Computationally binding quantum commitments

Dominique Unruh

University of Tartu

Abstract. We present a new definition of computationally binding commitment schemes in the quantum setting, which we call "collapse-binding". The definition applies to string commitments, composes in parallel, and works well with rewinding-based proofs. We give simple constructions of collapse-binding commitments in the random oracle model, giving evidence that they can be realized from hash functions like SHA-3. We evidence the usefulness of our definition by constructing three-round statistical zero-knowledge quantum arguments of knowledge for all NP languages.

1 Introduction

We study the definition and construction of computationally binding string commitment schemes in the quantum setting. A commitment scheme is a two-party protocol consisting of two phases, the commit and the open phase. The goal of the commitment is to allow the sender to transmit information related to a message m during the commit phase in such a way that the recipient learns nothing about the message (hiding property). But at the same time, the sender cannot change his mind later about the message (binding property). Later, in the open phase, the sender reveals the message m and proves that this was indeed the message that he had in mind earlier. We will focus on non-interactive classical commitments, that is, the commit and open phase consists of a single classical message. However, the adversary who tries to break the binding or hiding property will be a quantum-polynomial-time algorithm. At the first glance, it seems that the definition of the binding property in this setting is straightforward; we just take the classical definition but consider quantum adversaries instead of classical ones:

Definition 1 (Classical-style binding – informal). *No quantum-polynomial-time algorithm A can output, except with negligible probability, a commitment c (i.e., the message sent during the commit phase) as well as two openings u, u' that open c to two different messages m, m'.*

(Formal definition in Section 2.) Unfortunately, this definition turns out to be inadequate in the quantum setting. Ambainis, Rosmanis, and Unruh [1] show the existence of a commitment scheme (relative to a special oracle) such that: The commitment is classical-style binding. Yet there exists a quantum-polynomial-time adversary A that outputs a commitment c, then expects a message m as input, and then provides valid opening information for c and m. That is, the adversary can open the commitment c to any message of his choosing, even if he learns that message only after committing. This is in clear contradiction

to the intuition of the binding property. How is this possible, as Definition 1 says that the adversary cannot produce two different openings for the same commitment? In the construction from [1], the adversary has a quantum state $|\Psi\rangle$ that allows him to compute one opening for a message of his choosing, however, this computation will destroy the state $|\Psi\rangle$. Thus, the adversary cannot compute two openings simultaneously, hence the commitment is classically-binding. But he can open the commitment to an arbitrary message once, which shows that the commitment scheme is basically useless despite being classically-binding.[1]

1.1 Prior definitions

We now discuss various definitions that appeared in the literature and that circumvent the above limitation of the classical-binding property. (We do not discuss the hiding property here, because that one does not have any comparable problems. See Definition 10 below for the definition of hiding.) In each case, we discuss some limitations of the definitions to motivate the need for a new definition for computationally binding commitments. The reader only interested in our results can safely skip this section.

Sum-binding. The most obvious solution is to simply require that the adversary cannot open successfully to each of two messages: That is:

Definition 2 (Sum-binding – informal). *Consider a bit commitment scheme. (I.e., one can only commit to $m = 0$ or $m = 1$.)*

Given an adversary A, let p_b be the probability that the recipient accepts in the following execution: A commits, then A is given b, and then A provides opening information for message b. A commitment is sum-binding *iff for any quantum-polynomial-time adversary A, $p_0 + p_1 \leq 1 + negligible$.*

Note that even with an ideal commitment, $p_0 + p_1 = 1$ is possible (the adversary just picks $b := 0$ in the commit phase with probability p_0, and $b := 1$ else). So $p_0 + p_1 \leq 1 + negligible$ is the best we can expect if we allow for a negligible probability of an attack. The sum-binding definition has occurred implicitly and explicitly in different variants in [4,15,13,6,8]. We use the name sum-binding here to distinguish it from the other definitions of binding discussed here since it does not have an established name.

Although it avoids the attack described above, the sum-binding definition has a number of disadvantages:

– It is specific to the bit commitment case. There is no straightforward generalization to the the string commitment case (i.e., where the message m does not have to be a single bit). See [6] for discussion why obvious approaches fail.

[1] Note that for classical adversaries, the classical-binding property gives useful guarantees: If an adversary can produce an opening for any message m using some classical algorithm, he can also produce two openings for different messages m, m' by running that algorithm twice.

– It is unclear how the definition behaves when we use the commitment several times. (I.e., it is not clear how it behaves under composition.) For example, given bits $m_1, \ldots, m_n$, what are the security guarantees if we commit to each of the m_i? (Be it in parallel, or sequentially.) Basically, we would expect that all commitments together form a binding commitment on the string $m = m_1 \ldots m_n$, but this is something we cannot even express using the sum-binding definition.
– It is not clear how useful sum-binding commitments are as subprotocols in larger protocols. That is, is the sum-binding property strong enough to allow to prove the security of complex protocols using commitments? While there are constructions of sum-binding in the literature (e.g., [13]), we are not aware of research where (computational) sum-binding commitments are used as subprotocols.

CDMS-binding. Crépeau, Dumais, Mayers, and Salvail [6] suggest a generalization of the sum-binding property to string commitments. The basic idea is: Instead of bounding $p_0 + p_1 \leq 1 + \textit{negligible}$ where p_m is the probability that the adversary open his commitment as $m \in \{0,1\}$, we could bound $\sum_m p_m \leq 1 + \textit{negligible}$ where m ranges over all bitstrings. However, as discussed in [6], this would be too strong a requirement. (Basically, this is because the sum $\sum_m p_m$ has exponentially many summands, so even negligible attack probabilities can add up to large probabilities.) Instead, they proposed the following definition:

Definition 3 (CDMS-binding – informal). *Let F be a family of functions. Fix a string commitment scheme. For $f \in F$, let $\tilde{p}_y^f$ be the probability that the recipient accepts in the following execution: A commits. A gets y. A tries to open the commitment to some m with $f(m) = y$.*

We call the commitment scheme F-CDMS-binding iff for all adversaries A and all $f \in F$, we have $\sum_y \tilde{p}_y^f \leq 1 + \textit{negligible}$.

Now if all $f \in F$ have a polynomial-size range, the sum $\sum_y \tilde{p}_y^f$ will have polynomially many summands. The intuition behind this definition is that every function $f \in F$ represents some property of the committed message m (e.g., $f(m)$ is the parity of m). Then, if a commitment scheme is F-CDMS-binding, this intuitively means that the although the adversary might be able to change his mind about the message m, he cannot change his mind about $f(m)$. (E.g., if the parity function is in F, this means that the adversary will be committed to the parity of the message m.) [6] successfully used this definition (for a specific class F) to show that using quantum communication and a commitment, we can construct an oblivious transfer protocol. (Note however that their protocol is different and more complex than the original OT protocol from [2].)

Although the CDMS-binding definition generalizes the sum-binding definition to the case of string commitments, it comes with its own challenges:
– The definition is parametrized by a specific family F of functions that specifies in which way the commitment should be binding. This function family has to be chosen dependent on the particular use case. This makes the definition less universal and canonical.

- To the best of our knowledge, no construction of CDMS-binding commitments is known. Crépeau et al. [6] conjecture that the protocol from [7] can be extended to a CDMS-binding one for functions F with small range, but no proof or construction is given.
- It is not known whether the definition is composable. If we commit to messages $m_1, \ldots, m_n$ individually using F-CDMS-binding commitments, does this constitute an F'-CDMS-binding commitment on $m := m_1 \| \ldots \| m_n$? If so, for which F'?
- While CDMS-binding commitments have successfully been used in a larger protocol (namely, the OT protocol from [6]), we believe that in many contexts, the definition is still not very easy to use. At least in classical cryptography, one often uses the fact that it is possible to extract the committed message by rewinding (basically, one runs the open phase, saves the opened message, and rewinds to before the opening phase). It is not clear how to do that with CDMS-binding commitments. For example, it is not clear how one could use CDMS-binding commitments in the construction of sigma-protocols that are quantum arguments of knowledge (as done in Section 7 below using our definition of binding commitments).

Perfectly-binding commitments. One possibility to solve all the problems mentioned so far is simply to use perfectly-binding commitments.

Definition 4 (Perfectly-binding – informal). *A commitment scheme is perfectly-binding if there exists no tuple (c, m, u, m', u') with $m \neq m'$ such that u is a valid opening for c with message m, and u' is a valid opening for c with message m'.*

However, if we restrict ourselves to perfectly-binding commitments, we get the following disadvantages:

- A perfectly-binding commitment cannot be statistically hiding [15]. That is, the hiding property cannot hold against computationally unlimited adversaries. That means that we give up on information-theoretical security for one party just because we do not have a suitable definition for the computationally-binding property. For example, the constructions in [19] are only computational zero-knowledge (not statistical zero-knowledge) because perfectly-binding commitments are used.
- Perfectly-binding commitments cannot be short. That is, the length of the commitment must be as long as the length of the committed message. So by using only perfectly-binding commitments, we may lose efficiency.

UC commitments. One further possibility is to use commitments that are UC-secure [18]. Since the security of a protocol using a UC-secure commitment can be reduced to the security of the same protocol using an ideal (in particular perfectly-binding) commitment, UC-secure commitments are easy to use. Yet, this solution again comes with disadvantages:

- UC-commitments do not exist without the use of additional setup such as, e.g., a common reference strings (CRS). It is possible to chose the CRS in a pre-computation phase using a coin-toss protocol [12]. But that increases

the round complexity of the resulting protocol (and, incidentally, loses the UC security and possibly even the concurrent composability of the resulting protocol).
- In the construction of UC-secure commitment schemes, trapdoors are used that allow the simulator to extract the committed message. This implies that constructions of UC-secure commitment are usually more complex, less efficient, and use stronger computational assumptions.
- At least when using a CRS, UC commitments cannot be short.

Damgård, Fehr, Lunemann, Salvail, and Schaffner [9] use so-called dual-mode commitments, these are somewhat weaker than UC commitments. Yet, they also use extraction using a trapdoor in the CRS. Hence the disadvantages of UC commitments apply to dual-mode commitments as well.

Q-binding. Damgård, Fehr, and Salvail [11] give another definition for computationally binding string commitments. Intuitively, the definition says that an adversary who uses the commitment has negligible advantage in a "betting game" over an adversary that has to use perfect commitments. Here, a betting game is represented as an arbitrary predicate on the opened values in the commitments, and on some random input that the adversary learns only after committing. (E.g., a bet could be: the sum of all opened values equals the random value u that the adversary learns just before opening.) Somewhat more formally:

Definition 5 (Q-binding – informal). *For an adversary A and an predicate Q, consider the following game: A outputs commitments $C_1, \ldots, C_N$. Then A gets a random bitstring u. Then A opens a subset $\mathbf{A}$ of the commitments, let $(s_i)_{i\in\mathbf{A}}$ be the contents. A wins if $Q(\mathbf{A}, (s_i)_{i\in\mathbf{A}}, u) = 1$.*

A commitment scheme is Q-binding iff for any quantum-polynomial-time A and any predicate Q, the adversary A wins with probability at most $p_{\mathsf{IDEAL}} + negl$, where p_{IDEAL} is the maximum winning probability when using a perfectly binding commitment.

The definition overcomes some of the problems of the CDMS-binding definition. In particular, there is no need to parametrize the definition with a class F of functions, specifically chosen to fit the use case at hand. Also, the Q-binding definition composes in parallel: if a commitment scheme is Q-binding, then the commitment scheme resulting from committing to each of $m_1, \ldots, m_n$ individually is Q-binding, too. (This should come as no surprise, since the Q-binding definition itself explicitly refers to a polynomial number of parallel copies of the commitment scheme.) The definition seems particularly well-suited for commit-and-choose constructions (i.e., where one party commits to a set of values, and the other party selects which of them should be opened), since security when opening a specific subset is built into the definition. [11] give a generic construction for unconditionally hiding Q-binding equivocal trapdoor commitments from a certain class of sigma-protocols. They show that using such commitments, sigma-protocols can be converted into statistical quantum zero-knowledge arguments in the CRS model.

However, their definition also comes with a number of challenges:

– The only construction of unconditionally hiding Q-binding commitments known is actually an equivocal trapdoor commitment. Trapdoor commitments usually need stronger assumptions. Note also that no protocols using non-equivocal Q-binding commitments are known (the zero-knowledge protocols in [11] need the trapdoor because they are constructed following the "no quantum rewinding paradigm"). And, due to the absence of rewinding, the zero-knowledge protocols only work in the CRS model.
– The possibility for parallel composition might be limited: It follows directly from the definition that Q-binding commitments on $m_1, \ldots, m_n$ are a Q-binding commitment on $m = m_1 \ldots m_n$. However, it is not clear what happens if we commit to $m_1, \ldots, m_n$ using *different* Q-binding commitments. (Or the same Q-binding commitment, but using different public keys.)
– The definition is specialized for the commit-and-choose paradigm. It is unclear how it can be used in rewinding-based proofs. (On the other hand, in commit-and-choose situations, Q-binding commitments might be more suitable than those we propose; whether this is the case constitutes future work.)

Summarizing, Q-binding commitments seem to be well suited for commit-and-choose constructions, but for proofs involving rewinding, we need another definition.

DFRSS-binding. Damgård, Fehr, Renner, Salvail, and Schaffner [10] presented a definition for the unconditional binding property, targeted mainly for the bounded quantum storage model; the following is a direct adaptation of their definition to the computational setting:

Definition 6 (DFRSS-binding – adapted). *In a commitment, let V denote the recipient's classical state, and Z the sender's classical state.*

A bit commitment is DFRSS-binding iff for any quantum-polynomial-time sender $\tilde{C}$, there exists a randomized function B' such that the following holds:

Let $\tilde{C}$ and the honest recipient execute the commit phase. Compute $b' := B'(V, Z)$. Let $\tilde{C}(b')$ and the honest recipient execute the open phase. Let b denote the opened bit (or $\perp$ if the recipient does not accept). Then $\Pr[b' \neq b]$ is negligible.

In other words, given the classical part of the state of the recipient *and* the sender, it is possible to extract what bit the sender will open to. (The extraction does not have to be efficiently feasible.) The definition can be extended to string commitments by letting B' range over bitstrings.

We have changed the original definition from [10] to refer to quantum-polynomial-time adversaries. (We also reformulated it for easier readability, changing a number of technical details in the process. However, the current definition is in the spirit of the original. And our discussion also applies to the original formulation.)

The definition was originally intended for protocols in the bounded quantum storage model. What happens if we use it in the standard model, i.e., with no limit on the quantum memory of the sender? In this case, it is always possible for the malicious sender to perform all his operations in superposition, and only the recipient will perform measurements. Then, in Definition 6, the register Z will be

empty. Hence the definition requires that the committed bit b' can be computed from the recipient's state V alone. This immediately implies that the scheme cannot be statistically hiding, and that the commitments cannot be shorter than the message.

Hence the DFRSS-binding definition shares the drawbacks of the perfectly binding definition, unless we are in the bounded quantum storage model. (We stress that [10] never claimed that the definition should be used outside the bounded quantum storage model.)

1.2 Our contribution

We give a new definition for the computational-binding property for commitment schemes, called "collapse-binding" (Section 2). This definition is composable (several collapse-binding commitments are also collapse-binding together), works well with quantum rewinding (see below), does not conflict with statistical hiding (as perfectly-binding commitments would), allows for short commitments (i.e., the commitment can be shorter than the committed message, in contrast to perfectly-binding commitments, and to extractable commitments in the CRS model). Basically, collapse-binding commitments seem to be in the quantum setting what computationally-binding commitments are in the classical setting.

We show that collision-resistant hash functions are not sufficient for getting collapse-binding or even just sum-binding commitments (Section 3), at least when using standard constructions, and relative to an oracle. We present a strengthening of collision-resistant hash functions, "collapsing hash functions" that can serve as a drop-in replacement for collision-resistant hash functions (Section 4). Using collapsing hash functions, we show several standard constructions of commitments to be collapse-binding (Section 5).

We conjecture that standard cryptographic hash functions such as SHA-3 [17] are collapsing (and thus lead to collapse-binding commitments). We give evidence for this conjecture by proving that the random oracle is a collapsing hash function.

We show that the definition of collapse-binding commitments is usable by extending the construction of quantum proofs of knowledge from [19] (Section 7). Their construction uses perfectly-binding commitments (actually, strict-binding, which is slightly stronger) to get proofs of knowledge. We show that when replacing the perfectly-binding commitments with collapse-binding ones, we get statistical zero-knowledge quantum arguments of knowledge. In particular, this shows that collapse-binding commitments work well together with rewinding.

1.3 Our techniques

Collapse-binding commitments. To explain the definition of collapse-binding commitments, first consider a perfectly-binding commitment. That is, when an adversary A outputs a commitment c, there is only one possible message m_c that A can open c to. Hence, if the adversary A outputs a superposition of messages that he can open c to, that superposition will necessarily be in the state $|m_c\rangle$. Hence, we can characterize perfectly-binding commitments by requiring: when an

adversary outputs a superposition of messages that he can open the commitment c to, that superposition will necessarily be a single computational basis vector (i.e., no non-trivial superposition).

To express this more formally, consider the circuit in Figure 1 (a). Here the adversary A outputs a commitment c (classical message). Furthermore, he outputs three quantum registers S, U, M. S contains his state. M is supposed to contain a superposition of messages, U a superposition of corresponding opening informations. Then we apply the measurement V_c. This measurement measures whether U, M contain matching opening information/message. More formally, V_c measures whether U, M is a superposition of states $|u, m\rangle$ such that u is valid opening information for message m and commitment c. Let $ok = 1$ if the measurement succeeds. Then we feed the registers S, U, M back to the second part B of the adversary. B outputs a classical bit b. As discussed before, a commitment is perfectly-binding iff for all adversaries A, the state of M after measuring $ok = 1$ is a computational basis vector.

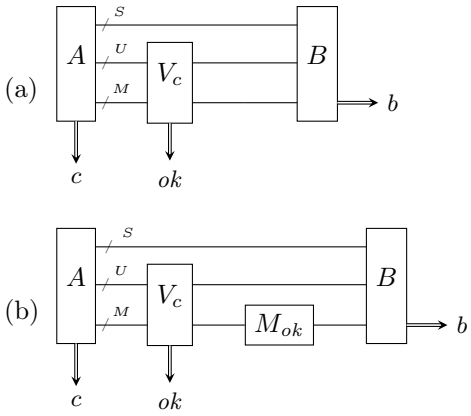

Fig. 1: Games from the definition of collapse-binding commitments.

The state of a register is a computational basis vector (or, synonymously: is in a collapsed state) iff measuring that register in the computational basis does not change that state. Consider the circuit in Figure 1 (b). Here we added a measurement M_{ok} on M after V_c. M_{ok} is a complete measurement in the computational basis, but is executed only if $ok = 1$. Since M_{ok} disturbs the state of M iff that state is not a computational basis vector, we can rephrase the definition of perfectly-binding commitments:

A commitment is perfectly-binding iff, for all computationally unlimited adversaries A, B, $\Pr[b = 1]$ is equal in Figures 1 (a) and 1 (b) where b is the output (i.e., guess) of B.[2]

Now we are ready to weaken this characterization to get a computational binding property. Basically, we require that the same holds for quantum-polynomial-time adversaries:

Definition 7 (Collapse-binding – informal). *A commitment is collapse-binding iff, for all quantum-polynomial-time adversaries A, B, $\Pr[b = 1]$ in Figure 1 (a) is negligibly close to $\Pr[b = 1]$ in Figure 1 (b).*

In other words, with a perfectly-binding commitment, the adversary cannot produce a superposition of different messages that are contained in the commitment.

[2] Our exposition above was not very rigorous, but it is easy to see that this is indeed an "if and only if".

But with a collapse-binding commitment, the adversary is forced to produce a state *that looks like it is not a superposition* of different messages. For the purpose of computational security, this will often be as good.

We quickly explain why collapse-binding commitments work well with quantum rewinding. In the case of quantum rewinding (e.g., in the analysis of proofs of knowledge [19]), one problem is that we might need to run an adversary until he opens a commitment c, then to measure the opened message, and then to go back to an earlier state by applying the inverse of the adversary. The problem is that measuring the opened message will disturb the state of the adversary, and thus make rewinding impossible. Except: if the opened message cannot be distinguished from being already in a collapsed state (as guaranteed by collapse-binding), then measuring the opened message does not disturb the state in a noticeable way and we can rewind. (See the discussion on arguments of knowledge below.)

Constructing collapse-binding commitments. Collapse-binding commitments are useful only if they exist. Perfectly-binding commitments are easily seen to be collapse-binding, but then we cannot have statistically hiding or short commitments. In the classical setting, we get practical computationally-binding commitments from a collision-resistant hash function H. The most obvious construction is to send $c := H(m\|u)$ for uniformly random u of suitable length. We call this the "canonical commitment". The canonical commitment is easily seen to be classical-style binding if H is collision-resistant, and it is statistically hiding if H is a random oracle. To get rid of the random-oracle requirement, we can use a somewhat more complex constructions by Halevi and Micali [14] instead. Unfortunately, both the canonical commitment and the Halevi-Micali commitments are not collapse-binding if H is merely collision-resistant. In fact, relative to a specific oracle and using a specific collision-resistant hash function, there is a total break where the adversary can unveil the commitment to any message of his chosing. To show this, we tweak the technique from [1] to construct a hash function H such that the adversary can sample an image c of H together with a quantum state $|\Psi\rangle$ such that: Given the state $|\Psi\rangle$, for any m, the adversary can find a random u with $H(m\|u) = c$. But this process destroys $|\Psi\rangle$, so the adversary cannot find two preimages of c; the hash function is collision-resistant. But the canonical commitment, based on this H, is trivially broken. Similar constructions break the Halevi-Micali commitments.

Since collision-resistance seems too weak a property in the quantum setting (at least for our purposes), we give a strengthening of collision-resistance: collapsing hash functions:

Definition 8 (Collapsing hash function – informal). *An adversary is valid if he outputs a classical value c, and a register M containing a superposition of messages m with $H(m) = c$. We call H collapsing iff no quantum-polynomial-time adversary can distinguish whether we measure M in the computational basis or not, before giving the register M back to the adversary. (This is formalized with games similar to those in Figure 1.)*

We can show that collapsing hash functions are collision-resistant, and they share a number of structural properties with collision-resistant functions. E.g., injective functions are collapsing, and the composition $H \circ H'$ of collapsing functions is collapsing.

Due to the similarity between the definition of collapsing hash functions and collapse-binding commitments, we can show that the canonical commitment and the Halevi-Micali commitments are collapse-binding if H is collapsing.

However, this leaves the question: do collapsing functions exist in the first place? We conjecture that common industrial hash function like SHA3 [17] are actually collapsing (not only collision-resistant). In fact, we argue that the collapsing property should be a requirement for the design of future hash functions (in the sense that a hash function where the collapsing property is in doubt should not be selected for industry standards), since collision-resistance is not sufficient if we wish to achieve post-quantum secure cryptography. We support our conjecture that sufficiently unstructured functions are collapsing by proving that the random oracle is collapsing:

Random oracles are collapsing. We now sketch on a high level our proof that random oracles are collapsing, or, equivalently, that a random function is collapsing with high probability. In our analysis, we assume that the adversary can query the random oracle on the superposition of different inputs; this is necessary for having a realistic modeling of hash functions [3]. As a first step, we identify a new property, "half-collision resistance":

Definition 9 (Half-collision resistance – informal). *A half-collision of H is a string x such that there exists an $x' \neq x$ with $H(x') = H(x)$. A hash function H is half-collision resistant if no adversary does the following: He outputs a half-collision with non-negligible probability. And he never outputs a non-half-collision. (The adversary may output $\perp$ though.)*

That is, half-collision resistance says that the adversary cannot find non-injective inputs to H without sometimes accidentally outputting injective inputs. We show: if H is half-collision resistant, it is collapsing.

The proof idea is: if H is not collapsing, the adversary can produce a superposition M of messages m with $H(m) = c$ and notice whether M is being measured. The latter implies that M must be a superposition of at least two messages m with $H(m) = c$. Hence by measuring M, the adversary gets a half-collision. Much additional work is needed to make sure that the adversary does not accidentally measure the register M when it is not a nontrivial superposition.

(The half-collision resistance property might be useful independent of the proof that the random oracle is collapsing. When trying to construct collapsing hash functions based on other assumptions, half-collision resistance might be easier to verify since its definition consists of purely classical games.)

Next we construct a random function $H^* : X \to Y$ with $|Y| = \frac{2}{3}|X|$. That is, H^* is slightly compressing. The domain of H^* is partitioned into two sets X_1, X_2 with $|X_1| = 2|X_2|$. H^* is injective on X_2, and 2-to-1 on X_1. Besides those constraints, H^* is uniformly random. We can then show that H^* is half-collision

resistant. (Basically, this means that the adversary cannot identify the subset X_1.) Furthermore, we can show that H^* is indistinguishable from a random function $H : X \to Y$. Since H^* is half-collision resistant, it is collapsing. And since H is indistinguishable from H^*, H is collapsing.

We now know that random functions $H : X \to Y$ are collapsing if $|Y| = \frac{2}{3}|X|$ (i.e., if they are slightly compressing). However, we want that H is collapsing for arbitrary X and Y, as long as Y has superpolynomial size. For $|X| \leq |Y|$, H is indistinguishable from a random injection, which in turn is collapsing. The interesting case is $|X| > |Y|$ (namely, when H is compressing). In this case, we show (following an idea from [24]) that H can be written as $H = f_n \circ \cdots \circ f_1$ where all f_i are slightly compressing. Since all f_i are collapsing, so is H. This shows that a random function H is collapsing, in other words, that the random oracle is collapsing (if its range has superpolynomial size).

Quantum arguments of knowledge. We illustrate the use of collapse-binding commitments by revisiting the construction of proofs of knowledge from Unruh [19]. Unruh showed that a sigma-protocol (i.e., a particular kind of three round proof system) is a quantum proof of knowledge if it has two properties: *special soundness* (from two interactions with the same first and different second messages one can efficiently compute a witness) and *strict soundness* (the first and second message of a valid interaction determine the third). In the classical setting, only special soundness is needed. In the quantum setting, strict soundness is additionally required to allow for quantum rewinding: In the proof from [19], we run the malicious prover to get his response (the third message). Then we measure the response. Then we rewind the prover (by applying the inverse of the unitary transformation representing the prover). Then we run the prover again to get a second answer. Special soundness then implies that from the two responses, we get a witness. However, we need to make sure that measuring the prover's response before rewinding does not disturb the state (too much). In [19], this follows from strict soundness: strict soundness guarantees that the response is uniquely determined, and thus measuring the response does not disturb the state. To achieve strict soundness, [19] lets the prover commit to all possible responses in the first message using perfectly-binding commitments.[3] The drawback of this solution is that the commitments cannot be statistically hiding, so we cannot get statistical zero-knowledge proofs using the method from [19].

What happens if we replace the perfectly-binding commitments by collapse-binding commitments containing the response? In that case, the response will not necessarily be information-theoretically determined by the first two messages. However, the definition of collapse-binding commitments guarantees that measuring that response will be indistinguishable from not measuring it. Thus, if we measure the response, the state might be disturbed, but it will be computationally indistinguishable from not being disturbed. This is enough for the proof technique from [19] to go through, assuming the prover is computationally limited. The resulting protocol will not be a quantum proof of knowledge, but a quantum

[3] Actually, "strict-binding commitments" but this distinction is not relevant for this exposition.

argument of knowledge (i.e., secure only against computationally limited provers). But in contrast to [19], the proof system will be statistical zero-knowledge.

To summarize: from collapse-binding commitments (or from collapsing hash functions), we get three-round statistical zero-knowledge quantum arguments of knowledge for all languages in NP (with inverse polynomial knowledge error). To the best of our knowledge, not even three-round statistical zero-knowledge quantum *arguments* were known before.

1.4 Related work.

Commitments. Brassard, Crépeau, Jozsa, and Langlois [4] presented an information-theoretically hiding and binding commitment scheme using quantum communication. However, the protocol was flawed, Mayers [15] showed that information-theoretically hiding and binding commitments are impossible. (This is no contradiction to our results, because our commitments are not information-theoretically binding.) Dumais, Mayers, and Salvail [13] and Crépeau, Légaré, and Salvail [7] constructed statistically hiding commitments from quantum one-way permutations/functions, respectively. Their protocols use quantum communication, and are sum-binding. Crépeau, Dumais, Mayers, and Salvail [6] generalized the sum-binding definition to string commitments and constructed an OT protocol based on that definition. (However, it is not known whether the protocol composes even sequentially.) Damgård, Fehr, Lunemann, Salvail, and Schaffner [9] and Unruh [18] showed a much simpler OT protocol to be secure, assuming much stronger commitment definitions in the CRS model, but achieving stronger security notions (sequential composability/UC). Ambainis, Rosmanis, and Unruh [1] show that classical-style binding commitments are not necessarily even sum-binding.

Quantum random oracles. Random oracles were first explicitly considered in a quantum cryptographic context by Boneh, Dagdelen, Fischlin, Lehmann, Schaffner, and Zhandry [3] who stressed that the adversary should have superposition access to the random oracle. Zhandry [24] showed that the random oracle is collision-resistant. In contrast, we show (based on his result) that the random oracle is collapsing (a stronger property).

Quantum rewinding and proof systems. Watrous [23] showed how quantum rewinding can be used to prove the security of quantum zero-knowledge protocols. Unruh [19] showed how a different flavor of quantum rewinding can be used for proving the security of quantum proofs of knowledge; we extend their technique to quantum arguments of knowledge.

2 Definitions and basic properties

Preliminaries. For the necessary background in quantum computing, see, e.g., [16]. By $|i\rangle$ with $i \in I$ we denote the vectors of the computational basis of the Hilbert space with dimension $|I|$. We also use the symbol $|\cdot\rangle$ to refer to other (non-basis) vectors (e.g., $|\Psi\rangle$). And $\langle\Psi|$ is the conjugate transpose of $|\Psi\rangle$. $\|x\|$

refers to the Euclidean or ℓ^2-norm. We only consider finite dimensional Hilbert spaces. We denote $|+\rangle := \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and $|-\rangle := \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. For a linear operator A on a Hilbert space, we denote by $A^\dagger$ its conjugate transpose. We denote by I the identity. We call an operator A on a Hilbert space a projector iff it is an orthogonal projector, i.e., a linear map with $P^2 = P$ and $P = P^\dagger$. By $\mathrm{TD}(\rho, \rho')$ we denote the trace distance between ρ and ρ', and by $F(\rho, \rho')$ the fidelity.

Given an algorithm A, let $x \leftarrow A(y)$ denote the result of running A with inputs y, and assigning the output to x. Let $x \xleftarrow{\$} M$ denote assigning a uniformly random element of M to x. We will use η to denote the security parameter, that is a positive integer that will be passed to all algorithms and adversaries and that indicates the required security level. By $a\|b$ we denote the concatenation of bitstrings a and b.

We call an algorithm quantum-polynomial-time if it is a quantum algorithm and its runtime is bounded by a polynomial in its input length with probability 1. We call an algorithm classical-polynomial-time if it performs only classical operations and its runtime is bounded by a polynomial in its input length with probability 1. We write 1^η for a bitstring (of 1's) of length η. (The latter is useful for making algorithms run in polynomial-time in the length of the security parameter, e.g., $A(1^\eta)$ will run polynomial-time in η.)

Commitments. A commitment scheme $(com, verify)$ consists of a quantum-polynomial-time algorithm com and a deterministic quantum-polynomial-time algorithm $verify$.[4] $(c, u) \leftarrow com(1^\eta, m)$ returns a commitment c and the opening information u for the message m and security parameter η. c alone is supposed not to reveal anything about m (hiding). To open, we send (m, u) to the recipient who checks whether $verify(1^\eta, c, m, u) = 1$. Both com and $verify$ have classical input and output. com has a well-defined message space MSP_η that also depends on the security parameter η (e.g., $\{0,1\}^\eta$). Furthermore, for technical reasons, we assume that it is possible to find triples (c, m, u) with $verify(1^\eta, c, m, u) = 1$ with probability 1 in quantum-polynomial-time in η.

We first state some standard properties of commitments.

Definition 10. *Let $(com, verify)$ be a commitment scheme. We define:*

- ***Perfect completeness:*** *$(com, verify)$ has* perfect completeness *iff for all $m \in \mathsf{MSP}_\eta$, $\Pr[verify(1^\eta, c, m, u) = 1 : (c, u) \leftarrow com(1^\eta, m)] = 1$.*
- ***Computational hiding:*** *$(com, verify)$ is* computationally hiding *iff for any quantum-polynomial-time A and any polynomial ℓ, there is a negligible μ such that for any η, any $m_0, m_1 \in \mathsf{MSP}_\eta$ with $|m_0|, |m_1| \leq \ell(\eta)$, and any $|\Psi\rangle$,[5] $\left|P_0 - P_1\right| \leq \mu(\eta)$ where $P_i := \Pr[b = 1 : (c, u) \leftarrow com(1^\eta, m_i), b \leftarrow A(1^\eta, |\Psi\rangle, c)]$.*

[4] To be practical, those algorithms should of course be classical. We allow quantum-polynomial-time algorithms here to state our results in greater generality.

[5] $|\Psi\rangle$ is the auxiliary input of A that represents knowledge of A acquired, e.g., in prior protocol runs. One could use a mixed state instead, this would lead to an equivalent definition.

– ***Statistical hiding:*** *Like computational hiding, except that we quantify over all A (not just quantum-polynomial-time A).*

Definition 11 (Classical-style binding). *A commitment scheme is* classical-style binding *iff for any quantum-polynomial-time algorithm A, the following is negligible in η:* $\Pr[\mathit{verify}(1^\eta, c, m, u) = 1 \wedge \mathit{verify}(1^\eta, c, m', u') = 1 \wedge m \neq m' : (c, m, u, m', u') \leftarrow A(1^\eta)]$.

Definition 12 (Collapse-binding). *For algorithms A, B, consider the following games:*

$$\mathsf{Game}_1\colon\ (S, M, U, c) \leftarrow A(1^\eta), \mathit{ok} \leftarrow V_c(M, U), m \leftarrow M_{\mathit{ok}}(M), b \leftarrow B(1^\eta, S, M, U)$$
$$\mathsf{Game}_2\colon\ (S, M, U, c) \leftarrow A(1^\eta), \mathit{ok} \leftarrow V_c(M, U), \qquad b \leftarrow B(1^\eta, S, M, U)$$

Here S, M, U are quantum registers. V_c is a measurement whether M, U contains a valid opening, formally V_c is defined through the projector $\sum_{\mathit{verify}(1^\eta,c,m,u)=1}^{m,u} |m\rangle\langle m| \otimes |u\rangle\langle u|$. M_{ok} is a measurement of M in the computational basis if $\mathit{ok} = 1$, and does nothing if $\mathit{ok} = 0$ (i.e., it sets $m := \perp$ and does not touch the register M).

A commitment scheme is collapse-binding *iff for any quantum-polynomial-time algorithms A, B, the difference $\left|\Pr[b = 1 : \mathsf{Game}_1] - \Pr[b = 1 : \mathsf{Game}_2]\right|$ is negligible.*

Instead of measuring using V_c whether the adversary outputs a correct opening information, we can quantify only over adversaries that always output correct opening information. This leads to the following equivalent definition of collapse-binding commitments. This definition is often easier to handle when proving that a given scheme is collapse-binding.

Definition 13 (Collapse-binding – variant). *For algorithms A, B, consider the following games:*

$$\mathsf{Game}_1:\quad (S, M, U, c) \leftarrow A(1^\eta),\ m \leftarrow M_{\mathit{comp}}(M),\ b \leftarrow B(1^\eta, S, M, U)$$
$$\mathsf{Game}_2:\quad (S, M, U, c) \leftarrow A(1^\eta), \qquad\qquad b \leftarrow B(1^\eta, S, M, U)$$

Here S, M, U are quantum registers. $M_{\mathit{comp}}(M)$ is a measurement of M in the computational basis.

We call an adversary (A, B) valid *if $\Pr[\mathit{verify}(c, m, u) = 1] = 1$ when running $(S, M, U, c) \leftarrow A(1^\eta)$ and measuring M, U in the computational basis to obtain m, u.*

A commitment scheme is collapse-binding *iff for any quantum-polynomial-time valid adversary (A, B), the difference $\left|\Pr[b = 1 : \mathsf{Game}_1] - \Pr[b = 1 : \mathsf{Game}_2]\right|$ is negligible.*

In [20], we show Definitions 12 and 13 equivalent, and that the collapse-binding property is preserved under parallel composition of commitments.

3 Commitments from collision-resistant hash functions

In the following, we will often refer to hash functions. We will always assume that a hash function depends implicitly on the security parameter (in particular, the size of the range can depend on the security parameter). We also assume that the hash function is quantum-polynomial-time computable (in η and the input length).[6] Besides that, we do not assume any further properties such as collision-resistance unless explicitly mentioned.

Definition 14 (Canonical commitment scheme). *Given a hash function H and a parameter $\ell_u = \ell_u(\eta)$, the* canonical commitment scheme *for H is:*

- *Message space $\mathsf{MSP}_\eta := \{0,1\}^*$.*
- *$com_{can}(m)$: Pick $u \xleftarrow{\$} \{0,1\}^{\ell_u}$. Compute $c := H(m\|u)$. Return (c,u).*
- *$verify_{can}(c,m,u)$: Return 1 iff $H(m\|u) = c$.*

It is immediate to see that this scheme is classical-style binding if H is collision-resistant. However, in general it will not be hiding; for example, $H(m\|u)$ could leak the first bit of m. However, it is hiding if H is a random oracle:

Lemma 15. *Fix $\ell_u \geq 0$ and assume that $|Y| \leq 2^{\ell_u/8}$. For a random oracle $H : X \to Y$, the canonical commitment is statistically hiding.*

When using a hash function in the standard model, we can use the following commitment scheme instead:

Definition 16 (Bounded-length Halevi-Micali commitment [14]). *Fix integers $\ell = \ell(\eta)$, $n = n(\eta)$. Let $L := 4\ell + 2n + 4$. Let $H : \{0,1\}^L \to \{0,1\}^\ell$ be a hash function. Let $F = F(\eta)$ be a family of universal hash functions $f : \{0,1\}^L \to \{0,1\}^n$. We define the* bounded-length Halevi-Micali commitment *$(com_{HMb}, verify_{HMb})$ with $\mathsf{MSP}_\eta = \{0,1\}^n$ as:*

- *$com_{HMb}(m)$: Pick $f \in F$ and $u \in \{0,1\}^L$ uniformly at random, conditioned on $f(u) = m$. Compute $h := H(u)$. Let $c := (h,f)$. Return (c,u).*
- *$verify_{HMb}(c,m,u)$ with $c = (h,f)$: Check whether $f(u) = m$ and $h = H(u)$. If so, return 1.*

Definition 17 (Unbounded Halevi-Micali commitment [14]). *Fix an integer $\ell = \ell(\eta)$. Let $H : \{0,1\}^* \to \{0,1\}^\ell$ be a hash function. Let $L := 6\ell + 4$. Let F be a family of universal hash functions $f : \{0,1\}^L \to \{0,1\}^\ell$. We define the* unbounded Halevi-Micali commitment *$(com_{HMu}, verify_{HMu})$ as:*

- *$com_{HMu}(m)$: Pick $f \in F$ and $u \in \{0,1\}^L$ uniformly at random, conditioned on $f(u) = H(m)$. Compute $h := H(u)$. Let $c := (h,f)$. Return (c,u).*
- *$verify_{HMu}(c,m,u)$ with $c = (h,f)$: Check whether $f(u) = H(m)$ and $h = H(u)$. If so, return 1.*

[6] When working in the random oracle model: Quantum-polynomial-time computable given access to the random oracle.

Theorem 18 (Security of Halevi-Micali [14]). *If ℓ is superlogarithmic, then the Halevi-Micali commitment and the bounded-length Halevi-Micali commitment are statistically hiding. If H is collision-resistant, then the Halevi-Micali commitment and the bounded-length Halevi-Micali commitment are* classical-style *binding.*

Note that [14] did not prove the classical-style binding property against *quantum* adversaries. But the (very simple) proof of binding carries over unchanged to the quantum setting (if H is collision-resistant against quantum adversaries). The statistical hiding property holds against unlimited adversaries anyway, thus also against quantum adversaries.

The following theorem shows that collision-resistance does not seem to be enough to make the above constructions secure in the quantum setting, i.e., classical-style binding is all we get.

Theorem 19. *There is an oracle $\mathcal{O}$ relative to which there exists a collision-resistant[7] hash function H such that the canonical commitment scheme and both Halevi-Micali commitment schemes using H admit the following attack:*

There is a quantum-polynomial-time adversary $A^{\mathcal{O}}$ that outputs a commitment c, then expects a bit b, and then outputs with overwhelming probability a pair (m, u) such that $\mathit{verify}(c, m, u) = 1$ and the first bit of m is b.

Clearly, a commitment with that property should not be considered secure. This shows that collision-resistance is too weak a property for constructing commitments in the quantum setting, at least when using standard constructions.

The proof [20] uses the oracles constructed in [1]. In a nutshell, those oracles give the adversary access to sets S_y, such that the adversary can perform one single search in S_y for an element with a specific property, but cannot get two elements from the same S_y. Using a suitably constructed hash function H, finding m, u that open c corresponds to a search in S_y. Thus the adversary can use that search to break the binding property. But finding a collision in H corresponds to finding two elements from the same S_y, hence H is collision-resistant.

4 Collapsing hash functions

As seen in the previous section, for many protocols collision-resistance is not a sufficiently strong property in the quantum setting. In the following, we propose a strengthening of the collision-resistance property that seems more useful in the quantum setting, namely "collapsing" hash functions. We believe that collapsing hash functions are a natural assumption for real-life hash functions such as SHA-3 etc. This belief is supported by the fact that the random oracle is collapsing (see Section 6).

The definition of collapsing hash functions is similar to that of collapsing commitments (Definition 13).

[7] H is collision-resistant iff for any quantum-polynomial-time A, $\Pr[x \neq x' \wedge H(x) = H(x') : (x, x') \leftarrow A(1^n)]$ is negligible.

Definition 20 (Collapsing). *For a function H and algorithms A, B, consider the following games:*

$$\begin{aligned}
\mathsf{Game}_1: &\quad (S,M,c) \leftarrow A(1^\eta),\ m \leftarrow M_{comp}(M),\ b \leftarrow B(1^\eta,S,M)\\
\mathsf{Game}_2: &\quad (S,M,c) \leftarrow A(1^\eta),\qquad\qquad\quad\ b \leftarrow B(1^\eta,S,M)
\end{aligned}$$

Here S, M are quantum registers. $M_{comp}(M)$ is a measurement of M in the computational basis.

We call an adversary (A,B) valid *if $\Pr[H(m) = c] = 1$ when we run $(S,M,c) \leftarrow A(1^\eta)$ and measure M in the computational basis as m.*

A function H is collapsing *iff for any quantum-polynomial-time valid adversary (A,B), the difference $adv := \left|\Pr[b = 1 : \mathsf{Game}_1] - \Pr[b = 1 : \mathsf{Game}_2]\right|$ is negligible. (We call adv the* advantage*.)*

Notice that the definition of collapsing hash functions is inherently quantum, even though the object we consider (the hash function H) is classical. We know of no classical analogue to collapsing hash functions. However, a collapsing hash function will necessarily be collision-resistant, see Lemma 22 below.

We proceed to give a number of useful properties of collapsing hash functions.

Lemma 21. *An injective function H is collapsing with advantage 0.*
Lemma 22. *A collapsing hash function is collision resistant.*
Theorem 23. *If f and g are collapsing, so is $g \circ f$.*

5 Commitments from collapsing hash functions

In Section 3 we saw that collision-resistant hash functions are not sufficient for several standard constructions of commitment schemes. We will now show that those same constructions are secure in the quantum setting when using collapsing hash functions instead.

The following theorem allows us to extend the message space of a collapsing commitment by hashing the message with a collapsing hash function. Besides being useful in its own right, we need it in the analysis of the unbounded Halevi-Micali commitment.

Theorem 24. *Let f be a collapsing function. Let $(com, verify)$ be a collapse binding commitment scheme. Let $com_f(1^\eta, m) := com(1^\eta, f(m))$ and $verify_f(1^\eta, c, m, u) = verify(1^\eta, c, f(m), u)$. Then $(com_f, verify_f)$ is a collapse-binding commitment scheme.*

Lemma 25. *If H is collapsing, then the canonical commitment scheme $(com_{can}, verify_{can})$, and the bounded-length Halevi-Micali commitment $(com_{HMb}, verify_{HMb})$, and the unbounded Halevi-Micali commitment $(com_{HMu}, verify_{HMu})$ are collapse-binding. (For any choice of the parameters ℓ_u, ℓ, n.)*

We give the proof idea, the full proof is given in [20]. To show that the canonical commitment com_{can} is collapse-binding, we use the characterization of collapse-binding from Definition 13. We need to show that the adversary cannot distinguish between a measurement on register M and no measurement on register M, assuming the adversary outputs M, U containing a superposition of m, u with $verify_{can}(c, m, u) = 1$. The condition $verify_{can}(c, m, u) = 1$ is equivalent to $H(m\|u) = c$. Hence the adversary outputs in M, U a superposition of preimages of c under H. Since H is collapsing, this implies that the adversary cannot distinguish between a measurement on M, U and no measurement on M, U. This also implies (using some additional work) that the adversary cannot distinguish between a measurement on M and no measurement on M. Hence com_{can} is collapse-binding. The Halevi-Micali commitments are handled similarly.

6 Random oracles are collapsing

In Section 5 we saw that collapsing hash functions imply collapse-binding commitments. In this section, we explore the existence of collapsing hash functions. Specifically, we show that the random oracle is collapsing. This implies that there are simple collapse-binding commitments in the random oracle model. Furthermore, it supports the assumption that real-life hash functions such as SHA-3 etc. could be collapse-binding. Alternatively, we could also directly start with the assumption that SHA-3 is collapsing, in that setting the constructions from Section 5 would not need the random oracle. (In fact, we advocate that a hash function that is not collapsing should not be considered a secure practical hash function, and not recommended for future use.)

For the remainder of this section, X and Y are sets, and $H : X \to Y$ is a random oracle. Furthermore Y is finite, and $X \subseteq \{0,1\}^*$ (finite or infinite). And $q \geq 1$ always refers to an upper bound on the number of oracle queries performed by the adversary. The full proofs are given in [20].

We start by defining a seemingly unrelated property (half-collision resistance) that will turn out to imply the collapsing property. We will need half-collision resistance in our proof that the random oracle is collapsing. However, the concept of half-collision resistance might be of use for constructions in the standard model, too: since half-collision resistance is defined by a classical game, it might be easier to construct hash functions that are half-collision resistant.[8]

Definition 26. *A* half-collision *of a hash function $f : X \to Y$ is a value x such that $\exists x' \neq x. f(x) = f(x')$.*

An adversary A has advantage ε against half-collision resistance *iff*

[8] However, half-collision resistance is strictly stronger than collapsing, at least relative to an oracle, as we show next. Consider an oracle $\mathcal{O}$ picked according to the following distribution: Let $P_0, P_1 : \{0,1\}^n \to \{0,1\}^n$ be random permutations. Let $\mathcal{O}(b\|x) := P_b(x)$ for $b \in \{0,1\}, x \in \{0,1\}^n$. Then every input to $\mathcal{O}$ is a half-collision, thus $\mathcal{O}$ cannot be half-collision resistant. However P_0 and P_1 are indistinguishable from a random function [24], hence $\mathcal{O}$ is indistinguishable from $\mathcal{O}'(b\|x) := H_b(x)$ for random functions H_0, H_1. Note that $\mathcal{O}'$ is a random function, hence $\mathcal{O}'$ is collapsing by Theorem 31. Since $\mathcal{O}$ and $\mathcal{O}'$ are indistinguishable, $\mathcal{O}$ is collapsing as well.

– *with probability* 1*, the output of* A *is a half-collision or* $\perp$*, and*
– *with probability at* ε*,* A *outputs a half-collision.*

Lemma 27. *If* (A, B) *is valid and has advantage* μ *against the collapsing property of a hash function* f*, then there is an adversary* D *with advantage* $\geq \mu^2/4$ *against the half-collision resistance of* f*. The time-complexity of* D *is linear in that of* (A, B)*. (If* f *is given as an oracle,* D *makes* $4q + 4$ *queries to* f *when* (A, B) *makes* q *queries.)*

Proof sketch: By definition, a valid adversary A will always output in register M a superposition of messages m with $H(m) = c$ (all with the same c). So we have two cases: M contains a superposition of a single message m, or M contains a superposition of several messages that have the same image c, i.e., a superposition of half-collisions. Thus, in the second case, we can find a half-collisions by measuring M. But, an adversary against half-collision resistance must never output a non-half-collision (no false positives). Thus, we need a possibility to test whether M contains only a single message. (In this case, we abort.)

Note that when M contains only a single message, then the adversary B cannot distinguish between a measurement on M and no measurement on M. To exploit this, we run an execution where M is measured and an execution where M is not measured in superposition (roughly speaking), and we make it depend on a control qubit in state $|+\rangle$ which execution is used. Then, in the case where M contains only a single message, the control qubit stays unentangled with the rest of the circuit. By measuring whether the qubit is still in state $|+\rangle$, the half-collision resistance adversary can detect whether M contains one or several messages. (It may err and incorrectly assume that M contains only one message, but an error in that direction is permitted.) Thus we have constructed an adversary against half-collision resistance.

Lemma 28. *Assume* $|X| \leq |Y|$*. Then* H *is collapsing with advantage* $O(q^3/|Y|)$*.*

Proof sketch. Zhandry [24] shows that for $|X| \leq |Y|$, H can be distinguished from a random injection with probability at most $O(q^3/|Y|)$. An injection is collapsing with advantage 0 (Lemma 21).

For the next lemma, we fix some notation first: $[N] := \{1, \ldots, N\}$. For functions $f : [M] \to [N]$ and $g : [M'] \to [N]$, let $f + g : [M + M'] \to [N]$ be defined via $(f + g)(x) := f(x)$ for $x = 1, \ldots, M$ and $(f + g)(x) = g(x - M)$ for $x = M + 1, \ldots, M + M'$. For functions $f : [M] \to [N]$ and $g : [M'] \to [N']$, let $f|g : [M + M'] \to [N + N']$ be defined via $(f|g)(x) := f(x)$ for $x = 1, \ldots, M$ and $(f|g)(x) := g(x - M) + N$ for $x = M + 1, \ldots, M + M'$.

Lemma 29. *Assume that* $M \geq N$*. Let* $\hat{f}, \hat{g} : [N] \to [N]$ *and* $\hat{h} : [M] \to [M]$ *and* $\hat{\varphi} : [N + M] \to [N + M]$ *be uniformly distributed permutations (all independent), and let* $H : [2N + M] \to [N + M]$ *be a uniformly distributed function.*

Then for any q*-query adversary* A*,*

$$\left|\Pr[A^H = 1] - \Pr[A^{\hat{\varphi}\circ((\hat{f}+\hat{g})|\hat{h})} = 1]\right| \in O(q^3/N).$$

Proof sketch: We show this by rewriting $\hat{\varphi} \circ \big((\hat{f} + \hat{g})|\hat{h}\big)$ step by step, till it becomes H. In each step, the adversary distinguishes with probability $O(q^3/N)$ (denoted $\approx$ below) or 0 (denoted $\equiv$ below). For this we introduce additional functions $\varphi, v, w, \hat{v}, \hat{a}, \hat{b}, \hat{c}$ of suitable domains/ranges, all independent and uniformly random. The functions with a hat are injections. We compute:

$$\begin{aligned}\hat{\varphi} \circ ((\hat{f} + \hat{g})|\hat{h}) &\approx \varphi \circ ((\hat{f} + \hat{g})|\hat{h}) \equiv (v \circ (\hat{f} + \hat{g})) + (w \circ \hat{h}) \equiv (v \circ (\hat{f} + \hat{g})) + w \\ &\approx (\hat{v} \circ (\hat{f} + \hat{g})) + w \equiv (\hat{c} \circ \hat{a} \circ (\hat{f} + \hat{g})) + w \approx (\hat{c} \circ \hat{b}) + w \\ &\equiv \hat{c} + w \approx c + w \equiv H.\end{aligned}$$

Most of these equivalences either have elementary proofs, or are reduced to the fact that a random function and a random injection are indistinguishable. We get $H \approx \hat{\varphi} \circ \big((\hat{f} + \hat{g})|\hat{h}\big)$ which is the claim of the lemma.

Lemma 30. *Assume that $|Y| = \lceil \frac{2}{3}|X| \rceil$. Then H is collapsing with advantage $O(\sqrt{q^3/|X|})$.*

Proof sketch: For simplicity, we consider the case $|Y| = 2N$, $|X| = 3N$. Then, by Lemma 29 with $M := N$, H is indistinguishable from $H^* := \hat{\varphi} \circ \big((\hat{f} + \hat{g})|\hat{h}\big)$. Furthermore, for a random permutation π, H and $H \circ \pi$ are identically distributed, and $H \circ \pi$ is indistinguishable from $H^* \circ \pi$. Thus it is sufficient to show that $H^* \circ \pi$ is collapsing. In turn, by Lemma 27, it is sufficient to show that $H^* \circ \pi$ is half-collision resistant. To show that, observe that the half-collisions of H^* are the inputs $1, \ldots, 2N$, but not $2N + 1, \ldots, 3N$. Thus the half-collisions of $H^* \circ \pi$ are $P := \pi^{-1}(\{1, \ldots, 2N\})$. So, the half-collision resistance adversary has to find elements of P, without false positives, while given oracle access to $H^* \circ \pi$. But $H^* \circ \pi$ is indistinguishable from $H \circ \pi$, so the adversary would also be able to find elements in P given $H \circ \pi$. Since $H \circ \pi$ is a random function, independent of P, the adversary cannot do that without getting false positives. Hence $H^* \circ \pi$ is half-collision resistant and thus collapsing. Hence H is collapsing.

Theorem 31. *Let Y be finite, and $X \subseteq \{0,1\}^*$ (finite or infinite). Then $H : X \to Y$ is collapsing with advantage $O(\sqrt{q^3/|Y|})$.*

Proof sketch: H is indistinguishable from a composition $f_n \circ \cdots \circ f_1$ of random functions $f_n : X_n \to Y_n$ with $|X_{n+1}| = |Y_n| = \frac{2}{3}|X_n|$. By Lemma 30, each f_n is collapsing. Thus, by Theorem 23, $f_n \circ \cdots \circ f_1$ is collapsing and hence H is collapsing.

7 Zero-knowledge arguments of knowledge

In this section, we study the security of sigma-protocols. A sigma-protocol is a specific kind three-round proof system in which the verifier's message consists only of random bits. Sigma-protocols play an important role in classical constructions of zero-knowledge proof systems for two reasons: For a number of simple but important languages, sigma-protocols exist. And given sigma-protocols for simple

languages, there are efficient constructions for more complex languages. (There are constructions for conjunctions and disjunctions of sigma-protocols, as well as more complex threshold constructions [5].)

In the classical setting, it is relatively simple to give conditions under which sigma-protocols are zero-knowledge proofs of knowledge. In the quantum setting, however, analyzing the security of sigma-protocols turns out to be much harder. Watrous [23] presented a rewinding technique for proving the zero-knowledge property of sigma-protocols (see also Theorem 34 below). Unruh [19] showed that sigma-protocols are quantum proofs of knowledge under a specific additional condition called "strict soundness". This condition requires that the third message ("response") in a valid interaction is uniquely determined by the first two. However, strict soundness is a strong additional assumption. [19] showed how to achieve strict soundness by committing to the response already in the first message. However, the commitment scheme used for this needed to be perfectly-binding (actually, it needed to satisfy a somewhat stronger property, called "strict binding"). In particular, this implies that the commitment scheme cannot be information-theoretically hiding (hence the resulting protocol cannot be statistical zero-knowledge), and we cannot have short commitments (a perfectly-binding commitment will always be at least as long as the message inside).

Furthermore, Ambainis, Rosmanis, and Unruh [1] showed that the condition of strict soundness is necessary, at least relative to an oracle. They also showed that even if we assume that strict soundness holds, but only against computationally limited adversaries,[9] the resulting sigma-protocol will, in general, not be a quantum argument of knowledge.[10] Even more, it might not even be a quantum argument. That is, a computationally limited adversary can successfully prove a wrong statement.

In this section we show how we can use collapse-binding commitments as a drop-in replacement for the perfectly-binding commitments in the construction from [19]. One particular consequence is that given collapse-binding hash functions we can construct three-round statistical zero-knowledge quantum arguments of knowledge from sigma-protocols (without using a common-reference string). This assumes the sigma-protocol is statistical honest-verifier zero-knowledge and has special soundness. And that the challenge space (the set from which the verifier picks his random message) is polynomially-bounded. These properties, however, are also needed in the classical setting.

7.1 Interactive proof systems

An interactive proof system (P, V) for some relation R consists of two interactive quantum machines P and V that get classical inputs $(x, w) \in R$ and x, respectively. Afterwards, V outputs a bit. For formal definitions see [19]. (In general, P and V

[9] I.e., it is hard to find two different valid interactions where the first two messages are equal but the response is different.

[10] Argument and argument of knowledge are the variants of proof and proof of knowledge that consider a computationally limited malicious prover.

can exchange quantum messages, but our concrete constructions below will be classical.)

We consider two important properties of interactive proof systems: First, we want them to be arguments of knowledge. Informally, they should convince the verifier that the prover knows a witness w for the statement x (i.e., $(x,w) \in R$). Second, we want them to be zero-knowledge. Informally, the proof should not leaks anything about the witness besides its existence.

Quantum arguments of knowledge. The following definition of quantum arguments of knowledge follows the definition from [22], with one difference: we have formulated security against uniform malicious provers. That is, while in [22] the statement x and the auxiliary input $|\Psi\rangle$ are all-quantified, in our setting they are chosen by an quantum-polynomial-time algorithm Z. The reason we consider only uniform malicious provers here is: A non-uniform adversary can break any non-interactive commitment (with classical messages) that is not already perfectly-binding. (Namely, the auxiliary input can simply contain one commitment and two different openings.) Thus, since we consider only non-interactive commitments in this paper, we need a uniform definition of quantum arguments of knowledge. For a motivation of the remaining definitional choices, see [22].

Definition 32 (Quantum Arguments of Knowledge). *We call an interactive proof system (P,V) for a relation R* (uniformly) quantum-computationally extractable with knowledge error κ *if there exists a constant $d > 0$, a polynomially-bounded function $p > 0$, and a quantum-polynomial-time oracle algorithm K such that for any unitary quantum-polynomial-time algorithm P^*, for any polynomial ℓ, and for any quantum-polynomial-time algorithm Z (input generator), there exists a negligible μ such that for any security parameter $\eta \in \mathbb{N}$, we have that*

$$\begin{aligned}&\Pr[\langle \mathsf{P}^*(1^\eta,x,Z),\mathsf{V}(1^\eta,x)\rangle = 1 : (x,Z) \leftarrow \mathsf{Z}(1^\eta)] \geq \kappa(\eta) \implies \\ &\qquad \Pr[(x,w)\in R : (x,Z)\leftarrow \mathsf{Z}(1^\eta), w \leftarrow \mathsf{K}^{\mathsf{P}^*(1^\eta,x,Z)}(1^\eta,x)] \\ &\geq \tfrac{1}{p(\eta)}\Big(\Pr\big[\langle \mathsf{P}^*(1^\eta,x,Z),\mathsf{V}(1^\eta,x)\rangle = 1 : (x,Z)\leftarrow \mathsf{Z}(1^\eta)\big] - \kappa(\eta)\Big)^d - \mu(\eta).\end{aligned}$$

Here $\langle \mathsf{P}^*(1^\eta,x,Z),\mathsf{V}(1^\eta,x)\rangle$ is the output of V after an interaction between P^* and V on the respective inputs x and Z. Z is a quantum register, x is classical, both initialized using the algorithm Z. And $\mathsf{K}^{\mathsf{P}^*(1^\eta,x,Z)}$ refers to an execution of K with black-box access to $\mathsf{P}^*(1^\eta,x,Z)$. That is, K can apply the unitary U_x describing the prover P^* and its inverse $U_x^\dagger$. (See [19] for a more detailed description of that black-box execution model.)

Quantum zero-knowledge. Roughly speaking, (P,V) is *quantum-computationally zero-knowledge* iff for any quantum-polynomial-time malicious verifier V^*, there exists a quantum-polynomial-time simulator S such that for any $(x,w) \in R$, the output state of S is quantum computationally indistinguishable from the from the output state of V^* in an interaction with $\mathsf{P}(1^\eta,x,w)$.

Similarly, *quantum statistical zero-knowledge* is defined in the same way, except that V^* is not required to be quantum-polynomial-time.

We will not use the definition of quantum zero-knowledge directly, only the imported Theorem 34 from [22] will refer to it. We therefore omit the formal definition and refer to [22].

7.2 Sigma-protocols

We now introduce sigma-protocols (following [21] with modifications as mentioned in the footnotes). The notions are like the standard classical definitions, all that was done to adopt them to the quantum setting was to make the adversary quantum-polynomial-time.

A *sigma-protocol* for a relation R is a three-message proof system. It is described by its challenge space N_z (where $|N_z| \geq 2$), a classical-polynomial-time prover (P_1, P_2) and a deterministic classical-polynomial-time verifier V. The first message from the prover is $a \leftarrow P_1(1^\eta, x, w)$ and is called the *commitment*, the uniformly random reply from the verifier is $z \xleftarrow{\$} N_z$ (called *challenge*), and the prover answers with $r \leftarrow P_2(1^\eta, x, w, z)$ (the *response*). We assume P_1, P_2 to share state. Finally $V(1^\eta, x, a, z, r)$ outputs whether the verifier accepts.

Definition 33 (Computational special soundness). *There is a quantum-polynomial-time algorithm E_Σ (the extractor)*[11] *such that for any quantum-polynomial-time A, we have that*

$$\Pr[(x,w) \notin R \wedge z \neq z' \wedge ok = ok' = 1 : (x, a, z, r, z', r') \leftarrow A(1^\eta),$$
$$ok \leftarrow V(1^\eta, x, a, z, r),\ ok' \leftarrow V(1^\eta, x, a, z', r'),\ w \leftarrow E_\Sigma(1^\eta, x, a, z, r, z', r')]$$

is negligible.

Note that the above is a standard condition expected from sigma-protocols in the classical setting. In contrast, for a sigma-protocol to be a *quantum* proof of knowledge, a much more restrictive condition is required, strict soundness [19,1]. We show below how to circumvent this necessity by adding collapse-binding commitments to the sigma-protocol (at least when we only need a quantum *argument* of knowledge).

We also use the standard properties of honest verifier zero-knowledge (HVZK) and statistical honest-verifier zero-knowledge (SHVZK). They are of secondary importance for the proofs shown in this section, we defer them to [20].

Remark 1. Any sigma-protocol (N_z, P_1, P_2, V) can be seen as an interactive proof (P, V) in a natural way: P sends the output a of P_1 to V. V picks $z \xleftarrow{\$} N_z$ and sends it to P. P sends the resulting output r of P_2 to V. V checks the triple (a, z, r) using V.

The following theorem is shown in [22]:

[11] [21] requires a classical E_Σ here. By allowing E_Σ to be quantum here, we weaken the notion of computational special soundness slightly, and thus strengthen our results below.

Theorem 34 (HVZK implies zero-knowledge [22]). *Let* $\Sigma = (N_z, P_1, P_2, V)$ *be a sigma-protocol. We consider* Σ *as an interactive proof* (P, V)*, see Remark 1.*

If $|N_z|$ *is polynomially-bounded and is SHVZK, then* Σ *is quantum statistical zero-knowledge. If* $|N_z|$ *is polynomially-bounded and* Σ *is HVZK, then* Σ *is quantum computational zero-knowledge.*

Due to this theorem, it will be sufficient to verify that the sigma-protocols we construct are HVZK/SHVZK. We will hence not need to use the definition of quantum zero-knowledge explicitly in the following.

7.3 Constructing zero-knowledge arguments of knowledge

In [19], the following idea was used to construct quantum proofs of knowledge: We assume a sigma-protocol with special soundness and with polynomial-size $|N_z|$. We convert it into a sigma-protocol with strict soundness as follows: When the prover sends his commitment $a \leftarrow P_1(x, w)$, he additionally sends $com(r_z)$ for all $z \in N_z$ where r_z is the response to the challenge z. When the prover receives the challenge z, he opens $com(r_z)$ instead of sending r_z. If the commitment has the "strict binding" property, the resulting sigma-protocol has strict soundness (without losing the special soundness or HVZK property).[12] Strict binding is a strengthening of perfect binding, it means that not only the message in the commitment is information-theoretically determined, but also the opening information.

Given a sigma-protocol with strict and special soundness, we can show that it is a proof of knowledge. Basically, [19] runs the protocol twice (using the inverse of the unitary malicious prover to rewind) to get two responses r, r' for different challenges $z \neq z'$. The difficulty here is that measuring r can disturb the state of the malicious prover, leading to a corrupt value r'. The trick here is that due to the strict soundness, the value r is essentially uniquely determined, and therefore the measurement does not introduce too much disturbance.[13]

Unfortunately, that technique needs commitments with the strict binding property. First, it is easy to see that strict binding commitments must be longer than the messages they contain. Short strict binding commitments are not possible. Furthermore, the only known construction of strict binding commitments [19] uses quantum 1-1 one-way functions. No candidates for those are known.

We show below that the same technique of committing to the responses works with collapse-binding commitments. The crucial point in the analysis from [19] was that measuring the committed response does not change the state. The collapse-binding property guarantees something slightly weaker: when measuring the committed response, the state may change, but this cannot be noticed by a computationally limited adversary. So with collapse-binding commitments,

[12] This part was done only implicitly in [19], in the analysis of the Hamiltonian cycle proof system.

[13] There is some disturbance due to the fact that it is not determined whether r is a valid response or an invalid one.

an analog reasoning as in [19] can be used, except that we get security only against quantum-polynomial-time adversaries. I.e., we get a quantum argument of knowledge. We will now describe this in more detail.

First, we formalize the sigma-protocol in which we commit to the responses:

Definition 35 (Sigma-protocol with committed responses). *Let (N_z, P_1, P_2, V) be a sigma-protocol with polynomially-bounded $|N_z|$. Let $(com, verify)$ be a commitment scheme (with the responses of (N_z, P_1, P_2, V) as message space). We construct a sigma-protocol (N_z, P_1', P_2', V') as follows:*

- *$P_1'(1^\eta, x, w)$ runs: $a \leftarrow P_1(1^\eta, x, w)$. For each $z \in N_z$: $r_z \leftarrow P_2(1^\eta, x, w, z)$* [14] *and $(c_z, u_z) \leftarrow com(1^\eta, r_z)$. Let $a' := (a, (c_z)_{z \in N_z})$ and return a'.*
- *$P_2'(1^\eta, x, w, z)$ returns $r' := (r_z, u_z)$.*
- *$V'(1^\eta, x, a', z, r')$ with $a' = (a, (c_z)_{z \in N_z})$ and $r' = (r, u)$: Check whether $verify(1^\eta, c_z, r, u) = 1$ and $V(1^\eta, a, z, r) = 1$. Iff so, return 1.*

We show that the above construction is a quantum argument of knowledge:

Theorem 36 (Quantum argument of knowledge). *If (N_z, P_1, P_2, V) is a sigma-protocol with computational special soundness for a relation R, and $(com, verify)$ is collapse-binding, then (N_z, P_1', P_2', V') from Definition 35 is computationally quantum extractable for R with knowledge error $1/\sqrt{|N_z|}$.*

The proof of this theorem will rely on the following lemma from [19]. (That lemma is the core lemma of the rewinding technique from [19].)

Lemma 37 (Extraction via quantum rewinding [19]). *Let C be a set with $|C| = c$. Let $(P_i)_{i \in C}$ be projectors. Let $|\Phi\rangle$ be a unit vector. Let $V := \sum_{i \in C} \frac{1}{c} \||P_i|\Phi\rangle\|^2$ and $E := \sum_{i,j \in C, i \neq j} \frac{1}{c^2} \|P_i P_j |\Phi\rangle\|^2$. Then, if $V \geq \frac{1}{\sqrt{c}}$, $E \geq V(V^2 - \frac{1}{c})$.*

Proof of Theorem 36. Recall that any sigma-protocol can be seen as an interactive proof system by Remark 1. Let (P, V) denote the interactive proof system resulting from the sigma-protocol (N_z, P_1', P_2', V'). (In particular, the verifier V sends a random $z \in N_z$, and in the end checks whether $verify(1^\eta, c_z, r, u) = 1$ and $V(1^\eta, a, z, r) = 1$.)

Let P^* denote a malicious prover, i.e., a unitary quantum-polynomial-time algorithm. Since P^* attacks a sigma-protocol, it sends two messages. We can thus assume that P^* is of the following form:

- It operates on quantum registers Z, C, R, U. Here Z contains the internal state of P^* (initialized by algorithm Z). C is the register that will contain the first message $a' = (a, (c_z)_z)$ sent by P^*. R, U contains the second message $r' = (r, u)$ sent by P^*. And C, R, U are initialized with $|0\rangle$.
- The unitary U_x describes the unitary operation of P^* on Z, C during the first invocation of P^*. U_x is parametrized by the classical input x of P^*. The message $a' = (a, (c_z)_z)$ is obtained by measuring C in the computational basis.

[14] We can run P_2 several times using the final state of P_1 because P_1 is classical.

– The unitary U_z describes the unitary operation of P^* on Z, R, U during the second invocation of P^*. U_z is parametrized by the challenge z that P^* receives. The message $r' = (r, u)$ is obtained by measuring R and U in the computational basis.

We fix some additional notation for this proof:

– V_z: The projector on R, U onto the span of all $|r, u\rangle$ with $\mathit{verify}(1^\eta, c_z, r, u) = 1$. (That is, V_z measures whether measuring R, U would yield a valid opening of c_z.)
– W_z: The projector on R onto the span of all $|r\rangle$ with $V(1^\eta, a, z, r) = 1$. (That is, W_z measures whether measuring R yields a valid response r for challenge z.)
– $P_z := U_z^\dagger W_z V_z U_z$. Since V_z and W_z are projectors and diagonal in the computational basis, they commute and their product is a projector. And since U_z is a unitary, P_z is a projector (acting on registers Z, R, U).
– $x \leftarrow \mathbf{M}(X)$ denotes that x is assigned the result of measuring the register X in the computational basis.
– $ok \leftarrow P(X)$ means that ok is assigned 1 iff measuring the register X with projector P succeeds. (With P being, e.g., one of V_z, W_z, P_z.)
– We write $U(X)$ or $U(X)$ to mean that the unitary U is applied to the register X. (With U being, e.g., one of U_x, U_z).

With that notation, we can rewrite the success probability of the malicious prover as follows:

$$\begin{aligned}
\Pr_V &:= \Pr[\mathsf{P}^*(1^\eta, x, Z), \mathsf{V}(1^\eta, x)\rangle = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta)] \\
&= \Pr[ok_c = ok_v = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C), \\
&\qquad z \xleftarrow{\$} N_z,\ U_z(ZRU),\ r \leftarrow \mathbf{M}(R),\ u \leftarrow \mathbf{M}(U), \\
&\qquad ok_c = \mathit{verify}(1^\eta, c_z, r, u),\ ok_v = V(1^\eta, a, z, r)] \\
&= \Pr[ok = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C),\ z \xleftarrow{\$} N_z, \\
&\qquad ok \leftarrow P_z(ZRU)].
\end{aligned}$$

We now construct the extractor $\mathsf{K}^{\mathsf{P}^*(1^\eta, x, Z)}(1^\eta, x)$ required by Definition 32. It operates on quantum registers S, C, R, U as follows:

$$\begin{gathered}
(x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C),\ z, z' \xleftarrow{\$} N_z,\ U_z(ZRU), \\
ok_c \leftarrow V_z(RU), r \leftarrow \mathbf{M}(R),\ U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ r' \leftarrow \mathbf{M}(R), \\
w \leftarrow E_\Sigma(1^\eta, x, a, z, r, z', r'),\ \text{return } w.
\end{gathered}$$

Here E_Σ is the extractor of the sigma-protocol (N_z, P_1, P_2, V). This extractor exists because the sigma-protocol has computational special soundness (see Definition 33). Note that K only uses black-box access to P (via the unitaries $U_x, U_z, U_{z'}$ and their inverses).

We will now bound the success probability of the extractor

$$\begin{aligned}
\Pr_E &:= \Pr[(x,w)\in R : w \leftarrow \mathsf{K}^{\mathsf{P}^*(1^\eta,x,Z)}(1^\eta,x)] \\
&= \Pr[(x,w)\in R : (x,Z)\leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a,(c_z)_z)\leftarrow \mathbf{M}(C),\ z,z'\xleftarrow{\$} N_z, \\
&\qquad U_z(ZRU),\ ok_c\leftarrow V_z(RU),\ r\leftarrow \mathbf{M}(R),\ U_z^\dagger(ZRU),\ U_{z'}(ZRU), \\
&\qquad r'\leftarrow \mathbf{M}(R),\ w\leftarrow E_\Sigma(1^\eta,x,a,z,r,z',r')] \\
&= \Pr[(x,w)\in R : (x,Z)\leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a,(c_z)_z)\leftarrow \mathbf{M}(C),\ z,z'\xleftarrow{\$} N_z, \\
&\qquad U_z(ZRU),\ ok_c\leftarrow V_z(RU),\ r\leftarrow \mathbf{M}(R),\ ok_v\leftarrow V(1^\eta,x,a,z,r), \\
&\qquad U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ r'\leftarrow \mathbf{M}(R),\ ok_v'\leftarrow V(1^\eta,x,a,z',r'), \\
&\qquad w\leftarrow E_\Sigma(1^\eta,x,a,z,r,z',r')].
\end{aligned}$$

Due to the computational special soundness of (N_z,P_1,P_2,V), in the previous game, with overwhelming probability, $z\neq z'$ and $ok_v=1$ and $ok_{v'}=1$ implies $(x,w)\in R$. Thus there exists a negligible μ_1 such that

$$\begin{aligned}
\Pr_E \geq \Pr[&z\neq z'\wedge ok_v=ok_v'=1 : (x,Z)\leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a,(c_z)_z)\leftarrow \mathbf{M}(C), \\
&z,z'\xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c\leftarrow V_z(RU),\ r\leftarrow \mathbf{M}(R), \\
&ok_v\leftarrow V(1^\eta,x,a,z,r),\ U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ r'\leftarrow \mathbf{M}(R), \\
&ok_v'\leftarrow V(1^\eta,x,a,z',r')] - \mu_1 =: \Pr_E' - \mu_1.
\end{aligned}$$

Instead of computing $ok_v\leftarrow V(1^\eta,x,a,z,r)$ using the just measured r, we can instead measure whether the register R contains a value r that would make $V(1^\eta,x,a,z,r)=1$ true. I.e., we can replace $ok_v\leftarrow V(1^\eta,x,a,z,r)$ by a measurement using the projector W_z. Since at that point, R was just measured in the computational basis, the measurement using W_z does not disturb the state of the system. Similarly, we can replace $ok_v'\leftarrow V(1^\eta,x,a,z',r')$ by a measurement using $W_{z'}$. We get:

$$\begin{aligned}
\Pr_E' &= \Pr[z\neq z'\wedge ok_v=ok_v'=1 : (x,Z)\leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a,(c_z)_z)\leftarrow \mathbf{M}(C), \\
&\qquad z,z'\xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c\leftarrow V_z(RU),\ r\leftarrow \mathbf{M}(R),\ ok_v\leftarrow W_z(R), \\
&\qquad U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ r'\leftarrow \mathbf{M}(R),\ ok_v'\leftarrow W_{z'}(R)] \\
&= \Pr[z\neq z'\wedge ok_v=ok_v'=1 : (x,Z)\leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a,(c_z)_z)\leftarrow \mathbf{M}(C), \\
&\qquad z,z'\xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c\leftarrow V_z(RU),\ r\leftarrow \mathbf{M}_{ok_c}(R),\ ok_v\leftarrow W_z(R), \\
&\qquad U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ r'\leftarrow \mathbf{M}(R),\ ok_v'\leftarrow W_{z'}(R)].
\end{aligned}$$

In the last probability, $r\leftarrow \mathbf{M}_{ok_c}(R)$ refers to a measurement on R that is only executed if $ok_c=1$. (And $r:=\bot$ otherwise.) The last two probabilities are equal because $\mathbf{M}(R)$ and $\mathbf{M}_{ok_c}(R)$ only differ if $ok_c=0$, in which case "$z\neq z'\wedge ok_v=ok_v'=1$" is false anyway.

Since V_z measures whether R,U contains $|r,u\rangle$ with $\mathit{verify}(1^\eta,c_z,r,u)=1$, and since $(\mathit{com},\mathit{verify})$ is collapse-binding, and since the outcome r is never used,

we have that no quantum-polynomial-time adversary can distinguish between "$ok_c \leftarrow V_z(RU),\ r \leftarrow \mathbf{M}(R)$" and "$ok_c \leftarrow V_z(RU)$", except with negligible probability. (Cf. Definition 12.) Thus there is a negligible μ_2 such that

$$\begin{aligned}\Pr'_E \geq \Pr[&z \neq z' \wedge ok_v = ok'_v = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C),\\ &z, z' \xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c \leftarrow V_z(RU),\ ok_v \leftarrow W_z(R),\ U_z^\dagger(ZRU),\\ &U_{z'}(ZRU),\ r' \leftarrow \mathbf{M}(R),\ ok'_v \leftarrow W_{z'}(R)] - \mu_2 =: \Pr''_E - \mu_2.\end{aligned}$$

Since $\mathbf{M}(R)$ and $W_{z'}(R)$ and $V_{z'}(RU)$ commute, and since adding additional/removing operations after all values z, z', ok_v, ok'_v are fixed does not change the distribution of those values, we have that "$r' \leftarrow \mathbf{M}(R), ok'_v \leftarrow W_{z'}(R)$" and "$ok'_c \leftarrow V_{z'}(RU), ok'_v \leftarrow W_z(R), U_{z'}^\dagger(ZRU)$" lead to the same distribution of z, z', ok_v, ok'_v. This justifies $(*)$ in the following calculation:

$$\begin{aligned}\Pr''_E &\overset{(*)}{=} \Pr[z \neq z' \wedge ok_v = ok'_v = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C),\\ &\qquad z, z' \xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c \leftarrow V_z(RU),\ ok_v \leftarrow W_z(R),\ U_z^\dagger(ZRU),\\ &\qquad U_{z'}(ZRU),\ ok'_c \leftarrow V_{z'}(RU),\ ok'_v \leftarrow W_{z'}(R),\ U_{z'}^\dagger(ZRU)]\\ &\geq \Pr[z \neq z' \wedge ok_c = ok_v = 1 \wedge ok'_c = ok'_v = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\\ &\qquad (a, (c_z)_z) \leftarrow \mathbf{M}(C),\ z, z' \xleftarrow{\$} N_z,\ U_z(ZRU),\ ok_c \leftarrow V_z(RU),\ ok_v \leftarrow W_z(R),\\ &\qquad U_z^\dagger(ZRU),\ U_{z'}(ZRU),\ ok'_c \leftarrow V_{z'}(RU),\ ok'_v \leftarrow W_{z'}(R),\ U_{z'}^\dagger(ZRU)]\\ &= \Pr[z \neq z' \wedge ok = 1 \wedge ok = 1 : (x, Z) \leftarrow \mathsf{Z}(1^\eta),\ U_x(ZC),\ (a, (c_z)_z) \leftarrow \mathbf{M}(C),\\ &\qquad z, z' \xleftarrow{\$} N_z,\ ok \leftarrow P_z(ZRU),\ ok \leftarrow P_{z'}(ZRU)].\end{aligned}$$

Let $\alpha_{a'} := \Pr[a' = (a, (c_z)_z)]$ in the previous game, and let $|\psi_{a'}\rangle$ denote the post-measurement-state of registers Z, R, U after the measurement $(a, (c_z)_z) \leftarrow \mathbf{M}(C)$. Then

$$\Pr''_E = \sum_{a'} \alpha_{a'} \underbrace{\sum_{\substack{z,z'\\ z\neq z'}} \frac{1}{|N_z|^2} \Big\| P_{z'} P_z |\psi_{a'}\rangle \Big\|^2}_{=:E_{a'}}.$$

Furthermore, note that

$$\Pr_V = \sum_{a'} \alpha_{a'} \underbrace{\sum_{z} \frac{1}{|N_z|} \Big\| P_z |\psi_{a'}\rangle \Big\|^2}_{=:V_{a'}}.$$

Lemma 37 implies that if $V_{a'} \geq 1/\sqrt{|N_z|}$, then $E_{a'} \geq V_{a'}(V_{a'}^2 - 1/|N_z|)$. Or stated differently: $E_{a'} \geq \varphi(V_{a'})$ where $\varphi(x) := 0$ for $x < 1/\sqrt{|N_z|}$ and $\varphi(x) := x(x^2 - 1/|N_z|)$ for $x \geq 1/\sqrt{|N_z|}$. Since φ is convex on $[0, 1]$, by Jensen's inequality we get $\Pr''_E \geq \varphi(\Pr_V)$. In other words $\Pr''_E \geq \Pr_V(\Pr_V^2 - 1/|N_z|)$ whenever $\Pr_V \geq 1/\sqrt{|N_z|}$. Furthermore, the inequalities derived above give $\Pr_E \geq \Pr''_E - \mu$ for $\mu := \mu_1 + \mu_2$. And μ is negligible. It follows that:

$$\Pr_V \geq \frac{1}{\sqrt{N_z}} \qquad \Longrightarrow \qquad \Pr_E \geq \Pr_V\Big(\Pr_V^2 - \frac{1}{|N_z|}\Big) - \mu \geq \Big(\Pr_V - \frac{1}{\sqrt{|N_z|}}\Big)^3 - \mu.$$

Thus (P, V) is quantum-computationally extractable for R with knowledge error $\kappa := 1/\sqrt{|N_z|}$. □

In [20], we additionally show that the resulting protocol is also zero-knowledge. (This only uses the hiding property, and is hence independent of our new definitions.)

Theorem 38 (Zero-knowledge). *If $|N_z|$ is polynomially-bounded, and (N_z, P_1, P_2, V) is HVZK and $(com, verify)$ is computationally hiding, and com is a polynomial-time algorithm, then (N_z, P_1', P_2', V') is computational zero-knowledge.*

If $|N_z|$ is polynomially-bounded, and (N_z, P_1, P_2, V) is SHVZK and $(com, verify)$ is statistically hiding, and com is a polynomial-time algorithm, then (N_z, P_1', P_2', V') is statistical zero-knowledge.

8 Open problems

We list some questions for future research:

- We have constructed quantum arguments of knowledge from sigma-protocols by using collapse-binding commitments. However, our construction requires the challenge space N_z of the sigma-protocol to be of polynomially-bounded size. As a consequence, the resulting argument of knowledge will have a noticeable knowledge error; for a negligible knowledge error we need to use sequential repetition, resulting in a proof system with non-constant round complexity. Are there general constructions of arguments of knowledge from sigma-protocols that do not require the challenge space to be polynomially-bounded?
- Can we use collapse-binding commitments to construct a quantum OT protocol? For example, using the construction from [2] or a variation thereof?
- How are the various definitions of computationally binding commitments related? That is, which implications and separations exist between sum-binding, CDMS-binding, collapse-binding, and UC-secure commitments?

Acknowledgements. We thank Ansis Rosmanis for discussions on insecure commitments based on collision-resistant hash functions, and Serge Fehr for discussions on the DFRSS-binding definition. This research by the European Social Fund's Doctoral Studies and Internationalisation Programme DoRa, by the European Regional Development Fund through the Estonian Center of Excellence in Computer Science, EXCS, by European Social Fund through the Estonian Doctoral School in Information and Communication Technology, and by the Estonian ICT program 2011-2015 (3.2.1201.13-0022).

References

1. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems (the hardness of quantum rewinding). In: FOCS 2014. pp. 474–483. IEEE (2014)
2. Bennett, C.H., Brassard, G., Crépeau, C., Skubiszewska, M.H.: Practical quantum oblivious transfer. In: Crypto '91. LNCS, vol. 576, pp. 351–366. Springer (1991)

3. Boneh, D., Dagdelen, O., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Asiacrypt 2011. pp. 41–69. Springer (2011)
4. Brassard, G., Crépeau, C., Jozsa, R., Langlois, D.: A quantum bit commitment scheme provably unbreakable by both parties. In: FOCS '93. pp. 362–371. IEEE (1993)
5. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Crypto 94. LNCS, vol. 839, pp. 174–187. Springer (1994)
6. Crépeau, C., Dumais, P., Mayers, D., Salvail, L.: Computational collapse of quantum state with application to oblivious transfer. In: TCC 2004. LNCS, vol. 2951, pp. 374–393. Springer (2004)
7. Crépeau, C., Légaré, F., Salvail, L.: How to convert the flavor of a quantum bit commitment. In: Eurocrypt 2001. LNCS, vol. 2045, pp. 60–77. Springer (2001)
8. Crépeau, C., Salvail, L., Simard, J.R., Tapp, A.: Two provers in isolation. In: Asiacrypt 2011. LNCS, vol. 7072, pp. 407–430. Springer (2011)
9. Damgård, I., Fehr, S., Lunemann, C., Salvail, L., Schaffner, C.: Improving the security of quantum protocols via commit-and-open. In: Crypto 2009. LNCS, vol. 5677, pp. 408–427. Springer (2009)
10. Damgård, I., Fehr, S., Renner, R., Salvail, L., Schaffner, C.: A tight high-order entropic quantum uncertainty relation with applications. In: Crypto 2007. LNCS, vol. 4622, pp. 360–378. Springer (2007)
11. Damgård, I., Fehr, S., Salvail, L.: Zero-knowledge proofs and string commitments withstanding quantum attacks. In: Crypto 2004. pp. 254–272. LNCS, Springer (3152)
12. Damgård, I., Lunemann, C.: Quantum-secure coin-flipping and applications. In: Asiacrypt 2009. vol. 5912, pp. 52–69. Springer (2009)
13. Dumais, P., Mayers, D., Salvail, L.: Perfectly concealing quantum bit commitment from any quantum one-way permutation. In: Eurocrypt 2000. LNCS, vol. 1807, pp. 300–315. Springer (2000)
14. Halevi, S., Micali, S.: Practical and provably-secure commitment schemes from collision-free hashing. In: Crypto '96. LNCS, vol. 1109, pp. 201–215. Springer (1996)
15. Mayers, D.: Unconditionally Secure Quantum Bit Commitment is Impossible. PRL 78(17), 3414–3417 (1997)
16. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge, 10th anniv. edn. (2010)
17. NIST: SHA-3 standard: Permutation-based hash and extendable-output functions. Draft FIPS 202 (2014)
18. Unruh, D.: Universally composable quantum multi-party computation. In: Eurocrypt 2010. LNCS, vol. 6110, pp. 486–505. Springer (2010)
19. Unruh, D.: Quantum proofs of knowledge. In: Eurocrypt 2012. LNCS, vol. 7237, pp. 135–152. Springer (2012)
20. Unruh, D.: Computationally binding quantum commitments. IACR ePrint 2015/361 (2015), full version of this paper
21. Unruh, D.: Non-interactive zero-knowledge proofs in the quantum random oracle model. In: Eurocrypt 2015. vol. 9057, pp. 755–784 (2015)
22. Unruh, D.: Quantum proofs of knowledge. IACR ePrint 2010/212/20150211:174234 (2015), updated full version of [19]
23. Watrous, J.: Zero-knowledge against quantum attacks. SIAM J. Comput. 39(1), 25–58 (2009)
24. Zhandry, M.: A note on the quantum collision and set equality problems. Quantum Information & Computation 15(7&8), 557–567 (2015)