Towards Quantum One-Time Memories from Stateless Hardware

Anne Broadbent
Department of Mathematics and Statistics, University of Ottawa, Canada
abroadbe@uottawa.ca

Sevag Gharibian
Department of Computer Science, Paderborn University, Germany
Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA
sevag.gharibian@upb.de

Hong-Sheng Zhou
Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA
hszhou@vcu.edu

Abstract

A central tenet of theoretical cryptography is the study of the minimal assumptions required to implement a given cryptographic primitive. One such primitive is the one-time memory (OTM), introduced by Goldwasser, Kalai, and Rothblum [CRYPTO 2008], which is a classical functionality modeled after a non-interactive 1-out-of-2 oblivious transfer, and which is complete for one-time classical and quantum programs. It is known that secure OTMs do not exist in the standard model in both the classical and quantum settings. Here, we propose a scheme for using quantum information, together with the assumption of stateless (i.e., reusable) hardware tokens, to build statistically secure OTMs. Via the semidefinite programming-based quantum games framework of Gutoski and Watrous [STOC 2007], we prove security for a malicious receiver, against a linear number of adaptive queries to the token, in the quantum universal composability framework, but leave open the question of security against a polynomial amount of queries. Compared to alternative schemes derived from the literature on quantum money, our scheme is technologically simple since it is of the "prepare-and-measure" type. We also show our scheme is "tight" according to two scenarios.

2012 ACM Subject Classification Theory of computation → Cryptographic protocols

Keywords and phrases quantum cryptography, one-time memories, semi-definite programming

Digital Object Identifier 10.4230/LIPIcs.TQC.2020.6

Related Version A full version of the paper is available at `https://arxiv.org/abs/1810.05226`.

Funding *Anne Broadbent*: U.S. Air Force Office of Scientific Research under award number FA9550-17-1-0083, Canada's NSERC, Ontario ERA, and the University of Ottawa's Research Chairs program.
Sevag Gharibian: NSF grants CCF-1526189 and CCF-1617710.

Acknowledgements We thank referees for pointing out the impossibility result against quantum queries applies only if we model the token as a *reversible* process, as well as for finding an error in a prior version of this work. We thank Kai-Min Chung and Jamie Sikora for related discussions.

1 Introduction

Theoretical cryptography centers around building cryptographic primitives secure against adversarial attacks. In order to allow a broader set of such primitives to be implemented, one often considers restricting the power of the adversary. For example, one can limit the *computing* power of adversaries to be polynomial bounded [68, 7], restrict the *storage* of adversaries to be bounded or noisy [49, 11, 22], or make *trusted setups* available to honest players [39, 6, 14, 16, 36, 55, 42, 46, 47, 48, 41, 40], to name a few. One well-known trusted

15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020).
Editor: Steven T. Flammia; Article No. 6; pp. 6:1–6:25
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

setup is *tamper-proof hardware* [38, 30], which is assumed to provide a specific input-output functionality, and which can only be accessed in a "black box" fashion. The hardware can maintain a state (i.e., is *stateful*) and possibly carry out complex functionality, but presumably may be difficult or expensive to implement or manufacture. This leads to an interesting research direction: Building cryptography primitives using the *simplest* (and hence easiest and cheapest to manufacture) hardware.

In this respect, two distinct simplified notions of hardware have captured considerable interest. The first is the notion of a *one-time memory (OTM)* [30], which is arguably the simplest possible notion of *stateful* hardware. An OTM, modeled after a non-interactive 1-out-of-2 oblivious transfer, behaves as follows: first, a player (called the *sender*) embeds two values s_0 and s_1 into the OTM, and then gives the OTM to another player (called the *receiver*). The receiver can now read his choice of precisely one of s_0 or s_1; after this "use" of the OTM, however, the unread bit is lost forever. Interestingly, OTMs are complete for implementing *one-time* use programs (OTPs): given access to OTMs, one can implement statistically secure OTPs for any efficiently computable program in the universal composability (UC) framework [32]. (OTPs, in turn, have applications in software protection and one-time proofs [30].) In the quantum UC model, OTMs enable *quantum* one-time programs [9]. (This situation is analogous to the case of *oblivious transfer* being complete for two-party secure function evaluation [39, 36].) Unfortunately, OTMs are inherently *stateful*, and thus represent a very strong cryptographic assumption – any physical implementation of such a device must somehow maintain internal knowledge between activations, i.e., it must completely "self-destruct" after a single use.

This brings us to a second important simplified notion of hardware known as a *stateless* token [17], which keeps no record of previous interactions. On the positive side, such hardware is presumably easier to implement. On the negative side, an adversary can run an experiment with stateless hardware as many times as desired, and each time the hardware is essentially "reset". (Despite this, stateless hardware has been useful in achieving *computationally secure* multi-party computation [17, 32, 19], and *statistically secure* commitments [23].) It thus seems impossible for stateless tokens to be helpful in implementing any sort of "self-destruct" mechanism. Indeed, classically stateful tokens are trivially more powerful than stateless ones, as observed in, e.g., [32]. This raises the question:

> *Can* quantum *information, together with a classical stateless token, be used to simulate "self destruction" of a hardware token?*

In particular, a natural question along these lines is whether quantum information can help implement an OTM. Unfortunately, it is known that quantum information *alone* cannot implement an OTM (or, more generally, any one-time program) [9]; see also Section 4 below. We thus ask the question: What are the minimal cryptographic assumptions required in a quantum world to implement an OTM?

1.1 Contributions and summary of techniques

We propose what is, to our knowledge, the first prepare-and-measure quantum protocol that constructs OTMs from stateless hardware tokens. For this protocol, we are able to rigorously prove information theoretic security against an adversary making a *linear* (in n, the security parameter) number of adaptive queries to the token. While we conjecture that security holds also for *polynomially* many queries, note that already in this setting of linearly many adaptive queries, our protocol achieves something impossible classically (i.e., classically, obtaining security against a linear number of queries is impossible). We also show stand-alone security against a malicious sender.

Historical Note. We proposed the concept that quantum information could provide a "stateless to stateful" transformation in a preliminary version of this work [8]; however, that work claimed security against a *polynomial* number of token queries, obtained via a reduction from the interactive to the non-interactive setting. We thank an anonymous referee for catching a subtle, but important bug which ruled out the proof approach of [8]. The current paper employs a different proof approach, which models interaction with the token as a "quantum game" via semidefinite programming. Since our original paper was posted, recent work [20] has shown an alternate quantum "stateful to stateless" transformation via quantum money constructions [3]. Specifically, in [20], security against a polynomial number of queries is achieved, albeit with respect to a new definition of "OTMs relative to an oracle" (while the security results of the present paper are with respect to the well-established simulation-based definition of [32, 38]). Furthermore, [20] directly applies known quantum money constructions, which require difficult-to-prepare highly entangled states. Our focus here, in contrast, is to take a "first-principles" approach and build a technologically simple-to-implement scheme which requires no entanglement, but rather the preparation of just one of four single qubit states, $|0\rangle, |1\rangle, |+\rangle, |-\rangle$. Indeed, the two works are arguably complementary in that the former focuses primarily on *applications* of "stateful" single-use tokens, while our focus is on the most technologically simple way to *implement* such "stateful" tokens.

Construction. Our construction is inspired by Wiesner's *conjugate coding* [65]: the quantum portion of the protocols consists in n quantum states chosen uniformly at random from $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ (note this encoding is independent of the classical bits of the OTM functionality). We then couple this n-qubit quantum state, $|\psi\rangle$ (the *quantum key*) with a *classical* stateless hardware token, which takes as inputs a choice bit b, together with an n-bit string y. If $b = 0$, the hardware token verifies that the bits of y that correspond to *rectilinear* ($|0\rangle$ or $|1\rangle$, i.e., Z basis) encoded qubits of $|\psi\rangle$ are consistent with the measurement of $|\psi\rangle$ in the computational basis, in which case the bit s_0 is returned. If $b = 1$, the hardware token verifies that the bits of y that correspond to *diagonal* ($|+\rangle$ or $|-\rangle$, i.e., X basis) encoded qubits of $|\psi\rangle$ are consistent with the measurement of $|\psi\rangle$ in the diagonal basis, in which case the bit s_1 is returned.[1] The honest use of the OTM is thus intuitive: for choice bit $b = 0$, the user measures each qubit of the quantum key in the rectilinear basis to obtain an n-bit string y, and inputs (b, y) into the hardware token. If $b = 1$, the same process is applied, but with measurements in the diagonal basis.

Assumption. Crucially, we assume the hardware token accepts *classical* input only (alternatively and equivalently, the token immediately measures its quantum input in the standard basis), i.e., it cannot be queried in superposition. Although this may seem a strong assumption, in Section 4 we show that any token which can be queried in superposition in a reversible way, cannot be used to construct a secure OTM (with respect to our setting in which the adversary is allowed to apply arbitrary quantum operations). Similar classical-input hardware has previously been considered in, e.g., [60, 9].

Security and intuition. Stand-alone security against a malicious sender is relatively simple to establish, since the protocol consists in a single message from the sender to the receiver, and stand-alone security only requires simulation of the *local* view of the adversary.

[1] We note that a simple modification using a classical one-time pad could be used to make *both* the quantum state and hardware token independent of s_0 and s_1: the token would output one of two uniformly random bits r_0 and r_1, which could each be used to decrypt a single bit, s_0 or s_1.

The intuition underlying security against a malicious receiver is clear: in order for a receiver to extract a bit s_b as encoded in the OTM, she must perform a complete measurement of the qubits of $|\psi\rangle$ in order to obtain a classical key for s_b (since, otherwise, she would likely fail the test as imposed by the hardware token). But such a measurement would invalidate the receiver's chance of extracting the bit s_{1-b}! This is exactly the "self-destruct"-like property we require in order to implement an OTM. This intuitive notion of security was present in Wiesner's proposal for quantum money [65], and is often given a physical explanation in terms of the no-cloning theorem [67] or Heisenberg uncertainty relation [35].

Formally, we work in the statistical (i.e., information-theoretic) setting of the quantum *Universal Composability* (UC) framework [59], which allows us to make strong security statements that address the *composability* of our protocol within others. As a proof technique, we describe a simulator, such that for any "quantum environment" wishing to interact with the OTM, the environment statistically cannot tell whether it is interacting with the *ideal* OTM functionality or the *real* OTM instance provided by our scheme. The security of this simulator requires a statement of the following form: Given access to a (randomly chosen) "quantum key" $|\psi_k\rangle$ and corresponding stateless token V_k, it is highly unlikely for an adversary to successfully extract keys for *both* the secret bits s_0 and s_1 held by V_k. We are able to show this statement for any adversary which makes a linear number of queries, by which we mean an adversary making m queries succeeds with probability at most $O(2^{2m-0.228n})$ (for n the number of quantum key bits in $|\psi_k\rangle$). In other words, if the adversary makes at most $m = cn$ queries with $c < 0.114$, then its probability of cheating successfully is exponentially small in n. We conjecture, however, that a similar statement holds for any $m \in \text{poly}(n)$, i.e., that the protocol is secure against polynomially many queries.

To show security against linearly many queries, we exploit the semidefinite programming-based quantum games framework of Gutoski and Watrous (GW) [33] to model interaction with the token. Intuitively, GW is useful for our setting, since it is general enough to model multiple rounds of adaptive queries to the token, even when the receiver holds quantum "side information" in the form of $|\psi\rangle$. We describe this technique in Sections 2.1 and 3.4, and provide formal details in the full version. Summarizing, we show the following.

▶ **Main Theorem (informal).** *There exists a protocol* Π*, which together with a classical stateless token and the ability to randomly prepare single qubits in one of four pure states, implements the OTM functionality with statistical security in the UC framework against a corrupted receiver making a linear number of adaptive queries.*

As stated above, we conjecture that our protocol is actually secure against polynomially many adaptive queries. However, we are unable to show this claim using our present proof techniques, and hence leave this question open. Related to this, we make the following comments: (1) As far as we are aware, the Main Theorem above is the only known formal proof of any type of security for conjugate coding in the interactive setting with $\Omega(1)$ queries. Moreover, as stated earlier, classically security against $\Omega(1)$ queries is trivially impossible. (2) Our proof introduces the GW semidefinite programming framework from quantum interactive proofs to the study of conjugate coding-based schemes. This framework allows handling multiple challenges in a unified fashion: arbitrary quantum operations by the user, classical queries to the token, and the highly non-trivial assumption of quantum side information for the user (the "quantum key" state sent to the user.)

Towards security against polynomially many queries. Regarding the prospects of proving security against polynomially many adaptive queries, we generally believe it requires a significant new insight into how to design a "good" feasible solution to the primal semidefinite program (SDP) obtained via GW. However, in addition to our proof for linear security, in the full version we give evidence potentially supporting our conjecture for polynomial security. Namely, we first simplify the SDPs obtained from GW, and derive the corresponding dual SDPs. These derivations apply for any instantiation of the GW framework, i.e. they are not specific to our setting, and hence may prove useful elsewhere. We then give a feasible solution Y to the dual SDP. While Y is simple to state, it is somewhat involved to analyze. A heuristic analysis suggests Y's dual objective function value has precisely the behavior needed to show security, i.e. the value scales as $m/\sqrt{2^n}$, for m queries and n key bits. If Y were to be the *optimal* solution to the dual SDP, this would strongly suggest the optimal cheating probability is essentially $m/\sqrt{2^n}$. However, we explicitly show Y is not optimal, and so $m/\sqrt{2^n}$ is only a *lower bound* on the optimal cheating probability. Nevertheless, we conjecture that while Y is not optimal, it is *approximately* optimal; this would imply the desired polynomial security claim. Unfortunately, the only techniques we are aware of to show approximate optimality require a better primal SDP solution, which appears challenging.

Further related work. Our work contributes to the growing list of functionalities achievable with quantum information, yet unachievable classically. This includes: unconditionally secure key expansion [4], physically uncloneable money [65, 51, 53], a reduction from oblivious transfer to bit commitment [5, 21] and to other primitives such as "cut-and choose" functionality [27], and revocable time-release quantum encryption [61]. Importantly, these protocols all make use of the technique of conjugate coding [65], which is also an important technique used in protocols for OT in the bounded quantum storage and noisy quantum storage models [22, 63] (see [10] for a survey).

Various proof techniques have been developed in the context of conjugate coding, including entropic uncertainty relations [64]. In the context of QKD, another technique is the use of de Finetti reductions [58] (which exploit the symmetry of the scheme in order to simplify the analysis). Recently, semidefinite programming (SDP) approaches have been applied to analyze security of conjugate coding [51] for quantum money, in the setting of one round of interaction with a "stateful" bank. SDPs are also the technical tool we adopt for our proof (Section 3.4), though here we require the more advanced quantum games SDP framework of Gutoski and Watrous [33] to deal with multiple adaptive interactions with stateless tokens. Reference [53] has also made use of Gavinsky's [28] quantum retrieval games framework.

Somewhat similar to [53], Aaronson and Christiano [1] have studied quantum money schemes in which one interacts with a verifier. They introduce an "inner product adversary method" to lower bound the number of queries required to break their scheme.

We remark that [53] and [51] have studied schemes based on conjugate coding similar to ours, but in the context of quantum money. In contrast to our setting, the schemes of [53] and [51] (for example) involve dynamically chosen random challenges from a verifier to the holder of a "quantum banknote", whereas in our work here the "challenges" are fixed (i.e., measure all qubits in the Z or X basis to obtain secret bit s_0 or s_1, respectively), and the verifier is replaced by a stateless token. Thus, [51], for example, may be viewed as using a "stateful" verifier, whereas our focus here is on a "stateless" verifier (i.e., a token).

Also, prior work has achieved oblivious transfer using quantum information, together with some assumption (e.g., bit commitment [5], bounded quantum storage [22]). These protocols typically use an interaction phase similar to the "commit-and-open" protocol of [5]; because we are working in the non-interactive setting, these techniques appear to be inapplicable.

Finally, Liu [43, 44, 45] has given stand-alone secure OTMs using quantum information in the *isolated-qubit model.* Liu's approach is nice in that it avoids the use of trusted setups. In return, however, Liu must use the isolated-qubit model, which restricts the adversary to perform only single-qubit operations (no entangling gates are permitted); this restriction is, in some sense, necessary if one wants to avoid trusted setups, as a secure OTM in the plain quantum model cannot exist (see Section 4). In contrast, in the current work we allow unbounded and unrestricted quantum adversaries, but as a result require a trusted setup. In addition, we remark the security notion of OTMs of [43, 44, 45] is weaker than the simulation-based notion studied in this paper, and it remains an interesting open question whether the type of OTM in [43, 44, 45] is secure under composition (in the current work, the UC framework gives us security under composition for free).

Significance. Our results show a strong separation between the classical and quantum settings, since classically, stateless tokens cannot be used to securely implement OTMs. To the best of our knowledge, our work is the first to combine conjugate coding with *stateless* hardware tokens. Moreover, while our protocol shares similarities with prior work in the setting of quantum money, building OTMs appears to be a new focus here [2].

Our protocol has a simple implementation, fitting into the single-qubit prepare-and-measure paradigm, which is widely used as the "benchmark" for a "physically feasible" quantum protocol (in this model, one needs only the ability to prepares single-qubit states $|0\rangle, |1\rangle, |+\rangle, |-\rangle$, and to perform single-qubit projective measurements. In particular, no entangled states are required, and in principle no quantum memory is required, since qubits can be measured one-by-one as they arrive). In addition, from a theoretical cryptographic perspective, our protocol is attractive in that its implementation requires an assumption of a stateless hardware token, which is easier and cheaper to mass produce than a stateful token.

In terms of security guarantees, we allow *arbitrary* operations on behalf of a malicious quantum receiver in our protocol (i.e., all operations allowed by quantum mechanics), with the adversary restricted in that the stateless token is assumed only usable as a black box. The security we obtain is statistical, with the only computational assumption being on the number of *queries* made to the token (recall we show security for a linear number of queries, and conjecture security for polynomially many queries). Finally, our security analysis is in the quantum UC framework against a corrupted receiver; this means our protocol can be easily composed with many others; for example, combining our results with [9]'s protocol immediately yields UC-secure quantum OTPs against a dishonest receiver.

Finally, our scheme is "tight" with respect to two impossibility results (Section 4), both of which assume the adversary has black-box access to both the token and its inverse operation[3]. First, the assumption that the token be queried only in the computational basis cannot be relaxed: If the token can be queried in superposition, then an adversary can easily break an OTM scheme. Second, our scheme has the property that corresponding to each secret bit s_i held by the token, there are exponentially many valid keys one can input to the token to extract s_i. We show that for any "measure-and-access" OTM (i.e., an OTM in which one measures a given quantum key and uses the classical measurement result to access a token to extract data, of which our protocol is an example), a polynomial number of keys implies the ability to break the scheme with inverse polynomial probability (more generally, Δ keys allows probability at least $1/\Delta^2$ of breaking the scheme).

[2] We remark, however, that a reminiscent concept of single usage of quantum "tickets" in the context of quantum money is very briefly mentioned in Appendix S.4.1 of [53].

[3] This is common in the oracle model of quantum computation, where a function $f : \{0,1\}^n \mapsto \{0,1\}$ is implemented via the (self-inverse) unitary mapping $U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$.

Open Questions. While our work shows the fundamental advantage that quantum information yields in a stateful to stateless reduction, it does leave a number of open questions:

1. **Security against polynomially many queries.** Can our security proof be strengthened to show information theoretic security against a polynomial number of queries to the token? We conjecture this to be the case, but finding a formal proof has been elusive.
2. **Composable security against a malicious sender.** While we show composable security against a malicious receiver, our protocol can achieve standalone security against a malicious sender. Could an adaptation of our protocol ensure composable security against a malicious sender as well?[4]
3. **Non-reversible token.** Our impossibility result for quantum one-time memories with *quantum* queries (Section 4) assumes the adversary has access to reversible tokens; can a similar impossibility result be shown for non-reversible tokens? In Section 4, we briefly discuss why it may be difficult to extend the techniques of our impossibility results straightforwardly when the adversary does *not* have access to the inverse of the token.
4. **Imperfect devices.** While our prepare-and-measure scheme is technologically simple, it is still unrealizable with current technology, due to the requirement of perfect quantum measurements. We leave open the question of tolerance to a small amount of noise.

Organization. Section 2 covers preliminaries, including ideal functionalities for an OTM and stateless token, background on quantum channels, semidefinite programming, and the Gutoski-Watrous (GW) framework for quantum games. Section 3 gives our construction for an OTM based on a stateless hardware token; the proof ideas for security are also provided. Section 4 discusses "tightness" of our construction by showing two impossibility results for "relaxations" of our scheme. In the Appendix, we discuss classical UC and quantum UC (Appendix A); Appendix B establishes notation required in the definition of stand-alone security against a malicious sender. Due to space constraints, our formal security proof against a linear number of queries to the token (used to finish the security proof in Section 3) is deferred to the full version, along with simplifications of the GW SDP, derivation of its dual, and a dual feasible solution which we conjecture to be approximately optimal.

2 Preliminaries

Notation. Two binary distributions $\mathbf{X}$ and $\mathbf{Y}$ are *indistinguishable*, denoted $\mathbf{X} \approx \mathbf{Y}$, if $|\Pr(X_n = 1) - \Pr(Y_n = 1)| \leq \mathrm{negl}(n)$. We define single-qubit $|0\rangle_+ = |0\rangle$ and $|1\rangle_+ = |1\rangle$, so that $\{|0\rangle_+, |1\rangle_+\}$ form the *rectilinear basis*. We define $|0\rangle_\times = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle_\times = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, so that $\{|0\rangle_\times, |1\rangle_\times\}$ form the *diagonal basis*. For strings $x = x_1, x_2, \ldots x_n \in \{0,1\}^n$ and $\theta = \theta_1, \theta_2, \ldots, \theta_n \in \{+, \times\}^n$, define $|x\rangle_\theta = \bigotimes_{i=1}^n |x_i\rangle_{\theta_i}$. For $\mathcal{X}$ a finite dimensional complex Hilbert space, $\mathcal{L}(\mathcal{X})$, $\mathrm{Herm}(\mathcal{X})$, $\mathrm{Pos}(\mathcal{X})$, and $\mathcal{D}(\mathcal{X})$ denote the sets of linear, Hermitian, positive semidefinite, and density operators acting on $\mathcal{X}$, respectively. Notation $A \succeq B$ means $A - B$ is positive semidefinite.

Quantum universal composition (UC) framework. We study simulation-based security in this paper. In particular, we prove security of our construction against a malicious receiver in the quantum universal composition (UC) framework [59]. See Appendix A for a description of classical UC [14] and quantum UC [59]. In the next two paragraphs, we introduce the ideal functionalities of one-time memory and stateless hardware token.

[4] We note that this would require a different protocol, since in our current construction, a cheating sender could program the token to abort based on the user's input.

One-time memory (OTM). The one-time memory (OTM) functionality $\mathcal{F}_{\mathtt{OTM}}$ involves two parties, the sender and the receiver, and consists of two phases, "Create" and "Execute". Please see Functionality 1 below for details; for the sake of simplicity, we have omitted the session/party identifiers as they should be implicitly clear from the context. We sometimes refer to this functionality $\mathcal{F}_{\mathtt{OTM}}$ as an *OTM token*.

Functionality 1 Ideal functionality $\mathcal{F}_{\mathtt{OTM}}$.

1. **Create:** Upon input (s_0, s_1) from the sender, with $s_0, s_1 \in \{0, 1\}$, send create to the receiver and store (s_0, s_1).
2. **Execute:** Upon input $b \in \{0, 1\}$ from the receiver, send s_b to receiver. Delete any trace of this instance.

Stateless hardware. The original work of Katz [38] introduces the ideal functionality $\mathcal{F}_{\mathtt{wrap}}$ to model stateful tokens in the UC-framework. In the ideal model, a party that wants to create a token, sends the Turing machine to $\mathcal{F}_{\mathtt{wrap}}$. $\mathcal{F}_{\mathtt{wrap}}$ will then run the machine (keeping the state), when the designated party will ask for it. The same functionality can be adapted to model stateless tokens. It is sufficient that the functionality does not keep the state between two executions. A simplified version of the $\mathcal{F}_{\mathtt{wrap}}$ functionality as shown in [17] (that is very similar to the $\mathcal{F}_{\mathtt{wrap}}$ of [38]) is described below. Note that, again for the sake of simplicity, we have omitted the session/party identifiers as they should be implicitly clear from the context. Although the environment and adversary are unbounded, we specify that stateless hardware can be queried only a polynomial number of times. This is necessary; otherwise the hardware token model is vacuous (with unbounded queries, the entire input-output behavior of stateless hardware can be extracted).

Functionality 2 Ideal functionality $\mathcal{F}_{\mathtt{wrap}}$.

The functionality is parameterized by a polynomial $p(\cdot)$, and implicit security parameter n.

1. **Create:** Upon input (create, M) from the sender, where M is a Turing machine, send create to the receiver and store M.
2. **Execute:** Upon input (run, msg) from the receiver, execute $M(msg)$ for at most $p(n)$ steps, and let *out* be the response. Let *out* $:= \perp$ if M does not halt in $p(n)$ steps. Send *out* to the receiver.

Quantum channels. A linear map $\Phi : \mathcal{L}(\mathcal{X}) \mapsto \mathcal{L}(\mathcal{Y})$ is a *quantum channel* if Φ is trace-preserving and completely positive (TPCP). Such maps take density operators to density operators. A useful representation of linear maps (or "superoperators") $\Phi : \mathcal{L}(\mathcal{X}) \mapsto \mathcal{L}(\mathcal{Y})$ is the Choi-Jamiołkowski representation, $J(\Phi) \in \mathcal{L}(\mathcal{Y} \otimes \mathcal{X})$. The latter is defined (with respect to some choice of orthonormal basis $\{|i\rangle\}$ for $\mathcal{X}$) as $J(\Phi) = \sum_{i,j} \Phi(|i\rangle\langle j|) \otimes |i\rangle\langle j|$. The following properties of $J(\Phi)$ hold [18, 37]: (1) Φ is completely positive if and only if $J(\Phi) \succeq 0$, and (2) Φ is trace-preserving if and only if $\mathrm{Tr}_{\mathcal{Y}}(J(\Phi)) = I_{\mathcal{X}}$. In a nutshell, the Gutoski-Watrous (GW) framework generalizes this definition to *interacting* strategies [33].

Semidefinite programs. We review semidefinite programs (SDPs) from the perspective of quantum information, as done e.g., in the notes of Watrous [62] or [51]. Given any 3-tuple (A, B, Φ) for operators $A \in \mathrm{Herm}(\mathcal{X})$ and $B \in \mathrm{Herm}(\mathcal{Y})$, and Hermiticity-preseving linear map $\Phi : \mathcal{L}(\mathcal{X}) \mapsto \mathcal{L}(\mathcal{Y})$, one can state a *primal* and *dual* semidefinite program:

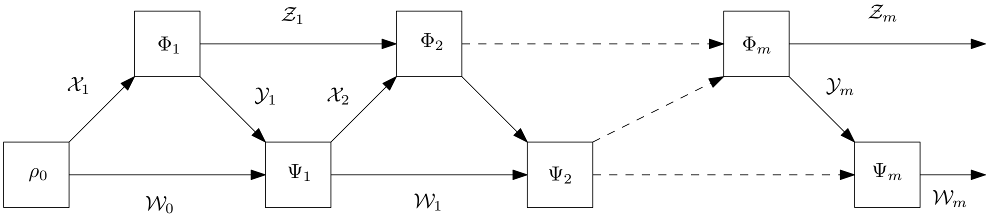

Figure 1 A general interaction between two quantum parties.

$$\begin{array}{llcll} & \underline{\text{Primal problem (P)}} & & & \underline{\text{Dual problem (D)}} \\ \sup & \operatorname{Tr}(AX) & & \inf & \operatorname{Tr}(BY) \\ \text{s.t.} & \Phi(X) = B, & & \text{s.t.} & \Phi^*(Y) \succeq A \\ & X \in \operatorname{Pos}(\mathcal{X}), & & & Y \in \operatorname{Herm}(\mathcal{Y}), \end{array}$$

where Φ^* denotes the *adjoint* of Φ, which is the unique map satisfying $\operatorname{Tr}(A^\dagger \Phi(B)) = \operatorname{Tr}((\Phi^*(A))^\dagger B)$ for all $A \in \mathcal{L}(\mathcal{Y})$ and $B \in \mathcal{L}(\mathcal{X})$. Not all SDPs have feasible solutions (i.e. a solution satisfying all constraints); in this case, optimal values are $-\infty$ for P and ∞ for D.

2.1 The Gutoski-Watrous framework for quantum games

We now recall the Gutoski-Watrous (GW) framework for quantum games [33], which can be used to model quantum interactions between spatially separated parties. The setup most relevant to our protocol here is depicted in Figure 1. Here, we imagine one party, A, prepares an initial state $\rho_0 \in \mathcal{D}(\mathcal{X}_1 \otimes \mathcal{W}_0)$. Register $\mathcal{X}_1$ is then sent to the second party ($\mathcal{W}_0$ is kept as private memory), B, who applies some quantum channel $\Phi_i : \mathcal{L}(\mathcal{X}_1) \mapsto \mathcal{L}(\mathcal{Y}_1 \otimes \mathcal{Z}_1)$. B keeps register $\mathcal{Z}_1$ as private memory, and sends $\mathcal{Y}_1$ back to A, who applies channel $\Psi_1 : \mathcal{L}(\mathcal{W}_0 \otimes \mathcal{Y}_1) \mapsto \mathcal{L}(\mathcal{X}_2 \otimes \mathcal{W}_1)$, and sends $\mathcal{X}_2$ to B. The protocol continues for m messages back and forth, until the final operation $\Psi_m : \mathcal{L}(\mathcal{W}_m \otimes \mathcal{Y}_m) \mapsto \mathbb{C}$, in which A performs a two-outcome measurement (specifically, a POVM $\Lambda = \{\Lambda_0, \Lambda_1\}$, meaning $\Lambda_0, \Lambda_1 \succeq 0$, $\Lambda_0 + \Lambda_1 = I$) in order to decide whether to reject (Λ_0) or accept (Λ_1). As done in [33], without loss of generality (by the Stinespring dilation theorem) all channels are given by linear isometries A_k, i.e. $\Phi_k(X) = A_k X A_k^\dagger$. Reference [33] refers to $(\Phi_1, \ldots, \Phi_m)$ as a *strategy* and $(\rho_0, \Psi_1, \ldots, \Psi_m)$ as a *co-strategy*. In our setting, the former is "non-measuring", meaning it makes no final measurement after Φ_m is applied, whereas the latter is "measuring", since we will apply a final measurement on space $\mathcal{W}_m$ (not depicted in Figure 1).

Intuitively, since our protocol (Section 3.1) begins with the token sending the user a quantum key $|x\rangle_\theta$, we will model the token as a *measuring co-strategy*, and the user as a *strategy*. The advantage to doing so is that the GW framework allows one to (recursively) characterize any such strategy (resp., co-strategy) via a set of linear (in)equalities and positive semi-definite constraints. (In this sense, the GW framework generalizes the Choi-Jamiołkowski representation for channels to a "Choi-Jamiołkowski" representation for strategies/co-strategies.) To state these constraints, we first write down the Choi-Jamiołkowski (CJ) representation of a strategy (resp., measuring co-strategy) from [33].

CJ representation of (non-measuring) strategy. The CJ representation of a strategy $(A_1, \ldots, A_m)$ is given by matrix [33]

$$\operatorname{Tr}_{\mathcal{Z}_m}(\operatorname{vec}(A)\operatorname{vec}(A)^\dagger), \tag{1}$$

where $A \in \mathcal{L}(\mathcal{X}_1 \otimes \cdots \otimes \mathcal{X}_m, \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_m \otimes \mathcal{Z}_m)$ is the product of the isometries A_i,

$$A := (I_{\mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_{m-1}} \otimes A_m) \cdots (A_1 \otimes I_{\mathcal{X}_2 \otimes \cdots \otimes \mathcal{X}_m}), \tag{2}$$

and the vec : $\mathcal{L}(\mathcal{S}, \mathcal{T}) \mapsto \mathcal{T} \otimes \mathcal{S}$ mapping is the linear extension of the map $|i\rangle\langle j| \mapsto |i\rangle|j\rangle$ defined on all standard basis states $|i\rangle, |j\rangle$.

CJ representation of (measuring) co-strategy. Let $P := \{\Lambda_0, \Lambda_1\}$ denote a POVM with reject and accept measurement operators Λ_0 and Λ_1, respectively. A measuring strategy which ends with a measurement via POVM Λ replaces, for $\Lambda_a \in \Lambda$, Equation (1) with [33]

$$Q_a := \mathrm{Tr}_{\mathcal{Z}_m}((\Lambda_a \otimes I_{\mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_m}) \operatorname{vec}(A) \operatorname{vec}(A)^\dagger) = \mathrm{Tr}_{\mathcal{Z}_m}(\operatorname{vec}(B_a) \operatorname{vec}(B_a)^\dagger), \tag{3}$$

for $B_a := (\sqrt{\Lambda_a} \otimes I_{\mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_m})A$. To convert this to a *co*-strategy, one takes the transpose of the operators defined above (with respect to the standard basis).

Optimization characterization over strategies and co-strategies. With CJ representations for strategies and co-strategies in hand, one can formulate [33] the optimal probability with which a strategy can force a corresponding co-strategy to output a desired result as follows. Fix any Q_a from a measuring co-strategy $\{Q_0, Q_1\}$, as in Equation (3). Then, Corollary 7 and Theorem 9 of [33] show that the maximum probability with which a (non-measuring) strategy can force the co-strategy to output result a is given by

$$\begin{aligned}
\text{min:}\quad & p && && (4)\\
\text{subject to:}\quad & Q_a \preceq pR_m && && (5)\\
& R_k = P_k \otimes I_{\mathcal{Y}_k} && \text{for } 1 \le k \le m && (6)\\
& \mathrm{Tr}_{\mathcal{X}_k}(P_k) = R_{k-1} && \text{for } 1 \le k \le m && (7)\\
& R_0 = 1 && && (8)\\
& R_k \in \mathrm{Pos}(\mathcal{Y}_{1,\ldots,k} \otimes \mathcal{X}_{1,\ldots,k}) && \text{for } 1 \le k \le m && (9)\\
& P_k \in \mathrm{Pos}(\mathcal{Y}_{1,\ldots,k-1} \otimes \mathcal{X}_{1,\ldots,k}) && \text{for } 1 \le k \le m && (10)\\
& p \in [0,1] && && (11)
\end{aligned}$$

Intuition. The minimum p denotes the optimal "success" probability, meaning the optimal probability of forcing the co-strategy to output a (Theorem 9 of [33]). The variables above, in addition to p, are $\{R_i\}$ and $\{P_i\}$, where the optimization is happening over all m-round co-strategies R_m satisfying Equation (5). How do we enforce that R_m encodes such an m-round co-strategy? This is given by the (recursive) Equations (6)-(10). Specifically, Corollary 7 of [33] states that R_m is a valid m-round co-strategy if and only if all of the following hold: (1) $R_m \succeq 0$, (2) $R_m = P_m \otimes I_{\mathcal{Y}_m}$ for $P_m \succeq 0$ and $\mathcal{Y}_m$ the last incoming message register to the co-strategy, (3) $\mathrm{Tr}_{\mathcal{X}_m}(P_m)$ is a valid $m-1$ round co-strategy (this is the recursive part of the definition). An intuitive sense as to why conditions (2) and (3) should hold is as follows: For any m-round co-strategy R_m, let R_{m-1} denote R_m restricted to the first $m-1$ rounds. Then, to operationally obtain R_{m-1} from R_m, the co-strategy first ignores the last incoming message in register $\mathcal{Y}_m$. This is formalized via a partial trace over $\mathcal{Y}_m$, which (once pushed through the CJ formalism[5]) translates into the $\otimes I_{\mathcal{Y}_k}$ term

[5] Recall that the CJ representation of the trace map is the identity matrix (up to scaling).

in Equation (6). Since the co-strategy is now ignoring the last *incoming* message $\mathcal{Y}_m$, any measurement it makes after $m-1$ rounds is independent of the last *outgoing* message $\mathcal{X}_m$. Thus, we can trace out $\mathcal{X}_m$ as well, obtaining a co-strategy R_{m-1} on just the first $m-1$ rounds; this is captured by Equation (7).

3 Feasibility of Quantum OTMs using Stateless Hardware

In this section, we present a *quantum* construction for one-time memories by using stateless hardware (Section 3.1). We also state our main theorem (Theorem 1). In Section 3.3, we describe the Simulator and prove Theorem 1 using the technical results of the full version. The intuition and techniques behind the proofs in the full version are sketched in Section 3.4.

3.1 Construction

We now present the OTM protocol Π in the $\mathcal{F}_{\texttt{wrap}}$ hybrid model, between a sender P_{s} and a receiver P_{r}. Here the security parameter is n.

- Upon receiving input (s_0, s_1) from the environment where $s_0, s_1 \in \{0,1\}$, the sender:
  - The sender chooses uniformly random $x \in_R \{0,1\}^n$ and $\theta \in_R \{+,\times\}^n$, and prepares $|x\rangle_\theta$. Based on (s_0, s_1, x, θ), the sender prepares program M as in **Program 1**.

Program 1 Program for hardware token.

Hardcoded values: $s_0, s_1 \in \{0,1\}$, $x \in \{0,1\}^n$, and $\theta \in \{+,\times\}^n$

Inputs: $y \in \{0,1\}^n$ and $b \in \{0,1\}$, where y is a claimed measured value for the quantum register, and b the evaluator's choice bit

1. If $b = 0$, check that the $\theta = +$ positions return the correct bits in y according to x. If Accept, output s_0. Otherwise output $\perp$.
2. If $b = 1$, check that the $\theta = \times$ positions return the correct bits in y according to x. If Accept, output s_1. Otherwise output $\perp$.

  - The sender sends $|x\rangle_\theta$ to the receiver.
  - The sender sends (create, M) to functionality $\mathcal{F}_{\texttt{wrap}}$, and the functionality sends create to notify the receiver.
- The receiver P_{r} operates as follows:
  Upon input b from the environment, and $|x\rangle_\theta$ from the receiver, and create notification from $\mathcal{F}_{\texttt{wrap}}$,
  - If $b = 0$, measure $|x\rangle_\theta$ in computational basis to get y. Input $(\mathsf{run}, (y, b))$ into $\mathcal{F}_{\texttt{wrap}}$.
  - If $b = 1$, apply $\mathrm{H}^{\otimes n}$ to $|x\rangle_\theta$, then measure in computational basis to get y. Input $(\mathsf{run}, (y, b))$ into $\mathcal{F}_{\texttt{wrap}}$.

  Return the output of $\mathcal{F}_{\texttt{wrap}}$ to the environment.
  It is easy to see that the output of $\mathcal{F}_{\texttt{wrap}}$ is s_b for both $b = 0$ and $b = 1$.

Note again that the hardware token, as defined in **Program 1**, accepts only classical input (i.e., it cannot be queried in superposition). As mentioned earlier, relaxing this assumption yields impossibility of a secure OTM implementation (assuming the receiver also has access to the token's inverse operation), as shown in Section 4.

3.2 Stand-Alone Security Against a Malicious Sender

We note that in protocol Π of Section 3.1, once the sender prepares and sends the token, she is no longer involved (and in particular, the sender does not receive any further communication from the receiver). We call such a protocol a *one-way* protocol. Because of this simple structure, and because the ideal functionality $\mathcal{F}_{\texttt{wrap}}$ also does not return any message to the sender, we can easily establish stand-alone security against a malicious sender (Appendix B).

3.3 UC-Security against a corrupt receiver

Our main theorem, which establishes security against a corrupt receiver is now stated.

▶ **Theorem 1.** *Construction* Π *above quantum-UC-realizes* $\mathcal{F}_{\mathtt{OTM}}$ *in the* $\mathcal{F}_{\mathtt{wrap}}$ *hybrid model with statistical security against an actively-corrupted receiver making at most* cn *number of adaptive queries to the token, for any fixed constant* $c < 0.114$.

To prove Theorem 1, we now construct and analyze an appropriate simulator.

3.3.1 The simulator

In order to prove Theorem 1, for an adversary $\mathcal{A}$ that corrupts the receiver, we build a simulator $\mathcal{S}$ (having access to the OTM functionality $\mathcal{F}_{\mathtt{OTM}}$), such that for any unbounded environment $\mathcal{Z}$, the executions in the real model and that in simulation are statistically indistinguishable. Our simulator $\mathcal{S}$ is given below:

- The simulator emulates an internal copy of the adversary $\mathcal{A}$ who corrupts the receiver. The simulator emulates the communication between $\mathcal{A}$ and the external environment $\mathcal{Z}$ by forwarding the communication messages between $\mathcal{A}$ and $\mathcal{Z}$.
- The simulator $\mathcal{S}$ needs to emulate the whole view for the adversary $\mathcal{A}$. First, $\mathcal{S}$ picks dummy inputs $\tilde{s}_0 = 0$ and $\tilde{s}_1 = 0$, and randomly chooses $x \in \{0,1\}^n$, and $\theta \in \{+,\times\}^n$, and generates program $\tilde{M}$. Then the simulator plays the role of the sender to send $|x\rangle_\theta$ to the adversary $\mathcal{A}$ (who controls the corrupted receiver). The simulator also emulates $\mathcal{F}_{\mathtt{wrap}}$ to notify $\mathcal{A}$ by sending create to indicate the hardware is ready for queries.
- For each query (run, (b,y)) to $\mathcal{F}_{\mathtt{wrap}}$ from the adversary $\mathcal{A}$, the simulator evaluates program $\tilde{M}$ (created based on $\tilde{s}_0, \tilde{s}_1, x, \theta$) as in the construction, and then acts as follows:
  1. If this is a rejecting input, output $\perp$.
  2. If this is the first accepting input, call the external $\mathcal{F}_{\mathtt{OTM}}$ with input b, and learn the output s_b from $\mathcal{F}_{\mathtt{OTM}}$. Output s_b.
  3. If this is a subsequent accepting input, output s_b (as above).

3.3.2 Analysis

We now show that the simulation and the real model execution are statistically indistinguishable. There are two cases in an execution of the simulation which we must consider:

- *Case 1: In all its queries to* $\mathcal{F}_{\mathtt{wrap}}$*, the accepting inputs of* $\mathcal{A}$ *have the same choice bit* b. In this case, the simulation is perfectly indistinguishable.
- *Case 2: In its queries to* $\mathcal{F}_{\mathtt{wrap}}$*,* $\mathcal{A}$ *produces accepting inputs for both* $b = 0$ *and* $b = 1$. In this case, it is possible that the simulation fails (the environment can distinguish the real model from the ideal model), since the simulator is only able to retrieve a single bit from the external OTM functionality $\mathcal{F}_{\mathtt{OTM}}$ (either corresponding to $b = 0$ or $b = 1$).

Thus, whereas in Case 1 the simulator behaves perfectly, in Case 2 it is in trouble. Fortunately, in Theorem 2 we show that the probability that Case 2 occurs is exponentially small in n, the number of qubits comprising $|x\rangle_\theta$, provided the number of queries to the token is at most cn for any $c < 0.114$. Specifically, we show that for an arbitrary m-query strategy (i.e., any quantum strategy allowed by quantum mechanics, whether efficiently implementable or not, which queries the token at most m times), the probability of Case 2 occurring is at most $O(2^{2m-0.228n})$. This concludes the proof.

3.4 Security analysis for the token: Intuition

Our simulation proof showing statistical security of our Quantum OTM construction of Section 3.1 relies crucially on Theorem 2, stated below. For this, we now introduce notation in line with the formal analysis of the full version.

With respect to the construction of Section 3.1, let us replace each two-tuple $(x, \theta) \in \{0,1\}^n \times \{+, \times\}^n$ by a single string $z \in \{0,1\}^{2n}$, which we denote the *secret key*. Bits $2i$ and $2i+1$ of z specify the basis and value of conjugate coding qubit i for $i \in \{1, \dots, n\}$ (i.e., $z_{2i} = \theta_i$ and $z_{2i+1} = x_i$). Also, rename the "quantum key" (or conjugate coding key) $|\psi_z\rangle := |x\rangle_\theta \in (\mathbb{C}^2)^{\otimes n}$. Thus, the protocol begins by having the sender pick a *secret key* $z \in \{0,1\}^{2n}$ uniformly at random, and preparing a joint state

$$|\psi\rangle = \frac{1}{2^n} \sum_{z \in |0,1\rangle^{2n}} |\psi_z\rangle_R |z\rangle_T. \tag{12}$$

The first register, R, is sent to the receiver, while the second register, T, is kept by the token. (Thus, the token knows the secret key z, and hence also which $|\psi_z\rangle$ the receiver possesses.) The mixed state describing the receiver's state of knowledge at this point is given by

$$\rho_R := \frac{1}{2^{2n}} \sum_{z \in \{0,1\}^{2n}} |\psi_z\rangle\langle\psi_z|.$$

▶ **Theorem 2.** *Given a single copy of ρ_R, and the ability to make m (adaptive) queries to the hardware token, the probability that an unbounded quantum adversary can force the token to output both bits s_0 and s_1 scales as $O(2^{2m-0.228n})$.*

Thus, the probability of an unbounded adversary (i.e., which applies arbitrary trace-preserving completely positive (TPCP) maps, which are not necessarily efficiently implementable) to successfully cheat using $m = cn$ for $c < 0.114$ queries is exponentially small in the quantum key size, n. The proof of Theorem 2 is in the full version; here, we give intuition.

Proof intuition. The challenge in analyzing security of the protocol is the fact that the receiver (a.k.a. the user) is not only given adaptive query access to the token, but also a copy of the quantum "resource state" ρ_R, which it may arbitrarily tamper with (in any manner allowed by quantum mechanics) while making queries. Luckily, the GW framework [33] (Section 2.1)) is general enough to model such "queries with quantum side information". The framework outputs an SDP, Γ (Equation (13)), the optimal value of which will encode the optimal cheating probability for a cheating user of our protocol. Giving a feasible solution for Γ will hence suffice to upper bound this cheating probability, yielding Theorem 2.

Coherently modeling quantum queries to the token. To model the interaction between the token and user, we first recall that all queries to the token must be classical by assumption. To model this process *coherently* in the GW framework, we hence imagine (solely for the purposes of the security analysis) that the token behaves as follows:

1. It first sends state ρ_R to the user.
2. When it receives as ith query a quantum state ρ_i from the user, it sends response string r_i to the user, and "copies" ρ_i via transversal CNOT gates to a private memory register $\mathcal{W}_i$, along with r_i. It does not access ρ_i again throughout the protocol, and only accesses r_i again in Step 3. For clarity, the token runs a classical circuit, and conditions each response r_i solely on the current incoming message, ρ_i.
3. After all communication, the token "measures" its responses $(r_1, \ldots, r_m)$ in the Z-basis to decide whether to accept (user successfully cheated) or reject (user failed to cheat).

The "copying" phase of Step 2 accomplishes two tasks: First, since the token will never read the "copies" of ρ_i again, the principle of deferred measurement [52] implies the transversal CNOT gates effectively simulate measuring ρ_i in the standard basis. In other words, without loss of generality the user is reduced to feeding a classical string $\widetilde{y}$ to the token. Second, we would like the entire security analysis to be done in a unified fashion in a single framework, the GW framework. To this end, we want the token itself to "decide" at the end of the protocol whether the user has successfully cheated (i.e. extracted both secret bits). Storing all responses r_i in Step 2 allows us to simulate such a final measurement in Step 3. We reiterate that, crucially, once the token "copies" ρ_i and r_i to W_i, it (1) never accesses (i.e. reads or writes to) ρ_i again and (2) only accesses r_i again in the final standard basis measurement of Step 3. Together, these ensure all responses r_i are independent, as required..

Formalization in GW framework. To place the discussion thus far into the formal GW framework, we return to Figure 1. The bottom "row" of Figure 1 will depict the token's actions, and the top row the user's actions. As outlined above, the protocol begins by imagining the token sends initial state $\rho_0 = \rho_R$ to the user via register $\mathcal{X}_1$. The user then applies an arbitrary sequence of TPCP maps Φ_i to its private memory (modeled by register $\mathcal{Z}_i$ in round i), each time sending a query $\widetilde{y}_i$ (which is, as discussed above a classical string without loss of generality) to the token via register $\mathcal{Y}_i$. Given any such query $\widetilde{y}_i$ in round i, the token applies its own TPCP map Ψ_i to determine how to respond to the query. In our protocol, the Ψ_i correspond to coherently applying a classical circuit, i.e. a sequence of unitary gates mapping the standard basis to itself. Specifically, their action is fully determined by Program 1, and in principle all Ψ_i are identical since the token is stateless (i.e., the action of the token in round i is unaffected by previous rounds $\{1, \ldots, i-1\}$). (We use the term "in principle", as recall from above that in the security analysis we model each Ψ_i as classically copying $(\widetilde{y}_i, r_i)$ to a distinct private register W_i.) Finally, after receiving the mth query $\widetilde{y}_m$ in register $\mathcal{Y}_m$, we imagine the token makes a measurement (not depicted in Fig. 1) based on the query responses $(r_1, \ldots, r_m)$ it returned; if the user managed to extract both s_0 and s_1 via queries, then the token "accepts"; otherwise it "rejects". (Again, we are using the fact that in our security analysis, the token keeps a history of all its responses r_i, solely for the sake of this final measurement.)

With this high-level setup, the output of the GW framework is a semidefinite program, Γ:

$$\begin{aligned}
\text{min:} \quad & p && && (13)\\
\text{subject to:} \quad & Q_1 \preceq R_{m+1} && && (14)\\
& R_k = P_k \otimes I_{\mathcal{Y}_k} && \text{for } 1 \leq k \leq m+1 && (15)\\
& \mathrm{Tr}_{\mathcal{X}_k}(P_k) = R_{k-1} && \text{for } 1 \leq k \leq m+1 && (16)\\
& R_0 = p && && (17)\\
& R_k \in \mathrm{Pos}(\mathcal{Y}_{1,\ldots,k} \otimes \mathcal{X}_{1,\ldots,k}) && \text{for } 1 \leq k \leq m+1 && (18)\\
& P_k \in \mathrm{Pos}(\mathcal{Y}_{1,\ldots,k-1} \otimes \mathcal{X}_{1,\ldots,k}) && \text{for } 1 \leq k \leq m+1 && (19)
\end{aligned}$$

Above, Q_1 encodes the actions of the token, i.e. the co-strategy in the bottom row of Figure 1. The variable p denotes the "cheating probability" (i.e., the probability with which both s_0 and s_1 are extracted), subject to linear constraints (Equations (15)-(19)) which enforce that operator R_{m+1} encodes a valid co-strategy (see Section 2.1). Theorem 9 of [33] now says that the minimum p above encodes precisely the optimal cheating probability for a user which is constrained only by the laws of quantum mechanics. Since Γ is a minimization problem, to upper bound the the cheating probability it hence suffices to give a feasible solution $(p, R_1, \dots, R_{m+1}, P_1, \dots, P_{m+1})$ for Γ, which will be our approach.

Intuition for Q_1 and an upper bound on p. It remains to give intuition as to how one derives Q_1 in Γ, and how an upper bound on the optimal p is obtained. Without loss of generality, one may assume that each of the token's TPCP maps Ψ_i are given by *isometries* $A_i : \mathcal{Y}_i \otimes \mathcal{W}_{i-1} \mapsto \mathcal{X}_{i+1} \otimes \mathcal{W}_i$, meaning $A_i^\dagger A_i = I_{\mathcal{Y}_i \otimes \mathcal{W}_{i-1}}$ (due to the Stinespring dilation theorem). (We omit the first isometry which prepares state ρ_0 in our discussion here for simplicity.) Let us denote their sequential application by a single operator $A := A_m \cdots A_1$. Then, the Choi-Jamiołkowski representation of A is given by [33] (Section 2.1) $\operatorname{Tr}_{\mathcal{Z}_m}(\operatorname{vec}(A) \operatorname{vec}(A)^\dagger)$, where we trace out the token's private memory register $\mathcal{Z}_m$. However, since in our security analysis, we imagine the token also makes a final measurement via some POVM $\Lambda = \{\Lambda_0, \Lambda_1\}$, whereupon obtaining outcome Λ_1 the token "accepts", and upon outcome Λ_0 the token rejects, we require a slightly more complicated setup. Letting $B_1 := \Lambda_1 A$, we define Q_1 as [33] $Q_1 = \operatorname{Tr}_{\mathcal{Z}_m}(\operatorname{vec}(B_1) \operatorname{vec}(B_1)^\dagger)$.

The full derivation of Q_1 is deferred to the full version; here, we state Q_1 with intuition:

$$\begin{aligned} Q_1 \quad &= \quad \frac{1}{4^n} \sum_{s \in T} |t_m s_{t_m}\rangle\langle t_m s_{t_m}|_{\mathcal{X}_{m+1}} \otimes \cdots \otimes |t_1 s_{t_1}\rangle\langle t_1 s_{t_1}|_{\mathcal{X}_2} \otimes \\ &\quad \left(\sum_{(\widetilde{y}, z) \in Y_t} |\widetilde{y}_m\rangle\langle \widetilde{y}_m|_{\mathcal{Y}_m} \otimes \cdots \otimes |\widetilde{y}_1\rangle\langle \widetilde{y}_1|_{\mathcal{Y}_1} \otimes |\psi_z\rangle\langle \psi_z|_{\mathcal{X}_1} \right). \end{aligned}$$

Intuitively, each string $t_i s_{t_i} \in \{0,1\}^3$ encodes the response r_i of the token given the ith query from the user; hence, the corresponding projectors in Q_1 act on spaces $\mathcal{X}_2$ through $\mathcal{X}_{m+1}$. Each string $\widetilde{y}_i \in \{0,1\}^{n+1}$ denotes the ith query sent from the user to the token, where each $\widetilde{y}_i = b_i \circ y_i$ in the notation of Program 1, i.e. $b_i \in \{0,1\}$ is the choice bit for each query. Each such message is passed via register $\mathcal{Y}_i$. The states $|\psi_z\rangle$ and strings z are defined as in the beginning of Section 3.4; recall $z \in \{0,1\}^{2n}$ and $|\psi_z\rangle \in (\mathbb{C}^2)^{\otimes n}$ denote the secret key and corresponding quantum key, respectively. Finally, the relation Y_t encodes the constraint that for all $i \in \{1, \dots, m\}$, the tuple $(\widetilde{y}_i, z)$ (i.e. the ith message to the token, $\widetilde{y}_i$, and secret key z) is consistent with the response returned by the token, t_i.

Upper bounding p. To now upper bound p, we give a feasible solution R_{m+1} satisfying the constraints of Γ. Note that giving even a solution which attains $p = 1$ for all n and m is *non*-trivial – such a solution is given in the full version. Here, we give a solution which attains $p \in O(2^{2m-0.228n})$, as claimed in Theorem 2. Namely, we set

$$R_{m+1} \quad = \quad \frac{1}{|T|} \sum_{t \in T} |t_m s_{t_m}\rangle\langle t_m s_{t_m}|_{\mathcal{X}_{m+1}} \otimes \cdots \otimes |t_1 s_{t_1}\rangle\langle t_1 s_{t_1}|_{\mathcal{X}_2} \otimes I_{Y_1 \otimes \cdots \otimes Y_m} \otimes \frac{I}{2^n}_{\mathcal{X}_1}.$$

This satisfies constraint (15) of Γ due to the identity term $I_{Y_1 \otimes \cdots \otimes Y_m}$. The renormalization factor $(|T|\, 2^n)^{-1}$ above ensures that tracing out all $\mathcal{X}_i$ registers yields $R_0 = 1$ in constraint (17) of Γ. We are thus reduced to choosing the minimum p satisfying constraint (14).

Now, observe we have chosen R_{m+1} to align with the block-diagonal structure of Q_1 on registers $\mathcal{X}_2, \ldots, \mathcal{X}_m$. Since registers $\mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_m$ and $\mathcal{X}_1$ of R_{m+1} are proportional to the identity matrix, it thus suffices to characterize the largest eigenvalue of Q_1, $\lambda_{\max}(Q_1)$. This is done in the full version, which shows $\lambda_{\max}(Q_1) = \frac{2}{4^n}\left(1 + \frac{1}{\sqrt{2}}\right)^n$. Combining this bound on $\lambda_{\max}(Q_1)$ with the parameters of R_{m+1} above now yields the desired claim that $p \in O(2^{2m-0.228n})$. For $m < 0.114n$ queries, this implies that the probability that a user of the token successfully cheats and thus that the simulation fails is exponentially small in the key size, n. Simplifications of the GW SDP, the derivation of its dual SDP, and a conjectured approximately optimal dual feasible solution are given in the full version.

4 Impossibility Results

We now discuss "tightness" of our protocol with respect to impossibility results. To begin, it is easy to argue that OTMs cannot exist in the plain model (i.e., without additional assumptions) in both the classical and quantum settings: in the classical setting, impossibility holds, since software can always be copied. Quantumly, this follows by a rewinding argument [9]. Here, we give two no-go results for the quantum setting which support the idea that our scheme is "tight" in terms of the minimality of the assumptions it uses. Both results assume the token is reversible, meaning the receiver can run both the token and its inverse operation. Note that if the receiver is *not* given access to the token's inverse operation, it is unlikely for our no-go techniques to go through. This is because, in the most general case where the token is an arbitrary unitary U, which the receiver may apply as a black box, simulating $U^{-1} = U^\dagger$ appears difficult [26, 57]; see the full version for a discussion.

Result 1: Tokens which can be queried in superposition. In our construction, we require that all queries to the token be classical strings, i.e., no querying in superposition is allowed. It is easy to argue via a standard rewinding argument that relaxing this requirement yields impossibility of a secure OTM, as long as access to the token's adjoint (inverse) operation is given, as we now show. Specifically, let M be a quantum OTM implemented using a hardware token. Since the token access is assumed to be reversible, we may model it as an oracle O_f realizing a function $f : \{0,1\}^n \mapsto \{0,1\}^m$ in the standard way, i.e., for all $y \in \{0,1\}^n$ and $b \in \{0,1\}^m$, $O_f|y\rangle|b\rangle = |y\rangle|b \oplus f(y)\rangle$. Now, suppose our OTM stores two secret bits s_0 and s_1, and provides the receiver with an initial state $|\psi\rangle \in A \otimes B \otimes C$, where A, B, and C are the algorithm's workspace, *query* (i.e., input to O_f), and *answer* (i.e., O_f's answers) registers, respectively. By definition, an honest receiver must be able to access precisely one of s_0 or s_1 with certainty, given $|\psi\rangle$. Thus, for any $i \in \{0,1\}$, there exists a quantum query algorithm $A_i = U_m O_f \cdots O_f U_2 O_f U_1$ for unitaries $U_i \in \mathcal{U}(A \otimes B \otimes C)$ such that $A_i|\psi\rangle = |\psi'\rangle_{AB}|s_i\rangle_C$. For any choice of i, however, this implies a malicious receiver can now classically copy s_i to an external register, and then "rewind" by applying $A_i^\dagger$ to $|\psi'\rangle_{AB}|s_i\rangle_C$ to recover $|\psi\rangle$. Applying $A_{i'}$ for $i' \neq i$ to $|\psi\rangle$ now yields the second bit i' with certainty as well. We conclude that a quantum OTM which allows superposition queries to a reversible stateless token is insecure.

▶ Remark 3. Above, the OTM outputs s_i with certainty. A similar argument holds if s_i is output with probability at least $1 - \epsilon$ for small $\epsilon > 0$ via the Gentle Measurement Lemma [66].

Result 2: Tokens with a bounded number of keys. We observed superposition queries to the token prevent an OTM from being secure. One can also ask how simple a hardware token with classical queries can be, while still allowing a secure OTM. Below, we consider such a strengthening in which the token is forced to have a bounded number of keys.

To formalize this, we define the notion of a “measure-and-access (MA)” OTM, i.e., an OTM in which given an initial state $|\psi\rangle$, an honest receiver applies a prescribed measurement to $|\psi\rangle$, and feeds the resulting classical string (i.e., key) y into the token O_f to obtain s_i. Our construction is an example of a MA memory in which each bit s_i has an *exponential* number of valid keys y such that $f(y) = s_i$. Can the construction can be strengthened such that each s_i has a bounded number (e.g., a polynomial number) of keys? We now show that such a strengthening would preclude security, assuming the token is reversible.

▶ **Lemma 4.** *Let M be an MA memory with oracle O_f, such that O_f cannot be queried in superposition. If a secret bit s_i has at most Δ keys y_i such that $f(y_i) = s_i$, then given a single copy of $|\psi\rangle$, one can extract both s_0 and s_1 from M with probability at least $1/\Delta^2$.*

Thus, if a secret bit b_i has at most polynomially many keys, then any measure-and-access OTM can be broken with at least inverse polynomial probability. The proof is in the full version. In this sense,in the setting of measure-and-access memories, our construction is tight – in order to bound the adversary’s success probability of obtaining both secret bits by an inverse exponential, we require each secret bit to have exponentially many valid keys.

References

1 Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In *Proc. 44th Symposium on Theory of Computing (STOC) 2012*, pages 41–60, 2012. Full version available as arXiv:1203.4740. doi:10.1145/2213977.2213983.

2 Donald Beaver. Secure multiparty protocols and zero-knowledge proof systems tolerating a faulty minority. *Journal of Cryptology*, 4(2):75–122, 1991.

3 Shalev Ben-David and Or Sattath. Quantum tokens for digital signatures. arXiv:1609.09047, 2018.

4 Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *International Conference on Computers, Systems and Signal Processing*, pages 175–179, 1984.

5 Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical quantum oblivious transfer. In Joan Feigenbaum, editor, *CRYPTO’91*, volume 576 of *LNCS*, pages 351–366. Springer, August 1992.

6 Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications. In *20th ACM STOC*, pages 103–112. ACM Press, May 1988.

7 Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo random bits. In *23rd FOCS*, pages 112–117. IEEE Computer Society Press, November 1982.

8 Anne Broadbent, Sevag Gharibian, and Hong-Sheng Zhou. Quantum one-time memories from stateless hardware. arXiv:1511.01363, November 2015.

9 Anne Broadbent, Gus Gutoski, and Douglas Stebila. Quantum one-time programs. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 344–360. Springer, August 2013. doi:10.1007/978-3-642-40084-1_20.

10 Anne Broadbent and Christian Schaffner. Quantum cryptography beyond quantum key distribution. *Designs, Codes and Cryptography*, 78(1):351–382, 2016.

11 Christian Cachin and Ueli Maurer. Unconditional security against memory-bounded adversaries. In *Advances in Cryptology - CRYPTO 1997*, LNCS, pages 292–306. Springer, 1997. doi:10.1007/BFb0052243.

12 Ran Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000.

13 Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067, 2000. URL: http://eprint.iacr.org/2000/067.

14 Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145. IEEE Computer Society Press, October 2001.

15 Ran Canetti, Yevgeniy Dodis, Rafael Pass, and Shabsi Walfish. Universally composable security with global setup. In Salil P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 61–85. Springer, February 2007.

16 Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *34th ACM STOC*, pages 494–503. ACM Press, May 2002.

17 Nishanth Chandran, Vipul Goyal, and Amit Sahai. New constructions for UC secure computation using tamper-proof hardware. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 545–562. Springer, April 2008.

18 Man-Duen Choi. Completely positive linear maps on complex matrices. *Linear Alg. Appl.*, 10:285, 1975.

19 Seung Geol Choi, Jonathan Katz, Dominique Schröder, Arkady Yerukhimovich, and Hong-Sheng Zhou. (efficient) universally composable oblivious transfer using a minimal number of stateless tokens. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 638–662. Springer, February 2014. doi:10.1007/978-3-642-54242-8_27.

20 Kai-Min Chung, Marios Georgiou, Ching-Yi Lai, and Vassilis Zikas. Cryptography with disposable backdoors. eprint:2018/352, 2018.

21 Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner. Improving the security of quantum protocols via commit-and-open. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 408–427. Springer, August 2009.

22 Ivan Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. Cryptography in the bounded quantum-storage model. In *Symposium on Foundations of Computer Science - FOCS 2005*, pages 449–458. IEEE, 2005. doi:10.1109/SFCS.2005.30.

23 Ivan Damgård and Alessandra Scafuro. Unconditionally secure and universally composable commitments from physical assumptions. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part II*, volume 8270 of *LNCS*, pages 100–119. Springer, December 2013. doi:10.1007/978-3-642-42045-0_6.

24 Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In *Advances in Cryptology – Proc. CRYPTO 2010*, LNCS, pages 685–706. Springer, 2010.

25 Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Actively secure two-party evaluation of any quantum operation. In *Advances in Cryptology – Proc. CRYPTO 2012*, volume 7417 of *LNCS*, pages 794–811. Springer, 2012. doi:10.1007/978-3-642-32009-5_46.

26 Bill Fefferman and Shelby Kimmel. Quantum vs. Classical Proofs and Subset Verification. In Igor Potapov, Paul Spirakis, and James Worrell, editors, *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, volume 117 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:23, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.MFCS.2018.22.

27 Serge Fehr, Jonathan Katz, Fang Song, Hong-Sheng Zhou, and Vassilis Zikas. Feasibility and completeness of cryptographic tasks in the quantum world. In Amit Sahai, editor, *TCC 2013*, volume 7785 of *LNCS*, pages 281–296. Springer, March 2013. doi:10.1007/978-3-642-36594-2_16.

28 Dmitry Gavinsky. Quantum money with classical verification. In *Computational Complexity (CCC), 2012 IEEE 27th Annual Conference on*, pages 42–52, June 2012. doi:10.1109/CCC.2012.10.

29 Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, *19th ACM STOC*, pages 218–229. ACM Press, May 1987.

30 Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. One-time programs. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 39–56. Springer, August 2008.

31 Shafi Goldwasser and Leonid A. Levin. Fair computation of general functions in presence of immoral majority. In Alfred J. Menezes and Scott A. Vanstone, editors, *CRYPTO'90*, volume 537 of *LNCS*, pages 77–93. Springer, August 1991.

32 Vipul Goyal, Yuval Ishai, Amit Sahai, Ramarathnam Venkatesan, and Akshay Wadia. Founding cryptography on tamper-proof hardware tokens. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 308–326. Springer, February 2010.

33 Gus Gutoski and John Watrous. Toward a general theory of quantum games. In *Proceedings of the 39th ACM Symposium on Theory of Computing (STOC 2007)*, pages 565–574, 2007. arXiv:quant-ph/0611234v2.

34 Sean Hallgren, Adam Smith, and Fang Song. Classical cryptographic protocols in a quantum world. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 411–428. Springer, August 2011.

35 Werner Heisenberg. Schwankungserscheinungen und quantenmechanik. *Zeitschrift fuer Physik*, 40(7):501–506, July 1927. doi:10.1007/BF01440827.

36 Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding cryptography on oblivious transfer - efficiently. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 572–591. Springer, August 2008.

37 Andrzej Jamiołkowski. Linear transformations which preserve trace and positive semidefiniteness of operators. *Rep. Math. Phys.*, 3:275, 1972.

38 Jonathan Katz. Universally composable multi-party computation using tamper-proof hardware. In Moni Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 115–128. Springer, May 2007.

39 Joe Kilian. Founding cryptography on oblivious transfer. In *20th ACM STOC*, pages 20–31. ACM Press, May 1988.

40 Daniel Kraschewski, Hemanta K. Maji, Manoj Prabhakaran, and Amit Sahai. A full characterization of completeness for two-party randomized function evaluation. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 659–676. Springer, May 2014. doi:10.1007/978-3-642-55220-5_36.

41 Daniel Kraschewski and Jörn Müller-Quade. Completeness theorems with constructive proofs for finite deterministic 2-party functions. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 364–381. Springer, March 2011.

42 Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. A unified framework for concurrent security: universal composability from stand-alone non-malleability. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 179–188. ACM Press, 2009.

43 Yi-Kai Liu. Building one-time memories from isolated qubits. In Moni Naor, editor, *ITCS 2014*, pages 269–286. ACM, January 2014.

44 Yi-Kai Liu. Single-shot security for one-time memories in the isolated qubits model. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 19–36. Springer, August 2014. doi:10.1007/978-3-662-44381-1_2.

45 Yi-Kai Liu. Privacy amplification in the isolated qubits model. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 785–814. Springer, April 2015. doi:10.1007/978-3-662-46803-6_26.

46 Hemanta K. Maji, Manoj Prabhakaran, and Mike Rosulek. Complexity of multi-party computation problems: The case of 2-party symmetric secure function evaluation. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 256–273. Springer, March 2009.

47 Hemanta K. Maji, Manoj Prabhakaran, and Mike Rosulek. A zero-one law for cryptographic complexity with respect to computational UC security. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 595–612. Springer, August 2010.

48 Ueli Maurer and Renato Renner. Abstract cryptography. In Bernard Chazelle, editor, *ICS 2011*, pages 1–21. Tsinghua University Press, January 2011.

49 Ueli M. Maurer. Protocols for secret key agreement by public discussion based on common information. In *Advances in Cryptology - CRYPTO 1992*, volume 740 of *LNCS*, pages 461–470. Springer, 1992. doi:10.1007/3-540-48071-4_32.

50 Silvio Micali and Phillip Rogaway. Secure computation (abstract). In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 392–404. Springer, August 1992.

51 Abel Molina, Thomas Vidick, and John Watrous. Optimal counterfeiting attacks and generalizations for Wiesner's quantum money. In Kazuo Iwama, Yasuhito Kawano, and Mio Murao, editors, *Theory of Quantum Computation, Communication, and Cryptography*, volume 7582 of *Lecture Notes in Computer Science*, pages 45–64, 2013.

52 M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

53 Fernando Pastawski, Norman Y Yao, Liang Jiang, Mikhail D Lukin, and J Ignacio Cirac. Unforgeable noise-tolerant quantum tokens. *Proceedings of the National Academy of Sciences*, 109(40):16079–16082, 2012.

54 Birgit Pfitzmann and Michael Waidner. A model for asynchronous reactive systems and its application to secure message transmission. In *Proc. 22nd IEEE Symposium on Security & Privacy (S&P) 2001*, pages 184–200. IEEE, 2001. Full version available at http://eprint.iacr.org/2000/066. doi:10.1109/SECPRI.2001.924298.

55 Manoj Prabhakaran and Mike Rosulek. Cryptographic complexity of multi-party computation problems: Classifications and separations. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 262–279. Springer, August 2008.

56 Manoj Prabhakaran and Amit Sahai. New notions of security: Achieving universal composability without trusted setup. In László Babai, editor, *36th ACM STOC*, pages 242–251. ACM Press, June 2004.

57 Marco Túlio Quintino, Qingxiuxiong Dong, Atsushi Shimbo, Akihito Soeda, and Mio Murao. Reversing unknown quantum transformations: Universal quantum circuit for inverting general unitary operations. *Phys. Rev. Lett.*, 123:210502, November 2019. doi:10.1103/PhysRevLett.123.210502.

58 Renato Renner. *Security of Quantum Key Distribution.* PhD thesis, ETH Zürich (Switzerland), September 2008. doi:10.1142/S0219749908003256.

59 Dominique Unruh. Universally composable quantum multi-party computation. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 486–505. Springer, May 2010.

60 Dominique Unruh. Everlasting multi-party computation. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 380–397. Springer, August 2013. doi:10.1007/978-3-642-40084-1_22.

61 Dominique Unruh. Revocable quantum timed-release encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 129–146. Springer, May 2014. doi:10.1007/978-3-642-55220-5_8.

62 John Watrous. Lecture 7: Semidefinite programming, 2011. Latest version available at: https://cs.uwaterloo.ca/~watrous/CS766/LectureNotes/07.pdf.

63 Stephanie Wehner, Christian Schaffner, and Barbara M. Terhal. Cryptography from noisy storage. *Physical Review Letters*, 100(22):220502, June 2008. doi:10.1103/PhysRevLett.100.220502.

64 Stephanie Wehner and Andreas Winter. Entropic uncertainty relations—a survey. *New J. Phys.*, 12(2):025009, February 2010. doi:10.1088/1367-2630/12/2/025009.

65 Stephen Wiesner. Conjugate coding. *ACM SIGACT News*, 15(1):78–88, 1983. Original article written circa 1970. doi:10.1145/1008908.1008920.

66 Andreas Winter. Coding theorem and strong converse for quantum channels. *IEEE Transactions on Information Theory*, 45:2481–2485, 1999.

67 William K. Wootters and Wojciech H. Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982 . doi:10.1038/299802a0.

68 Andrew Chi-Chih Yao. Theory and applications of trapdoor functions. In *23rd FOCS*, pages 80–91. IEEE Computer Society Press, November 1982.

A Universal Composition (UC) Framework

We consider simulation-based security. The Universal Composability (UC) framework was proposed by Canetti [14, 13], culminating a long sequence of simulation-based security definitions (*c.f.* [29, 31, 50, 2, 12]); please see also [54, 56, 15, 42, 48] for alternative/extended frameworks. Recently Unruh [59] extend the UC framework to the quantum setting. Next, we provide a high-level description of the original classical UC model by Canetti [14, 13], and then the quantum UC model by Unruh [59].

A.1 Classical UC Model ([14, 13])

Machines. The basic entities involved in the UC model are players $P_1, \ldots, P_k$ where k is polynomial of security parameter n, an adversary $\mathcal{A}$, and an environment $\mathcal{Z}$. Each entity is modeled as a interactive Turing machine (ITM), where $\mathcal{Z}$ could have an additional non-uniform string as advice. Each P_i has identity i assigned to it, while $\mathcal{A}$ and $\mathcal{Z}$ have special identities $id_{\mathcal{A}} := \texttt{adv}$ and $id_{\mathcal{Z}} := \texttt{env}$.

Protocol Execution. A protocol specifies the programs for each P_i, which we denote as $\pi = (\pi_1, \ldots, \pi_k)$. The execution of a protocol is coordinated by the environment $\mathcal{Z}$. It starts by preparing inputs to all players, who then run their respective programs on the inputs and exchange messages of the form $(id_{\texttt{sender}}, id_{\texttt{receiver}}, \texttt{msg})$. $\mathcal{A}$ can corrupt an arbitrary set of players and control them later on. In particular, $\mathcal{A}$ can instruct a corrupted player sending messages to another player and also read messages that are sent to the corrupted players. During the course of execution, the environment $\mathcal{Z}$ also interacts with $\mathcal{A}$ in an arbitrary way. In the end, $\mathcal{Z}$ receives outputs from all the other players and generates one bit output. We use $\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \pi]$ denote the distribution of the environment $\mathcal{Z}$'s (single-bit) output when executing protocol π with $\mathcal{A}$ and the P_i's.

Ideal Functionality and Dummy Protocol. Ideal functionality $\mathcal{F}$ is a trusted party, modeled by an ITM again, that perfectly implements the desired multi-party computational task. We consider an "dummy protocol", denoted $P^{\mathcal{F}}$, where each party has direct communication with $\mathcal{F}$, who accomplishes the desired task according to the messages received from the players. The execution of $P^{\mathcal{F}}$ with environment $\mathcal{Z}$ and an adversary, usually called the simulator $\mathcal{S}$, is defined analogous as above, in particular, $\mathcal{S}$ monitors the communication between corrupted parties and the ideal functionality $\mathcal{F}$. Similarly, we denote $\mathcal{Z}$'s output distribution as $\mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, P^{\mathcal{F}}]$.

▶ **Definition 5** (Classical UC-secure Emulation). *We say π (classically) UC-emulates π' if for any adversary $\mathcal{A}$, there exists a simulator $\mathcal{S}$ such that for all environments $\mathcal{Z}$,*

$$\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \pi] \approx \mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, \pi'] \tag{20}$$

We here consider that $\mathcal{A}$ and $\mathcal{Z}$ are computationally unbounded, and we call it statistical UC-security. We require the running time $\mathcal{S}$ is polynomial in that of $\mathcal{A}$. We call this property Polynomial Simulation.

Let $\mathcal{F}$ be a well-formed two party functionality. We say π (classically) UC-realizes $\mathcal{F}$ if for all adversary $\mathcal{A}$, there exists a simulator $\mathcal{S}$ such that for all environments $\mathcal{Z}$, $\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \pi] \approx \mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, P^{\mathcal{F}}]$. We also write $\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \pi] \approx \mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, \mathcal{F}]$ if the context is clear.

UC-secure protocols admit a general composition property, demonstrated in the following universal composition theorem.

▶ **Theorem 6** (UC Composition Theorem [13]). *Let π, π' and σ be n-party protocols. Assume that π UC-emulates π'. Then σ^{π} UC-emulates $\sigma^{\pi'}$.*

A.2 Quantum UC Model ([59])

Now, we give a high-level description of quantum UC model by Unruh [59].

Quantum Machine. In the quantum UC model, all players are modeled as quantum machines. A quantum machine is a sequence of quantum circuits $\{M^n\}_{n\in\mathbb{N}}$, for each security parameter n. M^n is a completely positive trace preserving operator on space $\mathcal{H}^{\texttt{state}} \otimes \mathcal{H}^{\texttt{class}} \otimes \mathcal{H}^{\texttt{quant}}$, where $\mathcal{H}^{\texttt{state}}$ represents the internal workspace of M^n and $\mathcal{H}^{\texttt{class}}$ and $\mathcal{H}^{\texttt{quant}}$ represent the spaces for communication, where for convenience we divide the messages into classical and quantum parts. We allow a non-uniform quantum advice[6] to the machine of the environment $\mathcal{Z}$, while all other machines are uniformly generated.

Protocol Execution. In contrast to the communication policy in classical UC model, we consider a network $\mathbf{N}$ which contains the space $\mathcal{H}_{\mathbf{N}} := \mathcal{H}^{\texttt{class}} \otimes \mathcal{H}^{\texttt{quant}} \otimes_i \mathcal{H}_i^{\texttt{state}}$. Namely, each machine maintains individual internal state space, but the communication space is shared among all . We assume $\mathcal{H}^{\texttt{class}}$ contains the message $(id_{\texttt{sender}}, id_{\texttt{receiver}}, \texttt{msg})$ which specifies the sender and receiver of the current message, and the receiver then processes the quantum state on $\mathcal{H}^{\texttt{quant}}$. Note that this communication model implicitly ensures authentication. In a protocol execution, $\mathcal{Z}$ is activated first, and at each round, one player applies the operation defined by its machine M^n on $\mathcal{H}^{\texttt{class}} \otimes \mathcal{H}^{\texttt{quant}} \otimes \mathcal{H}^{\texttt{state}}$. In the end $\mathcal{Z}$ generates a one-bit output. Denote $\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \Pi]$ the output distribution of $\mathcal{Z}$.

Ideal Functionality. All functionalities we consider in this work are classical, i.e., the inputs and outputs are classical, and its program can be implemented by an efficient classical Turing machine. Here in the quantum UC model, the ideal functionality $\mathcal{F}$ is still modeled as a quantum machine for consistency, but it only applies classical operations. Namely, it measures any input message in the computational basis to get a classical bit-string, and implements the operations specified by the classical computational task.

We consider an "dummy protocol", denoted $P^{\mathcal{F}}$, where each party has direct communication with $\mathcal{F}$, who accomplishes the desired task according to the messages received from the players. The execution of $P^{\mathcal{F}}$ with environment $\mathcal{Z}$ and an adversary, usually called the simulator $\mathcal{S}$, is defined analogous as above, in particular, $\mathcal{S}$ monitors the communication between corrupted parties and the ideal functionality $\mathcal{F}$. Similarly, we denote $\mathcal{Z}$'s output distribution as $\mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, P^{\mathcal{F}}]$. For simplicity, we also write it as $\mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, \mathcal{F}]$.

▶ **Definition 7** (Quantum UC-secure Emulation). *We say Π quantum-UC-emulates Π' if for any quantum adversary $\mathcal{A}$, there exists a (quantum) simulator $\mathcal{S}$ such that for all quantum environments $\mathcal{Z}$,*

$$\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \Pi] \approx \mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, \Pi'] \tag{21}$$

[6] Unruh's model only allows classical advice, but we tend to take the most general model. It is easy to justify that almost all results remain unchanged, including the composition theorem. See [34, Section 5] for more discussion.

We consider here that $\mathcal{A}$ and $\mathcal{Z}$ are computationally unbounded, we call it (quantum) statistical UC-security. We require the running time $\mathcal{S}$ is polynomial in that of $\mathcal{A}$. We call this property Polynomial Simulation.

Similarly, (quantum) computational UC-security can be defined. Let $\mathcal{F}$ be a well-formed two party functionality. We say Π **quantum-UC-realizes** $\mathcal{F}$ if for all quantum adversary $\mathcal{A}$, there exists a (quantum) simulator $\mathcal{S}$ such that for all quantum environments $\mathcal{Z}$, $\mathsf{EXEC}[\mathcal{Z}, \mathcal{A}, \Pi] \approx \mathsf{EXEC}[\mathcal{Z}, \mathcal{S}, P^{\mathcal{F}}]$.

Quantum UC-secure protocols also admit general composition:

▶ **Theorem 8** (Quantum UC Composition Theorem [59, Theorem 11])**.** *Let Π, Π' and Σ be quantum-polynomial-time protocols. Assume that Π quantum UC-emulates Π'. Then Σ^{Π} quantum UC-emulates $\Sigma^{\Pi'}$.*

▶ Remark 9. Out of the two protocol parties (the sender and the receiver), we consider security only in the case of the receiver being a corrupted party. Note that we are only interested in cases where the same party is corrupted with respect to all composed protocol. Furthermore, we only consider static corruption.

B Stand-Alone Security in the case of a Malicious Sender

In order to define stand-alone security against a malicious sender (Definition 11), in our context, we closely follow definitions given in prior work [24], which we now recall. (Note that, instead of considering the *approximate* case for security, we are able to use the *exact* one.)

▶ **Definition 10.** *An n-step quantum two-party protocol with oracle calls, denoted $\Pi^{\mathcal{O}} = (\mathscr{A}, \mathscr{B}, \mathcal{O}, n)$ consists of:*

1. *input space $\mathcal{A}_0$ and $\mathcal{B}_0$ for parties $\mathscr{A}$ and $\mathscr{B}$ respectively.*
2. *memory spaces $\mathcal{A}_1, \dots \mathcal{A}_n$ and $\mathcal{B}_1, \dots \mathcal{B}_n$ for $\mathscr{A}$ and $\mathscr{B}$, respectively.*
3. *An n-tuple of quantum operations $(\mathscr{A}_1, \dots \mathscr{A}_n)$ for $\mathscr{A}$, $\mathscr{A}_i : \mathcal{L}(\mathcal{A}_{i-1}) \mapsto \mathcal{L}(\mathcal{A}_i), (1 \leq i \leq n)$.*
4. *An n-tuple of quantum operations $(\mathscr{B}_1, \dots \mathscr{B}_n)$ for $\mathscr{B}$, $\mathscr{B}_i : \mathcal{L}(\mathcal{B}_{i-1}) \mapsto \mathcal{L}(\mathcal{B}_i), (1 \leq i \leq n)$.*
5. *Memory spaces $\mathcal{A}_1, \dots, \mathcal{A}_n$ and $\mathcal{B}_1, \dots, \mathcal{B}_n$ can be written as $\mathcal{A}_i = {\mathcal{A}_i}^{\mathcal{O}} \otimes {\mathcal{A}_i}'$ and $\mathcal{B}_i = {\mathcal{B}_i}^{\mathcal{O}} \otimes {\mathcal{B}_i}'$, $(1 \leq i \leq n)$ and $\mathcal{O} = (\mathcal{O}_1, \dots, \mathcal{O}_n)$ is an n-tuple of quantum operations: $\mathcal{O}_i : \mathcal{L}(\mathcal{A}_i^{\mathcal{O}} \otimes \mathcal{B}_i^{\mathcal{O}}) \mapsto \mathcal{L}(\mathcal{A}_i^{\mathcal{O}} \otimes \mathcal{B}_i^{\mathcal{O}})$, $(1 \leq i \leq n)$.*

If $\Pi^{\mathcal{O}} = (\mathscr{A}, \mathscr{B}, \mathcal{O}, n)$ is an n-turn two-party protocol, then the final state of the interaction upon input $\rho_{\text{in}} \in \mathrm{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$ where $\mathcal{R}$ is a system of dimension $\dim \mathcal{A}_0 \dim \mathcal{B}_0$, is:

$$[\mathscr{A} \circledast \mathscr{B}](\rho_{\text{in}}) = (\mathbb{1}_{\mathcal{L}(\mathcal{A}'_n \otimes \mathcal{B}'_n \otimes \mathcal{R})} \otimes \mathcal{O}_n)(\mathscr{A}_n \otimes \mathscr{B}_n \otimes \mathbb{1}_{\mathcal{R}}) \dots (\mathbb{1}_{\mathcal{L}(\mathcal{A}'_1 \otimes \mathcal{B}'_1 \otimes \mathcal{R})} \otimes \mathcal{O}_1)(\mathscr{A}_1 \otimes \mathscr{B}_1 \otimes \mathbb{1}_{\mathcal{R}})(\rho_{\text{in}})\,. \tag{22}$$

As in [24], we specify that an oracle $\mathcal{O}$ can be a communication oracle or an ideal functionality oracle.

An *adversary* $\tilde{\mathscr{A}}$ for an honest party $\mathscr{A}$ in $\Pi^{\mathcal{O}} = (\mathscr{A}, \mathscr{B}, \mathcal{O}, n)$ is an n-tuple of quantum operations matching the input and outputs spaces of $\mathscr{A}$. A *simulator* for $\tilde{\mathscr{A}}$ is a sequence of quantum operations $(\mathcal{S}_i)_{i=1}^{n}$ where $\mathcal{S}_i$ has the same input-output spaces as the maps of $\tilde{\mathscr{A}}$ at step i. In addition, $\mathcal{S}$ has access to the ideal functionality for the protocol Π.

▶ **Definition 11.** *An n-step quantum two-party protocol with oracle calls, $\Pi^{\mathcal{O}} = (\mathscr{A}, \mathscr{B}, \mathcal{O}, n)$ is statistically* stand-alone *secure against a corrupt $\mathscr{A}$ if for every adversary $\tilde{\mathscr{A}}$ there exists a simulator $\mathcal{S}$ such that for every input ρ_{in},*

$$\mathrm{Tr}_{\mathcal{B}_n \otimes \mathcal{R}}(\tilde{\mathscr{A}} \circledast \mathscr{B}) = \mathrm{Tr}_{\mathcal{B}_n \otimes \mathcal{R}}(\mathcal{S} \circledast \mathscr{B})\,. \tag{23}$$

We note that Definition 11 is weaker than some other definitions for active security used in the literature, e.g., [25], because we ask only that the *local* view of the adversary be simulated.

Given the simple structure of our protocol and ideal functionality, the construction and proof of the simulator is straightforward as shown below.

▶ **Theorem 12.** *Protocol* Π *is statistically stand-alone secure against a corrupt sender.*

Proof. Since Π consists in a single message from the sender to the receiver (together with a call to the ideal functionality for the token), we have that $\mathscr{A} = (\mathscr{A}_1)$. Furthermore, since the ideal functionality $\mathcal{F}_{\texttt{wrap}}$ does not return anything to the sender, there is no need for our simulator $\mathcal{S}$ to call an ideal functionality.

We thus build $\mathcal{S}$ that runs $\mathscr{A}$ on the input in register $\mathcal{A}_0$. When $\mathscr{A}$ calls the $\mathcal{F}_{\texttt{wrap}}$ ideal functionality, the simulator does nothing. Since Π is a one-way protocol, and since the ideal functionality also does not allow communication from the receiver to the sender,

$$\mathrm{Tr}_{\mathcal{B}_n \otimes \mathcal{R}}(\tilde{\mathscr{A}} \circledast \mathscr{B}) = \mathscr{A}(Tr_{\mathcal{B}_0 \otimes \mathcal{R}}(\rho_{\mathrm{in}})) = \mathcal{S}(Tr_{\mathcal{B}_0 \otimes \mathcal{R}}(\rho_{\mathrm{in}}))\,. \tag{24}$$

This concludes the proof. ◀

C Proof of Lemma 4

For clarity, implicitly in our proof below, we model the oracle O_f as having three possible outputs: 0, 1, or 2, where 2 is output whenever O_f is fed an invalid key y. This is required for the notion of having "few" keys to make sense (i.e., there are 2^n candidate keys, and only two secret bits, each of which is supposed to have a bounded number of keys). Note that our construction indeed fits into this framework.

Proof. Observe first that an honest receiver Alice wishing to extract s_i acts as follows. She applies a unitary $U_i \in \mathcal{U}(A \otimes B)$ to get state

$$|\phi_1\rangle := U_i|\psi\rangle_{AB}|0\rangle_C. \tag{25}$$

She then measures B in the computational basis and postselects on result $y \in \{0,1\}^n$, obtaining state

$$|\phi_2\rangle := |\phi_y\rangle_A|y\rangle_B|0\rangle_C. \tag{26}$$

She now treats y as a "key" for s_i, i.e., she applies O_f to $B \otimes C$ to obtain her desired bit s_i, i.e.,

$$|\phi_3\rangle := |\phi_y\rangle_A|y\rangle_B|s_i\rangle_C. \tag{27}$$

A malicious receiver Bob wishing to extract s_0 and s_1 now acts similarly to the rewinding strategy for superposition queries. Suppose without loss of generality that s_0 has at most Δ keys. Then, Bob first applies U_0 to prepare $|\phi_1\rangle$ from Equation (25), which we can express as

$$|\phi_1\rangle = \sum_{y \in \{0,1\}^n} \alpha_y|\psi_y\rangle_A|y\rangle_B|0\rangle_C. \tag{28}$$

for $\sum_y |\alpha_y|^2 = 1$. Since measuring B next would allow us to retrieve s_0 in register C with certainty, we have that all y appearing in the expansion above satisfy $f(y) = s_0$. Moreover, since s_0 has at most Δ keys, there exists a key y' such that $|\alpha_{y'}|^2 \geq 1/\Delta$. Bob now measures B in the computational basis to obtain $|\phi_2\rangle$ from Equation (26), obtaining y' with probability at least $1/\Delta$. Feeding y' into O_f yields s_0. Having obtained y', we have that $|\langle\phi_1|\phi_2\rangle|^2 \geq 1/\Delta$, implying

$$\left|\langle\psi|U_0^\dagger|\phi_{y'}\rangle|y'\rangle\right|^2 \geq 1/\Delta, \tag{29}$$

i.e., Bob now applies $U_0^\dagger$ to recover a state with "large" overlap with initial state $|\psi\rangle$.

To next recover s_1, define $|\psi_{\text{good}}\rangle := U_1|\psi\rangle$ and $|\psi_{\text{approx}}\rangle := U_1 U_0^\dagger|\phi_{y'}\rangle|y'\rangle$. Bob applies U_1 to obtain

$$|\psi_{\text{approx}}\rangle = \beta_1|\psi_{\text{good}}\rangle + \beta_2|\psi_{\text{good}}^\perp\rangle, \tag{30}$$

where $\sum_i |\beta_i|^2 = 1$, $\langle\psi_{\text{good}}|\psi_{\text{good}}^\perp\rangle = 0$, and $|\beta_1|^2 \geq 1/\Delta$. Define

$$\Pi_{\text{good}} := \sum_{y\in\{0,1\}^n \text{ s.t. } f(y)=s_1} |y\rangle\langle y|.$$

Then, the probability that measuring B in the computational basis now yields a valid key for s_1 is

$$\langle\psi_{\text{approx}}|\Pi_{\text{good}}|\psi_{\text{approx}}\rangle \geq |\beta_1|^2 \geq \frac{1}{\Delta}, \tag{31}$$

where we have used the fact that $\Pi_{\text{good}}|\psi_{\text{good}}\rangle = |\psi_{\text{good}}\rangle$ (since an honest receiver can extract s_1 with certainty). We conclude that Bob can extract both s_0 and s_1 with probability at least $1/\Delta^2$. ◀