\mathcal{O}_{i-1}}(1^\lambda) = 1\right] - \Pr\left[\mathcal{D}^{\mathcal{O}_i}(1^\lambda) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Suppose $\mathcal{D}$ can distinguish $\mathcal{O}_{i-1}$ and $\mathcal{O}_i$. We construct a non-uniform adversary $\mathcal{B}$ that breaks CPA-security as follows:

1. We first define the advice distribution. For a security parameter $\lambda \in \mathbb{N}$, adversary $\mathcal{B}$ is given T_λ along with a transcript of $\mathcal{D}^{\mathcal{A}(T_\lambda)}(1^\lambda; r)$ for the first i queries and $i-1$ responses along with the randomness string r and the message m that $\mathcal{A}(T_\lambda)$ would derive (see step 2 of $\mathcal{A}$) in the ith query.
2. Adversary $\mathcal{B}$ starts running $\mathcal{D}(1^\lambda; r)$ and answers the first $i-1$ queries as in the transcript.
3. If the ith query satisfies $\kappa \leq \kappa^*(\lambda)$, respond with T_λ. Otherwise, submit the message pair $(0, m)$ to the CPA challenger for security parameter $1^{n(\kappa)}$ to get back st and respond with it.
4. For the remaining queries, answer how $\mathcal{S}(1^\lambda)$ would. Output whatever $\mathcal{D}$ outputs.

Adversary $\mathcal{B}$ is efficient so long as $\mathcal{D}$ is. If st is an encryption of m, adversary $\mathcal{B}$ simulates $\mathcal{O}_i$ and if st is an encryption of 0, adversary $\mathcal{B}$ simulates $\mathcal{O}_{i-1}$. By assumption on the security of Π_{SKE}, for $\kappa > \kappa^*(\lambda)$, no adversaries of size

$$s(\lambda) = 2^{n(\kappa)^\delta} > 2^{n(\kappa^*(\lambda))^\delta} = 2^{\log^{c'} \lambda},$$

can break CPA security with non-negligible probability, where $c' = (c+2)/(c+1) > 1$. Since $s(\lambda)$ is super polynomial in λ, adversary $\mathcal{B}$ contradicts CPA security of Π_{SKE}. Thus,

$$\Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{A}(T_\lambda)}}(1^\lambda) = 1\right] - \Pr\left[\mathcal{D}^{\boldsymbol{U}_{\mathcal{S}(1^\lambda)}}(1^\lambda) = 1\right] = \mathsf{negl}(\lambda)$$

follows by a hyrbid argument. To conclude the proof, consider a candidate reduction $\mathcal{R}$. If we have

$$\Pr\left[\mathcal{R}^{\boldsymbol{U}_{\mathcal{A}}}(1^\lambda) \text{ wins } \mathcal{C}(1^\lambda)\right] > c + \varepsilon(\lambda)$$

for non-negligible ε, then it must also be the case that

$$\Pr\left[\mathcal{R}^{\boldsymbol{U}_{\mathcal{S}(1^\lambda)}}(1^\lambda) \text{ wins } \mathcal{C}(1^\lambda)\right] > c + \varepsilon(\lambda) - \mathsf{negl}(\lambda),$$

which implies $(\mathcal{C}, c)$ is false. □

Corollary 5.16 (Impossibility of Everlasting One-Way Secure UTPKE from OWFs)**.** *Let $\lambda \in \mathbb{N}$ be a security parameter, $\delta \in (0,1)$ be a constant, and suppose post-quantum one-way functions exist for size 2^{λ^δ} adversaries. Then, any candidate UTPKE cannot have a black-box proof of everlasting one-way security based on a falsifiable assumption $(\mathcal{C}, c)$, unless the assumption is false.*

Proof. Follows by Theorem 5.15 and the fact that CPA-secure encryption with statistically far ciphertexts for different messages can be constructed from one-way functions. □

6 Secret Sharing Resilient to Joint and Unbounded Classical Leakage

In this section, we define and construct unbounded joint-leakage-resilient secret sharing (UJLRSS) for all policies from collusion-resistant UTE and classical secret sharing for all policies. Our definition extends the notion of unbounded leakage-resilient secret sharing defined in [ÇGLR24] to the setting of joint leakage.

6.1 Secret Sharing Definitions

In this section, we define UJLRSS and compare our definition to the recently introduced definitions of ULRSS in [ÇGLR24] and USS in [AGLL24]. Intuitively, the security notion we consider involves groups of non-qualifying parties as first stage adversaries, where each group classically leaks their shares, except for one group which gets its honest shares. Given the honest set of shares and all the classical leakage, it should be hard to distinguish any two secrets from each other. In addition, we recall the definition of classical secret sharing for policies. We give the formal definitions below, starting with the definition of an access policy. Much of the prose for this is taken from [JKK+17].

Definition 6.1 (Access Policy)**.** An access policy P on parties $[n]$ is a monotone set of subsets of $[n]$. In particular, $P \subseteq 2^{[n]}$ and for all $S \in P$ and $S \subseteq S'$ it holds that $S' \in P$.

In this work, we will be treating policies as functions $P : 2^{[n]} \to \{0,1\}$ that output 1 on input X if $X \in P$. Furthermore, we will assume P can be described by a *monotone Boolean circuit*. These are directed acyclic graphs in which leaves are labeled by input variables and every internal node is labeled by an AND or OR operation. We assume the gates in the circuit have fan-in k_{in} and fan-out at most k_{out}, so for a given gate g we represent the number of output wires as $k_{\mathsf{out}}(g)$. The computation is done in the natural way from the leaves to the root which corresponds to the output of the computation.

Classical secret sharing. We now recall the notion of classical secret sharing for access policies. This notion can be constructed for general policies from any classical CPA-secure encryption scheme, which itself can be constructed from one-way functions [VNS+03].

Definition 6.2 (Secret Sharing)**.** Let λ be a security parameter and n be the number of parties. An secret sharing scheme for access policy $P : 2^{[n]} \to \{0,1\}$ with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of PPT algorithms $\Pi_{\mathsf{SS}} = (\mathsf{Share}, \mathsf{Reconst})$ with the following syntax:

- $\mathsf{Share}(1^\lambda, 1^n, m) \to (s_1, \ldots, s_n)$: On input the security parameter λ, the number of parties n, and a message $m \in \mathcal{M}_\lambda$, the sharing algorithm outputs a tuple of shares $(s_1, \ldots, s_n)$ for each party.
- $\mathsf{Reconst}(X, s_X) \to m$: On input the shares of a subset of parties $(X \subseteq [n], s_X = \{s_i\}_{i\in X})$, the reconstruction algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{SS} satisfy the following properties:

- **Correctness:** For all $\lambda, n \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and X such that $P(X) = 1$, we have

$$\Pr\left[m \leftarrow \mathsf{Reconst}(X, s_X) : (s_1, \ldots, s_n) \leftarrow \mathsf{Share}(1^\lambda, 1^n, m)\right] \geq 1 - \mathsf{negl}(\lambda).$$

- **Security:** For a security parameter λ, number of parties n, a bit $b \in \{0,1\}$, and an adversary $\mathcal{A}$, we define the security game as follows:

  1. At the beginning of the game, adversary $\mathcal{A}$ outputs messages $m_0, m_1 \in \mathcal{M}_\lambda$, along with a set X such that $P(X) = 0$.
  2. The challenger samples $(s_1, \ldots, s_n) \leftarrow \mathsf{Share}(1^\lambda, 1^n, m_b)$, and gives s_X to $\mathcal{A}$. Adversary $\mathcal{A}$ outputs a bit b', which is the output of the experiment.

  We say Π_{SS} is secure if for all PPT adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

$$|\Pr[b' = 1 | b = 0] - \Pr[b' = 1 | b = 1]| = \mathsf{negl}(\lambda)$$

  in the above security game.

We now define UJLRSS for access policies.

Definition 6.3 (Unbounded Joint-Leakage-Resilient Secret Sharing)**.** Let λ be a security parameter and n be the number of parties. An unbounded joint-leakage-resilient secret sharing scheme for access policy $P : 2^{[n]} \to \{0,1\}$ with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{LRSS}} = (\mathsf{Share}, \mathsf{Reconst})$ with the following syntax:

- $\mathsf{Share}(1^\lambda, 1^n, m) \to (|s_1\rangle, \ldots, |s_n\rangle)$: On input the security parameter λ, the number of parties n, and a message $m \in \mathcal{M}_\lambda$, the sharing algorithm outputs a tuple of quantum shares $(|s_1\rangle, \ldots, |s_n\rangle)$ for each party.
- $\mathsf{Reconst}(X, |s_X\rangle) \to m$: On input the shares of a subset of parties $(X \subseteq [n], |s_X\rangle = \{|s_i\rangle\}_{i\in X})$, the reconstruction algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{LRSS} satisfy the following properties:

- **Correctness:** For all $\lambda, n \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and X such that $P(X) = 1$, we have

$$\Pr\left[m \leftarrow \mathsf{Reconst}(X, |s_X\rangle) : (|s_1\rangle, \ldots, |s_n\rangle) \leftarrow \mathsf{Share}(1^\lambda, 1^n, m)\right] \geq 1 - \mathsf{negl}(\lambda).$$

- **UJLR Security:** For a security parameter λ, number of parties n, a bit $b \in \{0,1\}$, and a two-stage adversary $(\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{B})$, we define the secret sharing security game with joint and unbounded classical leakage as follows:

  1. At the beginning of the game, $\mathcal{A}_0$ outputs messages $m_0, m_1 \in \mathcal{M}_\lambda$, along with a partition of $[n]$ denoted as $(V^*, V_1, \ldots, V_\ell)$ for $\ell \leq n-1$, where $P(V_i) = P(V^*) = 0$ for $i \in [\ell]$.
  2. The challenger samples $(|s_1\rangle, \ldots, |s_n\rangle) \leftarrow \mathsf{Share}(1^\lambda, 1^n, m_b)$, and gives $|s_{V_i}\rangle$ to $\mathcal{A}_i$ for $i \in [\ell]$.
  3. For $i \in [\ell]$, adversary $\mathcal{A}_i$ outputs a classical state st_i, which is given to $\mathcal{B}$. Additionally, the challenger gives $|s_{V^*}\rangle$ to $\mathcal{B}$.
  4. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.

  We say Π_{LRSS} satisfies UJLR security if for all QPT adversaries $(\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{B})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

$$|\Pr[b_{\mathcal{B}} = 1|b = 0] - \Pr[b_{\mathcal{B}} = 1|b = 1]| = \mathsf{negl}(\lambda)$$

  in the above security game.

Comparison with [ÇGLR24]. The definition of unbounded leakage-resilient secret sharing (ULRSS) in [ÇGLR24] has the same adversary structure as Definition 6.3, except each first-stage adversary gets a single share and tries to classically leak it. Our notion of UJLRSS allows any set of unqualified shares to be an input to the leakage function. While ULRSS does guarantee individual shares cannot be classically leaked, a first-stage adversary can easily break the construction of [ÇGLR24] if it is given two shares.

Comparison with [AGLL24]. The definition of USS in [AGLL24] has the same adversary structure as Definition 6.3, except each first-stage adversary gets a single share and tries to clone it, and the second stage adversaries do not get honest shares unless the first-stage adversary sends them. Collusion in the USS definition is captured via an entanglement graph, where an edge corresponds to two first stage adversaries sharing entanglement. The USS security notions are then compared by the number of connected components in the entanglement graph, with 1 component being the strongest. However, this is counter-intuitive for defining collusion, since if all parties collude there should never be security.

Furthermore, the USS definition does not guarantee that individual shares are unclonable. For instance, consider a 2-out-of-2 USS scheme where one share is a UE ciphertext encrypting the secret and the other is the (classical) UE secret key. The classical share can easily be cloned or classically leaked for a second stage adversary with the honest ciphertext share, which is also quite counter-intuitive.

However, some of this seems inherent, since having enough groups of parties colluding to cover the set $[n]$ would allow a cloning adversary to possibly output a different set of honest shares to each second-stage adversary, which would immediately break security. Thus, unbounded classical leakage-resilience yields both a more natural and satisfiable definition for the task of secret sharing.

6.2 Constructing UJLRSS for General Policies

In this section, we describe our construction of UJLRSS for all policies from collusion-resistant UTE and classical secret sharing for policies. The construction follows the structure of Yao's scheme (see [VNS+03, JKK+17]).

Construction 6.4 (UJLRSS for Policies)**.** Let λ be a security parameter, n be the number of parties, $P : 2^{[n]} \to \{0,1\}$ be a policy with corresponding circuit $\hat{P}$ of depth d and size t, and $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ be a message space. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{SS}} = (\mathsf{SS.Share}, \mathsf{SS.Reconst})$ be a classical secret sharing scheme for P with message space $\mathcal{M}$ and share space $\mathcal{S} = \{\mathcal{S}_\lambda\}_{\lambda\in\mathbb{N}}$.
- Let $\Pi_{\mathsf{UTE}} = (\mathsf{UTE.Gen}, \mathsf{UTE.Enc}, \mathsf{UTE.Dec})$ be a collusion-resistant UTE scheme with message space $\mathcal{S} \cup \{0,1\}^{\ell(\lambda)}$ and key space $\{0,1\}^\ell$.

We construct our unbounded joint-leakage-resilient secret sharing scheme $\Pi_{\mathsf{LRSS}} = (\mathsf{Share}, \mathsf{Reconst})$ as follows:

- $\mathsf{Share}(1^\lambda, 1^n, m)$: On input the security parameter λ, the number of parties n, and a message $m \in \mathcal{M}_\lambda$, the sharing algorithm does the following:
  1. Sample $(s'_1, \ldots, s'_n) \leftarrow \mathsf{SS.Share}(1^\lambda, 1^n, m)$.
  2. For the root (output) gate $g^* \in \hat{P}$, sample $(\mathsf{ek}_{g^*}, \mathsf{dk}_{g^*}) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$ and for $i \in [n]$ compute $\left|\mathsf{ct}_{i,1}^{(g^*)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_{g^*}, s'_i)$.
  3. For each gate g in the policy circuit $\hat{P}$ (starting from the root), let $w'_1, \ldots, w'_{k_{\mathsf{out}}(g)}$ represent the output wires with values $\mathsf{val}(w'_1), \ldots, \mathsf{val}(w'_{k_{\mathsf{out}}})$ (unless $g = g^*$) and $w_1, \ldots, w_{k_{\mathsf{in}}}$ represent the input wires. If $g \neq g^*$, sample $(\mathsf{ek}_g, \mathsf{dk}_g) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$ and do the following:
     - If g is an AND gate, sample the values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ uniformly from $\{0,1\}^\ell$ such that $\mathsf{val}(w_1) \oplus \cdots \oplus \mathsf{val}(w_{k_{\mathsf{in}}}) = \mathsf{dk}_g$.
     - If g is an OR gate, set the values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ all to dk_g.

     For $g \neq g^*$, $i \in [n]$, and $j \in [k_{\mathsf{out}}(g)]$, compute $\left|\mathsf{ct}_{i,j}^{(g)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g, \mathsf{val}(w'_j))$.
  4. Let W_i be the set of input wires associated with the ith input variable x_i. Then, for each $i \in [n]$ output the share
$$|s_i\rangle = \left(\{w, \mathsf{val}(w)\}_{w\in W_i}, \left\{g, \left\{\left|\mathsf{ct}_{i,j}^{(g)}\right\rangle\right\}_{j\in[k_{\mathsf{out}}(g)]}\right\}_{g\in\hat{P}}\right). \tag{4}$$
- $\mathsf{Reconst}(X, |s_X\rangle)$: On input the shares of a subset of parties $(X \subseteq [n], |s_X\rangle = \{|s_i\rangle\}_{i\in X})$, the reconstruction algorithm does the following:
  1. If $P(X) = 0$, output $\perp$. Otherwise, fix some $i^* \in X$ and for $i \in X$ parse $|s_i\rangle$ as in Equation (4). For $i \in X$ and $w \in W_i$ add $(w, \mathsf{val}(w))$ to a table T mapping wires to values.
  2. For $i = 1$ to $d - 1$, do the following:
     (a) For each gate g of depth i that outputs 1 on X (denoted as $\hat{P}_g(X) = 1$ going forward), compute dk_g using T and compute $\mathsf{val}(w'_j) \leftarrow \mathsf{UTE.Dec}\left(\mathsf{dk}_g, \left|\mathsf{ct}_{i^*,j}^{(g)}\right\rangle\right)$ for $j \in [k_{\mathsf{out}}]$.
     (b) For $j \in [k_{\mathsf{out}}]$, add $(w'_j, \mathsf{val}(w'_j))$ to T.
  3. For the output gate g^*, use T to compute dk_{g^*} and compute $s'_i \leftarrow \mathsf{UTE.Dec}\left(\mathsf{dk}_{g^*}, \left|\mathsf{ct}_{i,1}^{(g^*)}\right\rangle\right)$ for $i \in X$. Output $\mathsf{SS.Reconst}(X, s'_X)$.

Theorem 6.5 (Correctness)**.** *Suppose Π_{UTE} and Π_{SS} are correct. Then, Construction 6.4 is correct.*

Proof. Take any $\lambda, n \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and any set $X \subseteq [n]$ such that $P(X) = 1$. Let $(|s_1\rangle, \ldots, |s_n\rangle) \leftarrow \mathsf{Share}(1^\lambda, 1^n, m)$ and consider the output of $\mathsf{Reconst}(X, |s_X\rangle)$. Since $P(X) = P_{g^*}(X) = 1$, Reconst does not output $\perp$, and furthermore there exists a path of gates that output 1 from the leaves of $\hat{P}$ to the root (output) gate g^*. Thus, for all such gates g (of increasing depth) and a fixed $i^* \in X$, we have

$$\mathsf{val}(w'_j) \leftarrow \mathsf{UTE.Dec}\left(\mathsf{dk}_g, \left|\mathsf{ct}^{(g)}_{i^*,j}\right\rangle\right) \text{ for } j \in [k_{\mathsf{out}}(g)]$$

with overwhelming probability by correctness of Π_{UTE}. We also have $s'_i \leftarrow \mathsf{UTE.Dec}\left(\mathsf{dk}_{g^*}, \left|\mathsf{ct}^{(g^*)}_{i,1}\right\rangle\right)$ for $i \in X$ again by Π_{UTE} correctness, where s'_i are honest shares generated by SS.Share by definition of Share. Therefore, by Π_{SS} correctness, we have $m \leftarrow \mathsf{SS.Reconst}(X, s'_X)$ with overwhelming probability, since Π_{SS} is a secret sharing scheme for P. The theorem follows by a union bound. □

Theorem 6.6 (UJLR Security)**.** *Suppose Π_{SS} is secure for QPT adversaries and Π_{UTE} satisfies collusion-resistant security (Definition 2.6) with second stage encryption queries (Remark 3.23). Then, Construction 6.4 satisfies UJLR security.*

Proof. Let $n = n(\lambda), d = d(\lambda), t = (\lambda)$ be arbitrary polynomials. Suppose there exists a QPT adversary $(\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_n), \mathcal{B})$ that has non-negligible advantage δ in the UJLR security game (Definition 6.3). For a set $X \subseteq [n]$, such that $P(X) = 0$, let L_X be the list of gates g ordered by increasing depth such the $\hat{P}_g(X) = 0$. Let $L_X[j]$ denote the jth gate in L_X. We now define a sequence of hybrid experiments using indices $i \in [\ell]$ and $j \in [t]$:

- $\mathsf{Hyb}^{(b)}_{1,0}$: This is the UJLR security game with challenge bit $b \in \{0,1\}$. In particular:
  - At the beginning of the game, $\mathcal{A}_0$ outputs messages $m_0, m_1 \in \mathcal{M}_\lambda$, along with a partition $(V^*, V_1, \ldots, V_\ell)$ of $[n]$ for $\ell \leq n-1$, where $P(V_i) = P(V^*) = 0$ for $i \in [\ell]$.
  - The challenger samples $(s'_1, \ldots, s'_n) \leftarrow \mathsf{SS.Share}(1^\lambda, 1^n, m_b)$, $(\mathsf{ek}_{g^*}, \mathsf{dk}_{g^*}) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$, and for $i \in [n]$ computes $\left|\mathsf{ct}^{(g^*)}_{i,1}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_{g^*}, s'_i)$. For each $g \in \hat{P}$ (from the root), the challenger does the following:
    - If $g \neq g^*$, sample $(\mathsf{ek}_g, \mathsf{dk}_g) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$.
    - If g is an AND gate, sample the values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ uniformly from $\{0,1\}^\ell$ such that $\mathsf{val}(w_1) \oplus \cdots \oplus \mathsf{val}(w_{k_{\mathsf{in}}}) = \mathsf{dk}_g$.
    - If g is an OR gate, set the values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ all to dk_g.
    - If $g \neq g^*$, for $i \in [n]$, and $j \in [k_{\mathsf{out}}(g)]$ compute $\left|\mathsf{ct}^{(g)}_{i,j}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g, \mathsf{val}(w'_j))$.

    For each $i \in [n]$, the challenger sets

$$|s_i\rangle = \left(\{w, \mathsf{val}(w)\}_{w \in W_i}, \left\{g, \left\{\left|\mathsf{ct}^{(g)}_{i,j}\right\rangle\right\}_{j \in [k_{\mathsf{out}}(g)]}\right\}_{g \in \hat{P}}\right).$$

    The challenger gives $|s_{V_i}\rangle$ to $\mathcal{A}_i$ for $i \in [\ell]$.
  - For $i \in [\ell]$, adversary $\mathcal{A}_i$ outputs a classical state st_i, which is given to $\mathcal{B}$. Additionally, the challenger gives $|s_{V^*}\rangle$ to $\mathcal{B}$. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.
- $\mathsf{Hyb}^{(b)}_{i,j}$: Same as $\mathsf{Hyb}^{(b)}_{i,j-1}$ except for gate $g = L_X[j]$, the challenger samples

$$\left|\mathsf{ct}^{(g)}_{i',j'}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g, 0)$$

  for $i' \in V_i$ and $j' \in [k_{\mathsf{out}}(g)]$. Here we overload 0 to mean the all 0s string or its equivalent in $\mathcal{S}$.

Note that $\mathsf{Hyb}^{(b)}_{i,t}$ is the same as $\mathsf{Hyb}^{(b)}_{i+1,0}$ for $i \in [\ell - 1]$. We write $\mathsf{Hyb}^{(b)}_{i,j}(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of $\mathsf{Hyb}^{(b)}_{i,j}$ with adversary $(\mathcal{A}, \mathcal{B})$. We show that $\mathsf{Hyb}^{(b)}_{i,j-1}(\mathcal{A}, \mathcal{B})$ and $\mathsf{Hyb}^{(b)}_{i,j}(\mathcal{A}, \mathcal{B})$ are indistinguishable for any $i \in [\ell]$ and $j \in [t]$, and then show $\mathsf{Hyb}^{(0)}_{\ell,t}(\mathcal{A}, \mathcal{B})$ and $\mathsf{Hyb}^{(1)}_{\ell,t}(\mathcal{A}, \mathcal{B})$ are indistinguishable.

Lemma 6.7. *Suppose Π_{UTE} satisfies collusion-resistant security with second stage encryption queries. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}^{(b)}_{i,j-1}(\mathcal{A},\mathcal{B})=1\right]-\Pr\left[\mathsf{Hyb}^{(b)}_{i,j}(\mathcal{A},\mathcal{B})=1\right]\right|=\mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A},\mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We use $(\mathcal{A},\mathcal{B})$ to construct an adversary $(\mathcal{A}',\mathcal{B}')$ that breaks collusion-resistant security as follows:

1. Adversary $\mathcal{A}'$ starts by running $\mathcal{A}_0$ to get $m_0,m_1 \in \mathcal{M}_\lambda$ and $(V^*,V_1,\ldots,V_\ell)$. Adversary $\mathcal{A}'$ samples $(s'_1,\ldots,s'_n) \leftarrow \mathsf{SS.Share}(1^\lambda,1^n,m_b)$.
2. For each $g \in \hat{P}$ (from the root), adversary $\mathcal{A}'$ does the following:
   (a) If $g \neq L_{V_i}[j]$, sample $(\mathsf{ek}_g,\mathsf{dk}_g) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$.
   (b) If $g \neq L_{V_i}[j]$ is an AND gate, sample values $\mathsf{val}(w_1),\ldots,\mathsf{val}(w_{k_{\mathsf{in}}})$ uniformly from $\{0,1\}^\ell$ such that $\mathsf{val}(w_1)\oplus\cdots\oplus\mathsf{val}(w_{k_{\mathsf{in}}})=\mathsf{dk}_g$. If $g \neq L_{V_i}[j]$ is an OR gate, set all values $\mathsf{val}(w_1),\ldots,\mathsf{val}(w_{k_{\mathsf{in}}})$ to dk_g. If $g = L_{V_i}[j]$ is an AND gate, sample $\mathsf{val}(w) \xleftarrow{R} \{0,1\}^\ell$ for all input wires such that $\hat{P}_w(V_i)=1$ (the wire is 1 on input V_i).
   (c) If $g = L_{V_i}[j']$ such that $j'<j$, compute $\left|\mathsf{ct}^{(g)}_{i',j''}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g,0)$ for $j'' \in [k_{\mathsf{out}}(g)]$ and $i' \in V_i$.
   (d) If $g = L_{V_i}[j]$, submit $|V_i|$ copies of the messages $\{0,\mathsf{val}(w'_{j'})\}_{j'\in[k_{\mathsf{out}}(g)]}$ to the UTE challenger to get $\left|\mathsf{ct}^{(g)}_{i',j'}\right\rangle$ for $i' \in V_i$ and $j' \in [k_{\mathsf{out}}(g)]$.
   (e) If $g = L_{V_i}[j']$ such that $j'>j$, compute $\left|\mathsf{ct}^{(g)}_{i',j''}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g,\mathsf{val}(w'_{j''}))$ for $j'' \in [k_{\mathsf{out}}(g)]$ and $i' \in V_i$.
3. For $i' \in V_i$, adversary $\mathcal{A}'$ constructs $|s_{i'}\rangle$ as in Equation (4) and gives $|s_{V_i}\rangle$ to $\mathcal{A}_i$ to get st_i. Adversary $\mathcal{A}'$ outputs $\mathsf{st}=(\mathsf{st}_i,\{\mathsf{ek}_g,\mathsf{dk}_g\}_{g\neq L_{V_i}[j]},s'_{[n]},T)$, where T is the current wire to value mapping.
4. On input $\mathsf{dk}_{\hat{g}}$ for $\hat{g}=L_{V_i}[j]$ and the state $\mathsf{st}=(\mathsf{st}_i,\{\mathsf{ek}_g,\mathsf{dk}_g\}_{g\neq\hat{g}},s'_{[n]},T)$, adversary $\mathcal{B}'$ first samples the missing values for the input wires of $\hat{g}$.
5. For index gate pairs (i'',g) such that $g \in L_{V_{i'}}$ for some $i' \in [i-1]$ and additionally $i'' \in V_{i'}$, adversary $\mathcal{B}'$ computes $\left|\mathsf{ct}^{(g)}_{i'',j'}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g,0)$ for $j' \in [k_{\mathsf{out}}(g)]$. Otherwise, if $g \neq g^*$, adversary $\mathcal{B}'$ computes $\left|\mathsf{ct}^{(g)}_{i'',j}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g,\mathsf{val}(w'_{j'}))$ for $j' \in [k_{\mathsf{out}}(g)]$ and if $g=g^*$, it computes $\left|\mathsf{ct}^{(g^*)}_{i'',1}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g,s'_{i''})$. For generating ciphertexts without $\mathsf{ek}_{\hat{g}}$, adversary $\mathcal{B}'$ makes an encryption query to its challenger.
6. For $i' \in [n]\setminus V_i$, adversary $\mathcal{B}'$ constructs $|s_{i'}\rangle$ as in Equation (4) and gives $\left|s_{V_{i''}}\right\rangle$ to $\mathcal{A}_{i''}$ to get $\mathsf{st}_{i''}$ for $i'' \neq i$. It gives $(\mathsf{st}_1,\ldots,\mathsf{st}_\ell,|s_{V^*}\rangle)$ to $\mathcal{B}$ and outputs whatever $\mathcal{B}$ outputs.

Clearly $(\mathcal{A}',\mathcal{B}')$ is QPT if $(\mathcal{A},\mathcal{B})$ is. If the UTE challenger encrypts honest values, $(\mathcal{A}',\mathcal{B}')$ simulates $\mathsf{Hyb}^{(b)}_{i,j-1}(\mathcal{A},\mathcal{B})$ perfectly. If the UTE challenger encrypts 0s, $(\mathcal{A}',\mathcal{B}')$ simulates $\mathsf{Hyb}^{(b)}_{i,j}(\mathcal{A},\mathcal{B})$ perfectly. Thus, $(\mathcal{A}',\mathcal{B}')$ has advantage δ' in the collusion-resistant security game, a contradiction. □

Lemma 6.8. *Suppose Π_{SS} is secure for QPT adversaries. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}^{(0)}_{\ell,t}(\mathcal{A},\mathcal{B})=1\right]-\Pr\left[\mathsf{Hyb}^{(1)}_{\ell,t}(\mathcal{A},\mathcal{B})=1\right]\right|=\mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A},\mathcal{B})$ distinguish the experiments with non-negligible probability δ'. We use $(\mathcal{A},\mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks security of Π_{SS}:

1. Adversary $\mathcal{A}'$ starts by running $\mathcal{A}_0$ to get $m_0,m_1 \in \mathcal{M}_\lambda$ and $(V^*,V_1,\ldots,V_\ell)$. Adversary $\mathcal{A}'$ sends (m_0,m_1,V^*) to its challenger and gets back the shares s'_{V^*}.

2. Adversary $\mathcal{A}'$ samples $(\mathsf{ek}_{g^*}, \mathsf{dk}_{g^*}) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$, computes

$$\left|\mathsf{ct}_{i,1}^{(g^*)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_{g^*}, s_i') \text{ for } i \in V^*,$$

and computes

$$\left|\mathsf{ct}_{i,1}^{(g^*)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_{g^*}, 0) \text{ for } i \in [n] \setminus V^*.$$

3. For each $g \in \hat{P}$ (from the root), adversary $\mathcal{A}'$ does the following:
   (a) If $g \neq g^*$, sample $(\mathsf{ek}_g, \mathsf{dk}_g) \leftarrow \mathsf{UTE.Gen}(1^\lambda)$.
   (b) If g is an AND gate, sample values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ uniformly from $\{0,1\}^\ell$ such that $\mathsf{val}(w_1) \oplus \cdots \oplus \mathsf{val}(w_{k_{\mathsf{in}}}) = \mathsf{dk}_g$. If g is an OR gate, set all values $\mathsf{val}(w_1), \ldots, \mathsf{val}(w_{k_{\mathsf{in}}})$ to dk_g.
   (c) For gates $g \neq g^*$: if (i', g) is such that $g \in L_{V_i}$ for some $i \in [\ell]$ and additionally $i' \in V_i$, compute $\left|\mathsf{ct}_{i',j}^{(g)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g, 0)$ for $j \in [k_{\mathsf{out}}(g)]$. Otherwise, compute $\left|\mathsf{ct}_{i',j}^{(g)}\right\rangle \leftarrow \mathsf{UTE.Enc}(\mathsf{ek}_g, \mathsf{val}(w_j'))$ for $j \in [k_{\mathsf{out}}(g)]$.
4. Adversary $\mathcal{A}'$ constructs $|s_i\rangle$ as in Equation (4) and gives $|s_{V_i}\rangle$ to $\mathcal{A}_i$ for $i \in [\ell]$ to get st_i. Adversary $\mathcal{A}'$ gives $(\mathsf{st}_1, \ldots, \mathsf{st}_\ell, |s_{V^*}\rangle)$ to $\mathcal{B}$ and outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is QPT if $(\mathcal{A}, \mathcal{B})$ is. If the secret sharing challenger shares m_0, $\mathcal{A}'$ perfectly simulates $\mathsf{Hyb}_{\ell,t}^{(0)}(\mathcal{A}, \mathcal{B})$. If the secret sharing challenger shares m_1, $\mathcal{A}'$ perfectly simulates $\mathsf{Hyb}_{\ell,t}^{(1)}(\mathcal{A}, \mathcal{B})$. Thus, $\mathcal{A}'$ has advantage δ' in the secret sharing security game, a contradiction. □

Combining Lemmata 6.7 and 6.8 proves the theorem by a hybrid argument. □

Corollary 6.9 (UJLR Secret Sharing)**.** *Assuming the existence of post-quantum one-way functions, there exists a secure UJLR secret sharing scheme for all polynomial-size monotone Boolean circuits.*

Remark 6.10 (UJLRSS for Boolean Formulas). One can simplify Construction 6.4 to get UJLRSS for Boolean formulas from only query-bounded collusion-resistant UTE, which can be constructed from pseudorandom states. As sketched in Section 1.2, each share is a classical share of dk for the policy P along with a UTE ciphertext encrypting a classical share of m for P. Since secret sharing for Boolean formulas can be constructed with perfect information theoretic security, this allows any unqualified set of shares of dk to be perfectly simulated when appealing to UTE security for each group in the partition.

7 Untelegraphable Functional Encryption

In this section, we define and construct untelegraphable functional encryption (UTFE). Particularly, we construct both public-key and secret-key functional encryption with untelegraphable ciphertexts. We then sketch a simple way to leverage UTFE to construct untelegraphable differing inputs obfuscation by additionally working in the classical oracle model.

7.1 Building Blocks and Definitions

In this section, we define the notions that will be useful for our constructions of public-key and secret-key UTFE.

Pseudorandom function. We start by recalling the notion of a pseudorandom function (PRF).

Definition 7.1 (Pseudorandom Function)**.** Let $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda\in\mathbb{N}}$, $\mathcal{X} = \{\mathcal{X}_\lambda\}_{\lambda\in\mathbb{N}}$, and $\mathcal{Y} = \{\mathcal{Y}_\lambda\}_{\lambda\in\mathbb{N}}$ be ensembles of finite sets indexed by a security parameter λ. Let $\mathsf{F} = \{\mathsf{F}_\lambda\}_{\lambda\in\mathbb{N}}$ be an efficiently-computable collection of functions $\mathsf{F}_\lambda : \mathcal{K}_\lambda \times \mathcal{X}_\lambda \to \mathcal{Y}_\lambda$. We say that F is secure if for all efficient adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$:

$$\left|\Pr\left[\mathcal{A}^{\mathsf{F}_\lambda(k,\cdot)}(1^\lambda) = 1 : k \xleftarrow{R} \mathcal{K}_\lambda\right] - \Pr\left[\mathcal{A}^{f_\lambda(\cdot)}(1^\lambda) = 1 : f_\lambda \xleftarrow{R} \mathsf{Funs}[\mathcal{X}_\lambda, \mathcal{Y}_\lambda]\right]\right| = \mathsf{negl}(\lambda),$$

where $\mathsf{Funs}[\mathcal{X}_\lambda, \mathcal{Y}_\lambda]$ is the set of all functions from $\mathcal{X}_\lambda$ to $\mathcal{Y}_\lambda$.

Single-decryptor functional encryption. We now recall the definition of single-decryptor functional encryption (SDFE) [KN22, ÇG24b]. This is the natural generalization of single-decryptor encryption to the functional encryption setting. We define the challenge-only definition from [KN22], since it is sufficient for the constructions in this work. We omit the definition of testing a γ-good FE distinguisher since we only use a basic property of it in Corollary 7.5 and nowhere else. We refer to [KN22, Definition 7.29] for the full definition along with [KN22, Section 7.1] for the corresponding preliminaries.

Definition 7.2 (Single-Decryptor Functional Encryption)**.** A single-decryptor functional encryption scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and function space $\mathcal{F} = \{\mathcal{F}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{SDFE}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with the following syntax:

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{pk}, \mathsf{msk})$: On input the security parameter λ, the setup algorithm outputs a classical public key pk and master secret key msk.
- $\mathsf{KeyGen}(\mathsf{msk}, f) \to |\mathsf{sk}_f\rangle$: On input the master key msk and a function $f \in \mathcal{F}_\lambda$, the key generation algorithm outputs a quantum function secret key $|\mathsf{sk}_f\rangle$.
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$: On input the public key pk and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm outputs a classical ciphertext ct.
- $\mathsf{Dec}(|\mathsf{sk}_f\rangle, \mathsf{ct}) \to m$: On input a quantum function secret key $|\mathsf{sk}_f\rangle$ and a ciphertext ct, the decryption algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{SDFE} satisfy the following properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and $f \in \mathcal{F}_\lambda$,

$$\Pr\left[f(m) = y : \begin{array}{c}(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, m) \\ |\mathsf{sk}_f\rangle \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f) \\ y \leftarrow \mathsf{Dec}(|\mathsf{sk}_f\rangle, \mathsf{ct})\end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

- **γ-Anti-Piracy Security:** For a security parameter λ, an advantage parameter $\gamma \in [0, 1]$, and an adversary $\mathcal{A}$, we define the anti-piracy security game as follows:
  1. The challenger starts by sampling $(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and giving pk to adversary $\mathcal{A}$.
  2. Adversary $\mathcal{A}$ outputs a function $f \in \mathcal{F}_\lambda$. The challenger samples $|\mathsf{sk}_f\rangle \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ and gives $|\mathsf{sk}_f\rangle$ to $\mathcal{A}$.
  3. Adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$ along with two second stage adversaries $\mathcal{B} = (\rho[\mathsf{R}_\mathcal{B}], \boldsymbol{U}_\mathcal{B})$ and $\mathcal{C} = (\rho[\mathsf{R}_\mathcal{C}], \boldsymbol{U}_\mathcal{C})$, where ρ is a quantum state over registers $\mathsf{R}_\mathcal{B}$ and $\mathsf{R}_\mathcal{C}$, and $\boldsymbol{U}_\mathcal{B}$ and $\boldsymbol{U}_\mathcal{C}$ are general quantum circuits.
  4. The challenger runs the test for a γ-good FE distinguisher with respect to $(\mathsf{pk}, m_0, m_1, f)$ on $\mathcal{B}$ and $\mathcal{C}$. The challenger outputs $b = 1$ if both tests pass and $b = 0$ otherwise.

We say an SDFE scheme satisfies γ-anti-piracy security if for all QPT adversaries $\mathcal{A}$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, $\Pr[b = 1] \leq \mathsf{negl}(\lambda)$ in the above security game.

Theorem 7.3 (SDFE [KN22])**.** *Assuming sub-exponentially secure indistinguishability obfuscation and one-way functions, and the quantum hardness of learning with errors (LWE), there exists an SDFE scheme for* $\boldsymbol{P/poly}$ *that satisfies γ-anti-piracy security for any inverse polynomial $\gamma = \gamma(\lambda)$.*

Functional encryption with untelegraphable keys. We also define the untelegraphable analog of SDFE, which we call functional encryption with untelegraphable keys (FE-UTK) and show that SDFE implies FE-UTK.

Definition 7.4 (FE with Untelegraphable Keys)**.** An FE-UTK scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and function space $\mathcal{F} = \{\mathcal{F}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{FEUTK}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with the same syntax and correctness requirement as Definition 7.2. We require that Π_{FEUTK} satisfy the following security property:

- **UTK Security:** For a security parameter λ, a bit $b \in \{0, 1\}$, and a two-stage adversary $(\mathcal{A}, \mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1))$, we define the UTK security game as follows:

  1. The challenger starts by sampling $(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and giving pk to adversary $\mathcal{A}$.
  2. Adversary $\mathcal{A}$ outputs a function $f \in \mathcal{F}_\lambda$. The challenger samples $|\mathsf{sk}_f\rangle \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ and gives $|\mathsf{sk}_f\rangle$ to $\mathcal{A}$. Adversary $\mathcal{A}$ outputs a classical string st, which is then given to $\mathcal{B}$.
  3. On input st, adversary $\mathcal{B}_0$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$ along with a state $\mathsf{st}_{\mathcal{B}}$, which is given to $\mathcal{B}_1$.
  4. The challenger computes $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$ and gives ct to $\mathcal{B}_1$. Adversary $\mathcal{B}_1$ outputs a bit b', which is the output of the experiment.

  We say a FE-UTK scheme satisfies UTK security if for all QPT adversaries $(\mathcal{A}, \mathcal{B})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

  $$|\Pr[b' = 1|b = 0] - \Pr[b' = 1|b = 1]| = \mathsf{negl}(\lambda)$$

  in the above security game.

Corollary 7.5 (Anti-Piracy Security Implies UTK Security)**.** *Suppose a scheme Π_{SDFE} satisfies anti-piracy security. Then, Π_{SDFE} satisfies UTK security.*

Proof. Given an adversary $(\mathcal{A}, \mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1))$ that wins the UTK security game with non-negligible advantage $\delta > 0$, we construct $\mathcal{A}'$ that wins the γ-anti-piracy game for inverse polynomial $\gamma(\lambda)$:

1. Adversary $\mathcal{A}'$ runs $\mathcal{A}$ and gives st to $\mathcal{B}_0$ to get $\mathsf{st}_{\mathcal{B}}$ and messages m_0, m_1.
2. $\mathcal{A}'$ defines both $\mathcal{B}' = \mathcal{C}' = (\mathsf{st}_{\mathcal{B}}, \boldsymbol{U}_{\mathcal{B}_1})$ and gives them to the challenger.

By assumption, the probability of $(\mathcal{A}, \mathcal{B})$ guessing a random challenge bit is $1/2 + \delta$, where for infinitely many $\lambda \in \mathbb{N}$, the advantage $\delta(\lambda) \geq 1/p(\lambda)$ for some polynomial p. Thus, on such values of λ, the challenger's distinguisher test will pass with inverse polynomial probability by setting $\gamma(\lambda) = 1/p'(\lambda)$ for $p'(\lambda) = 100p(\lambda)$, say. Thus, for infinitely many $\lambda \in \mathbb{N}$, the γ-anti-piracy game will output 1 with some inverse polynomial probability, which is non-negligible. □

Untelegraphable functional encryption. We now formally define the notions of untelegraphable secret-key and public-key functional encryption (abbreviated UTSKFE and UTPKFE, respectively). These definitions extend plain public-key and secret-key functional encryption to have quantum ciphertexts which satisfy untelegraphability. In this work, we consider a relaxation of the standard untelegraphable security definition that would give out the master key to the second-stage adversary. Specifically, the alternate security game allows the second-stage adversary to make queries to *any* function secret key, even one that distinguishes the challenge messages. This relaxation is similar to the notion of SDFE, which has no master key it can give out since the ciphertext is the additional information given to the second-stage adversaries.

Definition 7.6 (Untelegraphable Secret-Key Functional Encryption)**.** An untelegraphable secret-key functional encryption scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and function space $\mathcal{F} = \{\mathcal{F}_\lambda\}_{\lambda\in\mathbb{N}}$ is a tuple of QPT algorithms $\Pi_{\mathsf{UTSKFE}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with the following syntax:

- $\mathsf{Setup}(1^\lambda) \to \mathsf{msk}$: On input the security parameter λ, the setup algorithm outputs a classical master secret key msk.
- $\mathsf{KeyGen}(\mathsf{msk}, f) \to \mathsf{sk}_f$: On input the master key msk and a function $f \in \mathcal{F}_\lambda$, the key generation algorithm outputs a function secret key sk_f.
- $\mathsf{Enc}(\mathsf{msk}, m) \to |\mathsf{ct}\rangle$: On input the master key msk and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm outputs a quantum ciphertext $|\mathsf{ct}\rangle$.
- $\mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle) \to m$: On input a function secret key sk_f and a quantum ciphertext $|\mathsf{ct}\rangle$, the decryption algorithm outputs a message $m \in \mathcal{M}_\lambda$.

We require that Π_{UTSKFE} satisfy the following properties:

- **Correctness:** For all $\lambda \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and $f \in \mathcal{F}_\lambda$,

$$\Pr\left[f(m) = y : \begin{array}{c} \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ |\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{msk}, m) \\ \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f) \\ y \leftarrow \mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

- **Untelegraphable Security with Unrestricted Key Queries:** For a security parameter λ, a bit $b \in \{0, 1\}$, and a two-stage adversary $(\mathcal{A}, \mathcal{B})$, we define the untelegraphable security game as follows:
  1. At the beginning of the game, the challenger samples $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$.
  2. Adversary $\mathcal{A}$ can now make key-generation queries by specifying a function $f \in \mathcal{F}_\lambda$. The challenger replies with $\mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$.
  3. Adversary $\mathcal{A}$ can also make encryption queries by specifying a message $m \in \mathcal{M}_\lambda$. The challenger replies with $|\mathsf{ct}\rangle \leftarrow \mathsf{Enc}(\mathsf{msk}, m)$.
  4. Adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$. If $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, the challenger generates a ciphertext $|\mathsf{ct}^*\rangle \leftarrow \mathsf{Enc}(\mathsf{msk}, m_b)$ and sends $|\mathsf{ct}^*\rangle$ to $\mathcal{A}$.
  5. Adversary $\mathcal{A}$ is again allowed to make key-generation queries, so long as $f(m_0) = f(m_1)$ holds and can still make encryption queries as before.
  6. Adversary $\mathcal{A}$ outputs a classical string st which is given to adversary $\mathcal{B}$.
  7. Adversary $\mathcal{B}$ can now make encryption queries and key-generation queries where the function $f \in \mathcal{F}_\lambda$ is not restricted. Adversary $\mathcal{B}$ outputs a bit $b_{\mathcal{B}}$, which is the output of the experiment.

  We say a UTSKFE scheme Π_{UTSKFE} satisfies untelegraphable security if for all QPT adversaries $(\mathcal{A}, \mathcal{B})$ there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$,

$$|\Pr[b_{\mathcal{B}} = 1|b = 0] - \Pr[b_{\mathcal{B}} = 1|b = 1]| = \mathsf{negl}(\lambda)$$

  in the above security game. We say a scheme Π_{UTSKFE} satisfies untelegraphable security *with master key revealing* if security still holds when $\mathcal{B}$ is given msk instead of key-generation access. We say a scheme Π_{UTSKFE} is *single-ciphertext secure* if the adversary only gets to make the challenge ciphertext query and does not otherwise have access to an encryption oracle.

Definition 7.7 (Untelegraphable Public-Key Functional Encryption)**.** We define untelegraphable PKFE the same as untelegraphable SKFE (Definition 7.6), with the following basic modifications:

- The setup algorithm Setup now outputs a public key and master key pair $(\mathsf{pk}, \mathsf{msk})$.
- The encryption algorithm Enc now takes as input the public key pk instead of the master key msk.
- Instead of making encryption queries, the adversary gets the public key pk from the challenger at the start of the security game.

Plain PKFE and SKFE. We also use plain public-key and secret-key FE in our constructions. The adaptive security definitions are the same as Definition 7.6 except the first stage adversary outputs a bit b'. The syntax is also the same except ciphertexts are classical. We omit the definitions due to redundancy.

7.2 Constructing Untelegraphable Secret-Key Functional Encryption

In this section, we construct single-ciphertext secure untelegraphable SKFE from FE-UTK (Definition 7.4) and standard SKFE.

Construction 7.8 (Untelegraphable SKFE)**.** Let λ be a security parameter. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{FEUTK}} = \mathsf{FEUTK}.(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme with untelegraphable keys with function space $\mathbf{P/poly}$.
- Let $\Pi_{\mathsf{SKFE}} = \mathsf{SKFE}.(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a secret-key FE scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and function space $\mathcal{F} = \{\mathcal{F}_\lambda\}_{\lambda\in\mathbb{N}}$.

We construct our UTSKFE scheme $\Pi_{\mathsf{UTSKFE}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Setup}(1^\lambda)$: On input the security parameter λ, the setup algorithm samples

$$(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}) \leftarrow \mathsf{FEUTK.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda),$$

  and outputs $\mathsf{msk} = (\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{SKFE}})$.
- $\mathsf{KeyGen}(\mathsf{msk}, f)$: On input the master key $\mathsf{msk} = (\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{SKFE}})$ and a function $f \in \mathcal{F}_\lambda$, the key generation algorithm samples $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f)$ and outputs $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_f)$.
- $\mathsf{Enc}(\mathsf{msk}, m)$: On input the master key $\mathsf{msk} = (\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{SKFE}})$ and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm computes $\mathsf{ct}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m)$ and outputs

$$|\mathsf{ct}\rangle \leftarrow \mathsf{FEUTK.KeyGen}(\mathsf{msk}_{\mathsf{FEUTK}}, G[\mathsf{ct}_{\mathsf{SKFE}}]),$$

  where $G[\mathsf{ct}_{\mathsf{SKFE}}]$ is the circuit which takes an SKFE function key sk'_f as input and outputs $\mathsf{SKFE.Dec}(\mathsf{ct}_{\mathsf{SKFE}}, \mathsf{sk}'_f)$.
- $\mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle)$: On input a function secret key sk_f and a quantum ciphertext $|\mathsf{ct}\rangle$, the decryption algorithm outputs $\mathsf{FEUTK.Dec}(|\mathsf{ct}\rangle, \mathsf{sk}_f)$.

Theorem 7.9 (Correctness)**.** *If Π_{FEUTK} and Π_{SKFE} are correct, then Construction 7.8 is correct.*

Proof. Take any $\lambda \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and $f \in \mathcal{F}_\lambda$. Let $\mathsf{msk} = (\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{SKFE}}) \leftarrow \mathsf{Setup}(1^\lambda)$. As in Enc, compute $\mathsf{ct}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m)$ and sample $|\mathsf{ct}\rangle \leftarrow \mathsf{FEUTK.KeyGen}(\mathsf{msk}_{\mathsf{FEUTK}}, G[\mathsf{ct}_{\mathsf{SKFE}}])$. As in KeyGen, sample $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f)$ and set $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_f)$. We consider the output of $\mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle)$. By correctness of Π_{FEUTK}, we have $G[\mathsf{ct}_{\mathsf{SKFE}}](\mathsf{sk}'_f) \leftarrow \mathsf{FEUTK.Dec}(|\mathsf{ct}\rangle, \mathsf{sk}_f)$. By correctness of Π_{SKFE}, we have $f(m) \leftarrow G[\mathsf{ct}_{\mathsf{SKFE}}](\mathsf{sk}'_f)$, as desired. □

Theorem 7.10 (Untelegraphable Security)**.** *Suppose Π_{SKFE} satisfies adaptive security and Π_{FEUTK} satisfies UTK security (Definition 7.4). Then, Construction 7.8 satisfies single-ciphertext untelegraphable security with unrestricted key queries (Definition 7.6).*

Proof. Suppose there exists a QPT two-stage adversary $(\mathcal{A}, \mathcal{B})$ that has non-negligible advantage δ in the UTSKFE security game and adversary $\mathcal{B}$ makes at most q queries. We define a sequence of hybrid experiments indexed by a bit $i \in [q]$:

- $\mathsf{Hyb}_0^{(b)}$: This is the UTSKFE security game with challenge bit $b \in \{0,1\}$. In particular:
  - The challenger starts by sampling
    $$(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}) \leftarrow \mathsf{FEUTK.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda).$$
    It sets $\mathsf{msk} = (\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{SKFE}})$.
  - When adversary $\mathcal{A}$ makes a key query on function $f \in \mathcal{F}_\lambda$, the challenger samples
    $$\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f)$$
    and replies with $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_f)$.
  - When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, the challenger computes $\mathsf{ct}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$ and replies with
    $$|\mathsf{ct}\rangle \leftarrow \mathsf{FEUTK.KeyGen}(\mathsf{msk}_{\mathsf{FEUTK}}, G[\mathsf{ct}_{\mathsf{SKFE}}]).$$
  - Adversary $\mathcal{A}$ outputs a classical state st after it is finished making key queries such that $f(m_0) = f(m_1)$. The challenger answers the queries as before and gives st to $\mathcal{B}$.
  - When adversary $\mathcal{B}$ makes a key query, the challenger answers as it did for $\mathcal{A}$, without checking $f(m_0) = f(m_1)$. Adversary $\mathcal{B}$ outputs a bit b', which is the output of the experiment.
- $\mathsf{Hyb}_i^{(b)}$: Same as $\mathsf{Hyb}_{i-1}^{(b)}$ except when answering adversary $\mathcal{B}$'s i^{th} key query on function f_i, the adversary replies with $\mathsf{sk}_{f_i} \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, 0)$, where we overload 0 to mean the all 0s string.

We write $\mathsf{Hyb}_i^{(b)}(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of $\mathsf{Hyb}_i^{(b)}$ with adversary $(\mathcal{A}, \mathcal{B})$. We now argue that each adjacent pair of distributions are indistinguishable.

Lemma 7.11. *Suppose Π_{FEUTK} is UTK secure. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*
$$\left|\Pr\left[\mathsf{Hyb}_{i-1}^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_i^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $(\mathcal{A}', \mathcal{B}' = (\mathcal{B}'_0, \mathcal{B}'_1))$ that breaks UTK security as follows:

1. At the start of the game, adversary $\mathcal{A}'$ gets $\mathsf{pk}_{\mathsf{FEUTK}}$ from its challenger. Adversary $\mathcal{A}'$ samples $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$.
2. When adversary $\mathcal{A}$ makes a key query on $f \in \mathcal{F}_\lambda$, adversary $\mathcal{A}'$ samples $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f)$ and replies with $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_f)$.
3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ computes $\mathsf{ct}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$ and queries the function $G[\mathsf{ct}_{\mathsf{SKFE}}]$ to the UTK challenger to get back a quantum state that it labels $|\mathsf{ct}\rangle$ and gives to $\mathcal{A}$.
4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversary $\mathcal{B}$ and $(\mathsf{st}, \mathsf{pk}_{\mathsf{FEUTK}})$ to $\mathcal{B}'_0$.
5. Adversary $\mathcal{B}'_0$ starts running adversary $\mathcal{B}$ with randomness r. For $\mathcal{B}$'s j^{th} key queries where $j < i$ on f_j, adversary $\mathcal{B}'_0$ replies with $\mathsf{sk}_{f_j} \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, 0)$. For the ith key query f_i, adversary $\mathcal{B}'_0$ samples $\mathsf{sk}'_{f_i} \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f_i)$ and outputs the message pair $(0, \mathsf{sk}'_{f_i})$ along with $\mathsf{st}_{\mathcal{B}} = (\mathsf{st}, \mathsf{pk}_{\mathsf{FEUTK}}, r, \{\mathsf{sk}_{f_j}\}_{j \in [i-1]})$. Adversary $\mathcal{B}'_1$ gets sk_{f_i} from the UTK challenger along with $\mathsf{st}_{\mathcal{B}}$, reruns $\mathcal{B}$ with randomness r, answers the first $i-1$ key queries as $\mathcal{B}'_0$ did, and answers the ith query with sk_{f_i}. For the remaining key queries on f_j, adversary $\mathcal{B}'_1$ samples $\mathsf{sk}'_{f_j} \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f_j)$ and replies with $\mathsf{sk}_{f_j} \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_{f_j})$.

6. Adversary $\mathcal{B}'_1$ outputs whatever $\mathcal{B}$ outputs.

Clearly $(\mathcal{A}', \mathcal{B}' = (\mathcal{B}'_0, \mathcal{B}'_1))$ is efficient if $(\mathcal{A}, \mathcal{B})$ is. If sk_{f_i} is an encryption of sk'_{f_i}, adversary $(\mathcal{A}', \mathcal{B}')$ simulates $\mathsf{Hyb}^{(b)}_{i-1}(\mathcal{A}, \mathcal{B})$. If sk_{f_i} is an encryption of 0, adversary $(\mathcal{A}', \mathcal{B}')$ simulates $\mathsf{Hyb}^{(b)}_{i}(\mathcal{A}, \mathcal{B})$. Thus, adversary $(\mathcal{A}', \mathcal{B}')$ breaks UTK security with advantage δ', a contradiction. □

Lemma 7.12. *Suppose Π_{SKFE} is adaptively secure for QPT adversaries. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}^{(0)}_q(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}^{(1)}_q(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks SKFE security as follows:

1. Adversary $\mathcal{A}'$ starts by sampling $(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{msk}_{\mathsf{FEUTK}}) \leftarrow \mathsf{FEUTK.Setup}(1^\lambda)$.
2. When adversary $\mathcal{A}$ makes a key query on f, adversary $\mathcal{A}'$ forwards it to the SKFE challenger to get sk'_f and replies with $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, \mathsf{sk}'_f)$.
3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ sends m_0, m_1 to the SKFE challenger to get back $\mathsf{ct}_{\mathsf{SKFE}}$, and replies with $|\mathsf{ct}\rangle \leftarrow \mathsf{FEUTK.KeyGen}(\mathsf{msk}_{\mathsf{FEUTK}}, G[\mathsf{ct}_{\mathsf{SKFE}}])$.
4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversary $\mathcal{B}$.
5. Adversary $\mathcal{A}'$ replies to all key queries by $\mathcal{B}$ with $\mathsf{sk}_f \leftarrow \mathsf{FEUTK.Enc}(\mathsf{pk}_{\mathsf{FEUTK}}, 0)$, and outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is efficient if $(\mathcal{A}, \mathcal{B})$ is. If $\mathsf{ct}_{\mathsf{SKFE}}$ is an encryption of m_0, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}^{(0)}_q(\mathcal{A}, \mathcal{B})$. If $\mathsf{ct}_{\mathsf{SKFE}}$ is an encryption of m_1, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}^{(1)}_q(\mathcal{A}, \mathcal{B})$. Thus, adversary $\mathcal{A}'$ breaks SKFE security with advantage δ', a contradiction. □

Combining Lemmata 7.11 and 7.12, the theorem follows by a standard hybrid argument. □

Corollary 7.13 (Untelegraphable SKFE)**.** *Assuming sub-exponentially secure indistinguishability obfuscation and one-way functions, and the quantum hardness of learning with errors (LWE), there exists a PKFE scheme for **P**/**poly** that satisfies untelegraphable security with unrestricted key queries.*

Proof. Follows immediately by combining Theorems 7.3, 7.9 and 7.10 and Corollary 7.5, and noting that indistinguishability obfuscation and one-way functions imply SKFE. □

7.3 Constructing Untelegraphable Public-Key Functional Encryption

In this section, we construct untelegraphable PKFE from untelegraphable SKFE, standard PKFE, and CPA-secure SKE. The construction and security proof follow the blueprint in [ABSV15].

Construction 7.14 (Untelegraphable PKFE)**.** Let λ be a security parameter. Our construction relies on the following additional primitives:

- Let $\Pi_{\mathsf{PKFE}} = \mathsf{PKFE}.(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key FE scheme with function space $\mathbf{P}/\mathbf{poly}$.
- Let $\Pi_{\mathsf{SKFE}} = \mathsf{SKFE}.(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an untelegraphable SKFE scheme with message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and function space $\mathcal{F} = \{\mathcal{F}_\lambda\}_{\lambda\in\mathbb{N}}$. Let $\mathsf{SKFE.KeyGen}$ have randomness space $\{0,1\}^{\rho(\lambda)}$.
- Let $\Pi_{\mathsf{SKE}} = \mathsf{SKE}.(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a secret-key encryption scheme.
- Let $\mathsf{F} = \{\mathsf{F}_\lambda\}_{\lambda\in\mathbb{N}}$, where $\mathsf{F}_\lambda : \mathcal{K}_\lambda \times \mathcal{F}_\lambda \to \{0,1\}^{\rho(\lambda)}$ be a pseudorandom function with key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda\in\mathbb{N}}$.

We construct our UTPKFE scheme $\Pi_{\mathsf{UTPKFE}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{Setup}(1^\lambda)$: On input the security parameter λ, the setup algorithm samples

$$(\mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk}_{\mathsf{PKFE}}) \leftarrow \mathsf{PKFE.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Gen}(1^\lambda),$$

  and outputs $(\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}}))$.

- $\mathsf{KeyGen}(\mathsf{msk}, f)$: On input the master key $\mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}})$ and a function $f \in \mathcal{F}_\lambda$, the key generation algorithm computes $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, 0)$ and outputs

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]),$$

  where $G[\mathsf{ct}_{\mathsf{SKE}}, f]$ is a circuit that takes as input a bit $b \in \{0,1\}$, three keys $\mathsf{msk}_{\mathsf{SKFE}}$, k, and $\mathsf{sk}_{\mathsf{SKE}}$, and outputs $\mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; \mathsf{F}_\lambda(k, f))$ if $b = 0$ or $\mathsf{SKE.Dec}(\mathsf{sk}_{\mathsf{SKE}}, \mathsf{ct}_{\mathsf{SKE}})$ if $b = 1$.

- $\mathsf{Enc}(\mathsf{pk}, m)$: On input the public key $\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}$ and a message $m \in \mathcal{M}_\lambda$, the encryption algorithm samples $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$ and $k \xleftarrow{R} \mathcal{K}_\lambda$, computes $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m)$ and

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)),$$

  and outputs $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

- $\mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle)$: On input a function secret key sk_f and a quantum ciphertext $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$, the decryption algorithm computes $\mathsf{sk}'_f \leftarrow \mathsf{PKFE.Dec}(\mathsf{sk}_f, \mathsf{ct}_{\mathsf{PKFE}})$ and outputs $\mathsf{SKFE.Dec}(\mathsf{sk}'_f, |\mathsf{ct}_{\mathsf{SKFE}}\rangle)$.

Theorem 7.15 (Correctness)**.** *If Π_{SKFE} and Π_{PKFE} are correct and F is secure, then Construction 7.14 is correct.*

Proof. Take any $\lambda \in \mathbb{N}$, $m \in \mathcal{M}_\lambda$, and $f \in \mathcal{F}_\lambda$. Let $(\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}})) \leftarrow \mathsf{Setup}(1^\lambda)$. As in Enc, let $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$, $k \xleftarrow{R} \mathcal{K}_\lambda$, $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m)$,

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)),$$

and set $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$. As in KeyGen, let $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, 0)$ and

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]).$$

We consider the output of $\mathsf{Dec}(\mathsf{sk}_f, |\mathsf{ct}\rangle)$. By definition of $G[\mathsf{ct}_{\mathsf{SKE}}, f]$ and Π_{PKFE} correctness, we have $\mathsf{sk}'_f \leftarrow \mathsf{PKFE.Dec}(\mathsf{sk}_f, \mathsf{ct}_{\mathsf{PKFE}})$, where $\mathsf{sk}'_f = \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; \mathsf{F}_\lambda(k, f))$. By security of F and Π_{SKFE} correctness, we have $m \leftarrow \mathsf{SKFE.Dec}(\mathsf{sk}'_f, |\mathsf{ct}_{\mathsf{SKFE}}\rangle)$ with overwhelming probability, as desired. □

Theorem 7.16 (Untelegraphable Security)**.** *Suppose Π_{PKFE} satisfies adaptive security for QPT adversaries, F is a secure PRF for QPT adversaries, Π_{SKE} satisfies CPA-security for QPT adversaries and correctness, and Π_{SKFE} satisfies single-ciphertext untelegraphable security with unrestricted key queries (Definition 7.6). Then, Construction 7.14 satisfies untelegraphable security with unrestricted key queries (Definition 7.7).*

Proof. Suppose there exists a QPT two-stage adversary $(\mathcal{A}, \mathcal{B})$ that has non-negligible advantage δ in the UTPKFE security game. We define a sequence of hybrid experiments:

- $\mathsf{Hyb}_0^{(b)}$: This is the UTPKFE security game with challenge bit $b \in \{0,1\}$. In particular:
  - The challenger starts by sampling

$$(\mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk}_{\mathsf{PKFE}}) \leftarrow \mathsf{PKFE.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Gen}(1^\lambda),$$

    sets $(\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}}))$, and gives pk to adversary $\mathcal{A}$.

– When adversary $\mathcal{A}$ makes a key query on function $f \in \mathcal{F}_\lambda$, the challenger computes $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, 0)$ and replies with

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]).$$

– When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, the challenger samples $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$ and $k \xleftarrow{R} \mathcal{K}_\lambda$, computes $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$ and

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)),$$

and replies with $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

– Adversary $\mathcal{A}$ outputs a classical state st after it is finished making key queries such that $f(m_0) = f(m_1)$. The challenger answers the queries as before and gives st to $\mathcal{B}$.

– When adversary $\mathcal{B}$ makes a key query, the challenger answers as it did for $\mathcal{A}$, without checking $f(m_0) = f(m_1)$. Adversary $\mathcal{B}$ outputs a bit b', which is the output of the experiment.

- $\mathsf{Hyb}_1^{(b)}$: Same as $\mathsf{Hyb}_0^{(b)}$ except the challenger now samples $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$ and $k \xleftarrow{R} \mathcal{K}_\lambda$ at the beginning of the game and for each key query on $f \in \mathcal{F}_\lambda$ it now computes $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; \mathsf{F}_\lambda(k, f))$ and $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, \mathsf{sk}'_f)$.

- $\mathsf{Hyb}_2^{(b)}$: Same as $\mathsf{Hyb}_1^{(b)}$ except the challenger now computes

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}}))$$

when constructing the challenge ciphertext.

- $\mathsf{Hyb}_3^{(b)}$: Same as $\mathsf{Hyb}_2^{(b)}$ except for each key query on $f \in \mathcal{F}_\lambda$ the challenger now samples $r_f \xleftarrow{R} \{0,1\}^{\rho(\lambda)}$ and computes $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; r_f)$. Note that r_f is reused if f is queried more than once.

We write $\mathsf{Hyb}_i^{(b)}(\mathcal{A}, \mathcal{B})$ to denote the output distribution of an execution of $\mathsf{Hyb}_i^{(b)}$ with adversary $(\mathcal{A}, \mathcal{B})$. We now argue that each adjacent pair of distributions are indistinguishable.

Lemma 7.17. *Suppose Π_{SKE} is CPA-secure for QPT adversaries. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_0^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks CPA-security as follows:

1. Adversary $\mathcal{A}'$ starts by sampling $(\mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk}_{\mathsf{PKFE}}) \leftarrow \mathsf{PKFE.Setup}(1^\lambda)$, $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$, and $k \xleftarrow{R} \mathcal{K}_\lambda$. It gives $\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}$ to $\mathcal{A}$.

2. When adversary $\mathcal{A}$ makes a key query on f, adversary $\mathcal{A}'$ computes

$$\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; \mathsf{F}_\lambda(k, f))$$

and submits $(0, \mathsf{sk}'_f)$ to the CPA challenger to get back $\mathsf{ct}_{\mathsf{SKE}}$. Adversary $\mathcal{A}'$ replies to $\mathcal{A}$ with

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]).$$

3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ computes $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$, computes

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)),$$

and replies with $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversary $\mathcal{B}$. When adversary $\mathcal{B}$ makes a key query, adversary $\mathcal{A}'$ answers as it did for $\mathcal{A}$, without checking $f(m_0) = f(m_1)$. Adversary $\mathcal{A}'$ outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is efficient if $(\mathcal{A}, \mathcal{B})$ is. If each $\mathsf{ct}_{\mathsf{SKE}}$ is an encryption of 0, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_0^{(b)}(\mathcal{A}, \mathcal{B})$. If each $\mathsf{ct}_{\mathsf{SKE}}$ is an encryption of sk'_f, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B})$. Thus, adversary $\mathcal{A}'$ breaks CPA-security with advantage δ', as desired. □

Lemma 7.18. *Suppose Π_{PKFE} is adaptively secure for QPT adversaries and Π_{SKE} is correct. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks PKFE security as follows:

1. At the start of the game, adversary $\mathcal{A}'$ gets $\mathsf{pk}_{\mathsf{PKFE}}$ from its challenger and gives it to $\mathcal{A}$. Adversary $\mathcal{A}'$ also samples $\mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Gen}(1^\lambda)$, $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$, and $k \xleftarrow{R} \mathcal{K}_\lambda$.

2. When adversary $\mathcal{A}$ makes a key query on f, adversary $\mathcal{A}'$ computes

$$\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; \mathsf{F}_\lambda(k, f))$$

   and $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, \mathsf{sk}'_f)$. Adversary $\mathcal{A}'$ queries its challenger for $G[\mathsf{ct}_{\mathsf{SKE}}, f]$ to get back sk_f, which it given to $\mathcal{A}$.

3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ computes $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$, submits the pair of messages

$$(0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot) \quad \text{and} \quad (1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})$$

   to its challenger to get $\mathsf{ct}_{\mathsf{PKFE}}$, and replies with $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversary $\mathcal{B}$. When adversary $\mathcal{B}$ makes a key query, adversary $\mathcal{A}'$ answers as it did for $\mathcal{A}$, without checking $f(m_0) = f(m_1)$. Adversary $\mathcal{A}'$ outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is efficient if $(\mathcal{A}, \mathcal{B})$ is. If $\mathsf{ct}_{\mathsf{PKFE}}$ is an encryption of $(0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot)$, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_1^{(b)}(\mathcal{A}, \mathcal{B})$. If $\mathsf{ct}_{\mathsf{PKFE}}$ is an encryption of $(1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})$, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B})$. Adversary $\mathcal{A}'$ is admissible, since for all functions f, we have

$$G[\mathsf{ct}_{\mathsf{SKE}}, f](0, \mathsf{msk}_{\mathsf{SKFE}}, k, \bot) = \mathsf{sk}'_f = G[\mathsf{ct}_{\mathsf{SKE}}, f](1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}}),$$

by correctness ofΠ_{SKE}. Thus, adversary $\mathcal{A}'$ breaks PKFE security with advantage δ', as desired. □

Lemma 7.19. *Suppose F is a secure PRF for QPT adversaries. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_3^{(b)}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $\mathcal{A}'$ that breaks PRF security as follows:

1. Adversary $\mathcal{A}'$ starts by sampling $\mathsf{msk}_{\mathsf{SKFE}} \leftarrow \mathsf{SKFE.Setup}(1^\lambda)$,

$$(\mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk}_{\mathsf{PKFE}}) \leftarrow \mathsf{PKFE.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Gen}(1^\lambda),$$

   sets $(\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}}))$, and gives pk to adversary $\mathcal{A}$.

2. When adversary $\mathcal{A}$ makes a key query on function $f \in \mathcal{F}_\lambda$, adversary $\mathcal{A}'$ queries its challenger on f to get r_f, computes $\mathsf{sk}'_f \leftarrow \mathsf{SKFE.KeyGen}(\mathsf{msk}_{\mathsf{SKFE}}, f; r_f)$ and $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, \mathsf{sk}'_f)$, and replies with

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]).$$

3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ computes $|\mathsf{ct}_{\mathsf{SKFE}}\rangle \leftarrow \mathsf{SKFE.Enc}(\mathsf{msk}_{\mathsf{SKFE}}, m_b)$, computes

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})),$$

and replies with $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversary $\mathcal{B}$. When adversary $\mathcal{B}$ makes a key query, adversary $\mathcal{A}'$ answers as it did for $\mathcal{A}$, without checking $f(m_0) = f(m_1)$. Adversary $\mathcal{A}'$ outputs whatever $\mathcal{B}$ outputs.

Clearly $\mathcal{A}'$ is efficient if $(\mathcal{A}, \mathcal{B})$ is. If each r_f is from F_λ, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_2^{(b)}(\mathcal{A}, \mathcal{B})$. If each r_f is from a random function, adversary $\mathcal{A}'$ simulates $\mathsf{Hyb}_3^{(b)}(\mathcal{A}, \mathcal{B})$. Thus, adversary $\mathcal{A}'$ breaks PRF security with advantage δ', as desired. □

Lemma 7.20. *Suppose Π_{SKFE} satisfies untelegraphable security. Then, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\lambda \in \mathbb{N}$, we have*

$$\left|\Pr\left[\mathsf{Hyb}_3^{(0)}(\mathcal{A}, \mathcal{B}) = 1\right] - \Pr\left[\mathsf{Hyb}_3^{(1)}(\mathcal{A}, \mathcal{B}) = 1\right]\right| = \mathsf{negl}(\lambda).$$

Proof. Suppose $(\mathcal{A}, \mathcal{B})$ distinguishes the experiments with non-negligible probability δ'. We use $(\mathcal{A}, \mathcal{B})$ to construct an adversary $(\mathcal{A}', \mathcal{B}')$ that breaks untelegraphable SKFE security as follows:

1. Adversary $\mathcal{A}'$ starts by sampling

$$(\mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk}_{\mathsf{PKFE}}) \leftarrow \mathsf{PKFE.Setup}(1^\lambda) \quad \text{and} \quad \mathsf{sk}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Gen}(1^\lambda),$$

sets $(\mathsf{pk} = \mathsf{pk}_{\mathsf{PKFE}}, \mathsf{msk} = (\mathsf{msk}_{\mathsf{PKFE}}, \mathsf{sk}_{\mathsf{SKE}}))$, and gives pk to adversary $\mathcal{A}$.

2. When adversary $\mathcal{A}$ makes a key query on function $f \in \mathcal{F}_\lambda$, adversary $\mathcal{A}'$ queries its challenger on f to get sk'_f, computes $\mathsf{ct}_{\mathsf{SKE}} \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}_{\mathsf{SKE}}, \mathsf{sk}'_f)$ and replies with

$$\mathsf{sk}_f \leftarrow \mathsf{PKFE.KeyGen}(\mathsf{msk}_{\mathsf{PKFE}}, G[\mathsf{ct}_{\mathsf{SKE}}, f]).$$

If f is queried multiple times, adversary $\mathcal{A}'$ reuses sk'_f.

3. When adversary $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}_\lambda$, if $f(m_0) = f(m_1)$ holds for all functions f queried by $\mathcal{A}$, adversary $\mathcal{A}'$ queries its challenger on (m_0, m_1) to get $|\mathsf{ct}_{\mathsf{SKFE}}\rangle$, computes

$$\mathsf{ct}_{\mathsf{PKFE}} \leftarrow \mathsf{PKFE.Enc}(\mathsf{pk}_{\mathsf{PKFE}}, (1, \bot, \bot, \mathsf{sk}_{\mathsf{SKE}})),$$

and replies with $|\mathsf{ct}\rangle = (|\mathsf{ct}_{\mathsf{SKFE}}\rangle, \mathsf{ct}_{\mathsf{PKFE}})$.

4. When adversary $\mathcal{A}$ outputs a classical state st, adversary $\mathcal{A}'$ gives st to adversaries $\mathcal{B}, \mathcal{B}'$. When adversary $\mathcal{B}$ makes a key query on f, adversary $\mathcal{B}'$ queries its challenger on f to get sk'_f and replies with sk_f as in step 2. Adversary $\mathcal{B}'$ outputs whatever $\mathcal{B}$ outputs.

Clearly $(\mathcal{A}', \mathcal{B}')$ is efficient and admissible if $(\mathcal{A}, \mathcal{B})$ is. If $|\mathsf{ct}_{\mathsf{SKFE}}\rangle$ is an encryption of m_0, adversary $(\mathcal{A}', \mathcal{B}')$ simulates $\mathsf{Hyb}_2^{(0)}(\mathcal{A}, \mathcal{B})$. If $|\mathsf{ct}_{\mathsf{SKFE}}\rangle$ is an encryption of m_1, adversary $(\mathcal{A}', \mathcal{B}')$ simulates $\mathsf{Hyb}_2^{(1)}(\mathcal{A}, \mathcal{B})$. Thus, adversary $(\mathcal{A}', \mathcal{B}')$ breaks untelegraphable SKFE security with advantage δ', a contradiction. □

Combining Lemmata 7.17 to 7.20, the theorem follows by a standard hybrid argument. □

Corollary 7.21 (Untelegraphable PKFE)**.** *Assuming sub-exponentially secure indistinguishability obfuscation and one-way functions, and the quantum hardness of learning with errors (LWE), there exists a PKFE scheme for* $\boldsymbol{P/poly}$ *that satisfies untelegraphable security with unrestricted key queries.*

Proof. Follows immediately by Theorems 7.15 and 7.16 and Corollary 7.13, and noting that indistinguishability obfuscation and one-way functions imply PKFE and CPA-secure SKE. □

Remark 7.22 (Untelegraphable diO). As an application of UTPKFE, we can construct untelegraphable differing inputs obfuscation in the classical oracle model. Security says given many copies of an obfuscation of C_0 or C_1, where finding x such that $C_0(x) \neq C_1(x)$ is computationally hard, it is hard to distinguish C_0 from C_1 even given a differing input x after outputting a classical telegraph of the obfuscations. An obfuscation of a circuit C is an untelegraphable encryption of message C along with a classical oracle of the key generation circuit, which on input x outputs sk_{f_x}, where f_x takes a circuit C as input and outputs $C(x)$. Security follows by switching the first stage classical oracle to one that outputs $\perp$ when queried on a differing input, and then appealing to UTPKFE security directly.

Acknowledgments

We thank Alper Çakan and Vipul Goyal for providing helpful comments on related work. Jeffrey Champion is supported by NSF CNS-2140975 and CNS-2318701.

References

[Aar19] Scott Aaronson. Shadow tomography of quantum states. *SIAM J. Comput.*, 49(5):STOC18–368, 2019. (Cited on page 2, 4, 17.)

[AB24] Prabhanjan Ananth and Amit Behera. A modular approach to unclonable cryptography. In *CRYPTO*, 2024. (Cited on page 2.)

[ABSV15] Prabhanjan Ananth, Zvika Brakerski, Gil Segev, and Vinod Vaikuntanathan. From selective to adaptive security in functional encryption. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 657–677. Springer, Berlin, Heidelberg, August 2015. (Cited on page 10, 48.)

[AGKZ20] Ryan Amos, Marios Georgiou, Aggelos Kiayias, and Mark Zhandry. One-shot signatures and applications to hybrid quantum/classical authentication. In *STOC*, page 255–268, 2020. (Cited on page 7, 26.)

[AGLL24] Prabhanjan Ananth, Vipul Goyal, Jiahui Liu, and Qipeng Liu. Unclonable secret sharing. In *ASIACRYPT*, 2024. (Cited on page 4, 37, 38.)

[AHU19] Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 269–295. Springer, Cham, August 2019. (Cited on page 11.)

[AK07] Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. In *CCC*, 2007. (Cited on page 2.)

[AK21] Prabhanjan Ananth and Fatih Kaleoglu. Unclonable encryption, revisited. Cryptology ePrint Archive, Report 2021/412, 2021. (Cited on page 12.)

[AKL+22] Prabhanjan Ananth, Fatih Kaleoglu, Xingjian Li, Qipeng Liu, and Mark Zhandry. On the feasibility of unclonable encryption, and more. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part II*, volume 13508 of *LNCS*, pages 212–241. Springer, Cham, August 2022. (Cited on page 2, 8, 13.)

[AKY24] Prabhanjan Ananth, Fatih Kaleoglu, and Henry Yuen. Simultaneous haar indistinguishability with applications to unclonable cryptography. *arXiv:2405.10274*, 2024. (Cited on page 2.)

[AN02] Dorit Aharonov and Tomer Naveh. Quantum np - a survey, 2002. (Cited on page 2.)

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 208–236. Springer, Cham, August 2022. (Cited on page 6, 16.)

[BDK+11] Boaz Barak, Yevgeniy Dodis, Hugo Krawczyk, Olivier Pereira, Krzysztof Pietrzak, François-Xavier Standaert, and Yu Yu. Leftover hash lemma, revisited. In *CRYPTO*, 2011. (Cited on page 15.)

[BL20] Anne Broadbent and Sébastien Lord. Uncloneable quantum encryption via oracles. In Steven T. Flammia, editor, *15th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2020, June 9-12, 2020, Riga, Latvia*, volume 158 of *LIPIcs*, pages 4:1–4:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. (Cited on page 2, 12, 13.)

[ÇG24a] Alper Çakan and Vipul Goyal. Unbounded leakage-resilient encryption and signatures. *IACR Cryptol. ePrint Arch.*, 2024. (Cited on page 4.)

[ÇG24b] Alper Çakan and Vipul Goyal. Unclonable cryptography with unbounded collusions and impossibility of hyperefficient shadow tomography. In *TCC*, 2024. (Cited on page 4, 43.)

[CG24c] Andrea Coladangelo and Sam Gunn. How to use quantum indistinguishability obfuscation. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *56th ACM STOC*, pages 1003–1008. ACM Press, June 2024. (Cited on page 2.)

[ÇGLR24] Alper Çakan, Vipul Goyal, Chen-Da Liu-Zhang, and João Ribeiro. Unbounded leakage-resilience and leakage-detection in a quantum world. In *TCC*, 2024. (Cited on page 4, 36, 37, 38.)

[CHK05] Ran Canetti, Shai Halevi, and Jonathan Katz. Adaptively-secure, non-interactive public-key encryption. In Joe Kilian, editor, *TCC 2005*, volume 3378 of *LNCS*, pages 150–168. Springer, Berlin, Heidelberg, February 2005. (Cited on page 5, 16.)

[Die82] D. Dieks. Communication by epr devices. *Phys. Lett. A*, 1982. (Cited on page 1.)

[DRS04] Yevgeniy Dodis, Leonid Reyzin, and Adam D. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *EUROCRYPT*, 2004. (Cited on page 15.)

[FK18] Bill Fefferman and Shelby Kimmel. Quantum vs. classical proofs and subset verification. In *Mathematical Foundations of Computer Science*, 2018. (Cited on page 2.)

[Got03] Daniel Gottesman. Uncloneable encryption. *Quantum Info. Comput.*, 2003. (Cited on page 2.)

[GW11] Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In *STOC*, 2011. (Cited on page 34.)

[HKP20] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 2020. (Cited on page 2, 4, 33.)

[HMNY21] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum encryption with certified deletion, revisited: Public key, attribute-based, and classical communication. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part I*, volume 13090 of *LNCS*, pages 606–636. Springer, Cham, December 2021. (Cited on page 5, 22.)

[HMNY22] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Certified everlasting zero-knowledge proof for QMA. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 239–268. Springer, Cham, August 2022. (Cited on page 8, 30.)

[ISV+17] Gene Itkis, Emily Shen, Mayank Varia, David Wilson, and Arkady Yerukhimovich. Bounded-collusion attribute-based encryption from minimal assumptions. In *PKC*, 2017. (Cited on page 6.)

[JKK+17] Zahra Jafargholi, Chethan Kamath, Karen Klein, Ilan Komargodski, Krzysztof Pietrzak, and Daniel Wichs. Be adaptive, avoid overcommitting. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part I*, volume 10401 of *LNCS*, pages 133–163. Springer, Cham, August 2017. (Cited on page 37, 39.)

[JL00] Stanislaw Jarecki and Anna Lysyanskaya. Adaptively secure threshold cryptography: Introducing concurrency, removing erasures. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 221–242. Springer, Berlin, Heidelberg, May 2000. (Cited on page 5, 16.)

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 126–152. Springer, Cham, August 2018. (Cited on page 4, 16.)

[KN22] Fuyuki Kitagawa and Ryo Nishimaki. Functional encryption with secure key leasing. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part IV*, volume 13794 of *LNCS*, pages 569–598. Springer, Cham, December 2022. (Cited on page 10, 43, 44.)

[KN23] Fuyuki Kitagawa and Ryo Nishimaki. One-out-of-many unclonable cryptography: Definitions, constructions, and more. In *TCC*, 2023. (Cited on page 2.)

[KNTY19] Fuyuki Kitagawa, Ryo Nishimaki, Keisuke Tanaka, and Takashi Yamakawa. Adaptively secure and succinct functional encryption: Improving security and efficiency, simultaneously. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part III*, volume 11694 of *LNCS*, pages 521–551. Springer, Cham, August 2019. (Cited on page 6, 17.)

[Kre21] W. Kretschmer. Quantum pseudorandomness and classical complexity. *TQC 2021*, 2021. (Cited on page 4, 16, 23.)

[LLPY24] Xingjian Li, Qipeng Liu, Angelos Pelecanos, and Takashi Yamakawa. Classical vs quantum advice under classically-accessible oracle. In *ITCS*, 2024. (Cited on page 2.)

[MPY24] Tomoyuki Morimae, Alexander Poremba, and Takashi Yamakawa. Revocable quantum digital signatures. In *TQC*, pages 5:1–5:24, 2024. (Cited on page 7, 26.)

[MQU07] Jörn Müller-Quade and Dominique Unruh. Long-term security and universal composability. In *TCC*, 2007. (Cited on page 8.)

[MST21] Christian Majenz, Christian Schaffner, and Mehrdad Tahmasbi. Limitations on uncloneable encryption and simultaneous one-way-to-hiding. *arXiv:2103.14510*, 2021. (Cited on page 2.)

[NN23] Anand Natarajan and Chinmay Nirkhe. A distribution testing oracle separating qma and qcma. In *CCC*, 2023. (Cited on page 2.)

[NZ24] Barak Nehoran and Mark Zhandry. A computational separation between quantum no-cloning and no-telegraphing. In *ITCS*, 2024. (Cited on page 2, 4.)

[Par70] James L. Park. The concept of transition in quantum mechanics. *Foundations of Physics*, 1970. (Cited on page 1.)

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, pages 84–93. ACM Press, 2005. (Cited on page 26.)

[VNS+03] V. Vinod, Arvind Narayanan, K. Srinathan, C. Pandu Rangan, and Kwangojo Kim. On the power of computational secret sharing. In *INDOCRYPT*, 2003. (Cited on page 37, 39.)

[VR89] K. Vogel and H. Risken. Determination of quasiprobability distributions in terms of probability distributions for the rotated quadrature phase. *Physical Review A*, 1989. (Cited on page 2.)

[Wer98] R. F. Werner. Optimal cloning of pure states. *Physical Review A*, 1998. (Cited on page 2.)

[Wie83] Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983. (Cited on page 1.)

[WZ82] William K. Wootters and Wojciech Zurek. A single quantum cannot be cloned. *Nature*, 1982. (Cited on page 1.)

[Zha12] Mark Zhandry. Secure identity-based encryption in the quantum random oracle model. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 758–775. Springer, Berlin, Heidelberg, August 2012. (Cited on page 11.)