Private proofs of when and where

Uma Girish
Columbia

Greg Gluch
UC Berkeley

Shafi Goldwasser
UC Berkeley

Tal Malkin
Columbia

Leo Orshansky
Columbia

Henry Yuen
Columbia

Abstract

Position verification schemes are interactive protocols where entities prove their physical location to others; this enables interactive proofs for statements of the form "I am at a location L." Although secure position verification cannot be achieved with classical protocols (even with computational assumptions), they are feasible with *quantum* protocols.

In this paper we introduce the notion of *zero-knowledge position verification*, which extends position verification in two ways:

1. enabling entities to prove more sophisticated statements about their locations at different times (for example, proving statements about one's position in the *past* "I was *not* near location L at noon yesterday").
2. maintaining privacy for any other detail about their true location besides the statement they are proving.

We construct zero-knowledge position verification from secure position verification and post-quantum one-way functions. The central tool in our construction is a primitive we call *position commitment*, which allows entities to privately *commit* to their physical position in a particular moment, which is then revealed at some later time.

1 Introduction

Position verification protocols enable an entity (which we call the *prover*) to prove the statement "I am at location L" to a set of verifiers. These protocols are analogous to interactive proofs in complexity theory and cryptography, except the statements proven are about a *physical property* of the prover, rather than the logical properties of some mathematical object (e.g., "The graph φ is 3-colorable"). Position verification gives rise to intriguing new cryptographic possibilities, such as using someone's geospatial location as a secure credential for decrypting a message or performing a private computation [CGMO09, BCF+14].

The idea behind position verification is that a group of verifiers can exchange signals with a purported prover in a purported location, and based on the timing of the received signals, obtain trustworthy proof of the prover's location. Without strong assumptions, such proofs cannot be obtained in a purely classical setting: a number of papers [CGMO09, KMS11, BCF+14] showed that it is always possible for colluding adversaries to efficiently spoof any classical position verification protocol. This classical impossibility result holds even if we make computational assumptions; thus for position verification to be possible one has to consider different models, such as trusted

hardware models [GW98], bounded storage models [CGMO09], or the quantum model [KMS11, BCF+14, Mal16]. The quantum position verification (QPV) model is perhaps the most compelling, as it leverages physical principles such as unclonability and uncertainty to potentially achieve secure location verification.

However, even if secure position verification is achievable, it raises a fundamental privacy concern: an entity's position in space and time is one of its most sensitive physical attributes. In many applications, the goal is not to reveal a precise location, but rather to prove some statement about it. For example, one might wish to prove to a judge that one was *not* near a crime scene ("I was not at location L."), or to certify that a nation-state's armaments remained within its borders ("The weapon was always inside a region.").

In this work we ask:

Can position verification protocols be made private?

We thus ask whether there exist proofs that *don't* reveal anything additional about the locations of the involved entities (other than what is required by the application). In other words, is it possible to obtain *zero-knowledge position proofs*? This would be analogous to traditional zero-knowledge proofs in complexity theory and cryptography.

We argue that many natural settings would benefit from zero-knowledge location-proving protocols. A striking example is a recent investigation [Pre25] which revealed that presidential bodyguards inadvertently exposed the movements of heads of state by using the fitness-tracking app Strava. A protocol capable of proving statements such as "Alice completed a run with a particular *shape* within a given time," *while revealing nothing about the run's geographic location*, would directly mitigate the vulnerability highlighted in [Pre25]. Other potential applications include *privacy-preserving verification of compliance with nuclear treaties*—for instance, the Antarctic Treaty [59], which forbids the placement of nuclear weapons in Antarctica, could benefit from proofs that a given device is not present on the continent. Further examples are *private alibis* (proving that someone was not at a crime scene without revealing their actual location) and *privacy-preserving export-control enforcement* (e.g., proving that a GPU is operated within a U.S. data center).

Contributions. Our contributions are:

1. We formalize zero-knowledge position verification—more precisely, spacetime verification (see Section 1.1)—using a simulation-based security definition.

2. We construct zero-knowledge position verification protocols assuming the existence of post-quantum one-way functions and a certain class of position verification protocols, which is achieved by previous work (see Section 1.2).

3. We introduce a new primitive, *position commitment*, which allows a prover to commit to its geographic location L at time t, and to open the commitment at a later time. We give a construction of position commitment and show how it serves as a key building block for zero-knowledge position verification.

We note that, beyond providing privacy, our protocols also enable the proving of statements about one's location at an arbitrary point in the *past*. This could potentially allow, for example, someone to prove their past location to a judge.

1.1 Position verification for regions of spacetime

Before explaining zero-knowledge position verification, we first define the notion of position verification more carefully.

We model space as $\mathbb{R}^d$ for some dimension d, and time is determined by a universal clock $t \in \mathbb{R}$. We assume that all parties know the time and their positions exactly. We assume that all signals between parties travel at some fixed velocity (e.g., one unit of distance per unit of time) to model the speed of light constraint in physics. We assume that parties can choose for their communication to be either broadcast or sent directionally, i.e., each party can transmit a signal along a ray. Finally, we assume that the behavior of all parties can be described by quantum mechanics – that is, they can perform quantum computation and send quantum information to each other. (We elaborate more on the modeling in Section 2.2).

A position verification protocol is a triple $\Pi = (P, V, R)$ where $V = \{V_1, \dots, V_k\}$ is a set of *verifier* algorithms, $R \subset \mathbb{R}^d \times \mathbb{R}$ is a region of *allowable prover points* in spacetime, and $P = \{P_\alpha\}_{\alpha \in R}$ is a set of *prover* algorithms indexed by R. The protocol Π is *complete* if for all $\alpha = (L, t) \in R$, the prover P_α is positioned at L at time t, and causes V to accept with high probability. The protocol has *position security against a class $\mathscr{C}$ of spoofers* if all sets $\{P_1, \dots, P_m\}$ of *spoofers* in the class $\mathscr{C}$, none of whom have positions in the set R, are rejected with high probability. (The formal definition can be found in Definition 2.7).

The security of a position verification protocol crucially depends on a class $\mathscr{C}$ of spoofers. It is known that quantum position verification schemes cannot be secure against spoofers that use more than exponential amounts of quantum entanglement, because of the existence of a general attack [Vai03, BK11, BCF+14]. However, there are secure position verification protocols against spoofers with bounded entanglement [BCS22] or bounded query complexity in the random oracle model [Unr14, LLQ22]. For concreteness we will focus on spoofers with limited entanglement; however, our results generalize to broader classes of adversaries.

In this definition, we think of the verifiers as being convinced of the statement "There is a prover located in spacetime region R". In the literature, position verification protocols have been typically defined for proving a specific position L^* and time t^*. We call such protocols *singleton position verification protocols*, because this corresponds to a singleton set $R = \{(L^*, t^*)\}$. Allowing a larger set of positions to be proved is important for our upcoming notion of zero-knowledge position verification.

A canonical position verification protocol. The reader may find it useful to carry the following example of a quantum position verification protocol in mind. The protocol verifies that a prover is at the midpoint between two verifiers on a line; this is called the *f-BB84* protocol [BCS22]. Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a boolean function. Using shared randomness, verifiers V_1, V_2 sample uniformly random strings $x, y \in \{0,1\}^n$ and a uniformly random bit $b \in \{0,1\}$. The first verifier V_1 creates a qubit in the state $|\psi\rangle = H^{f(x,y)} |b\rangle$ (i.e., a random BB84 state in a basis determined by $f(x,y)$). The verifier V_1 sends x and $|\psi\rangle$ towards the midpoint and V_2 sends y towards the midpoint (see Figure 1 for an illustration). Each verifier expects a single bit b' back from the purported prover and accept if $b = b'$.[1] When $f(x,y)$ is the inner product function this protocol is secure against spoofers that share $O(\log n)$ qubits of entanglement [BCS22], and it is conjectured

[1] The honest prover at the midpoint who receives $x, y, |\psi\rangle$ at the same time is supposed to compute $f(x,y)$ (which we assume can be done instantaneously), and measure the qubit $|\psi\rangle$ in the appropriate basis to recover the bit b.

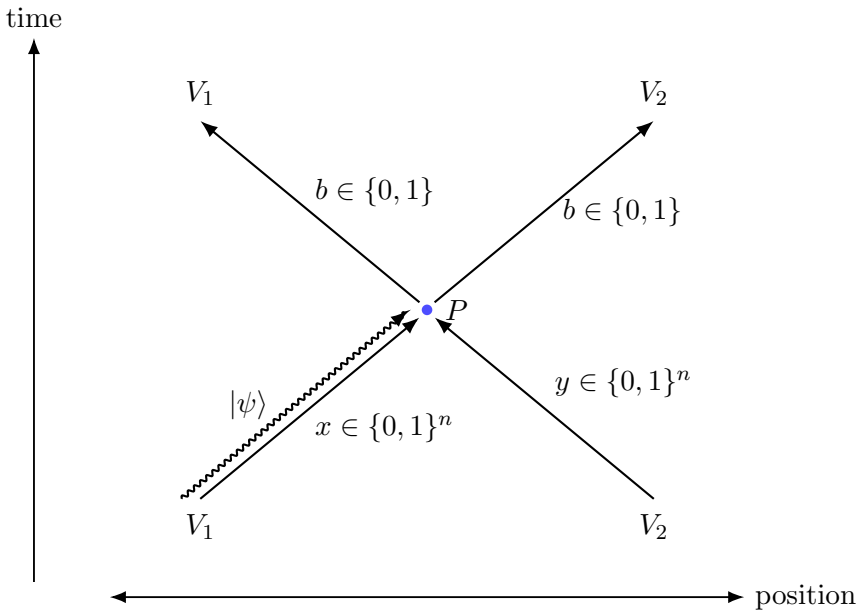

Figure 1: A spacetime diagram of the f-BB84 protocol [BCS22], specialized to 1 dimension position verification. Time goes up, position is horizontal, and signals travel along 45° angles.

that there exists a polynomial-time computable f such that f-BB84 is secure against spoofers with superpolynomial qubits of entanglement.

1.2 Zero-knowledge position verification

We now give an informal definition of zero-knowledge position verification. Intuitively, the privacy notion we want to capture is that the "view" of the verifiers when interacting with a prover P located in the allowed region R can be simulated independently of its exact position within R. Let $\Pi = (P, V, R)$ be a position verification protocol with honest prover P, honest verifiers $V = \{V_1, \ldots, V_k\}$, and a set of allowable prover spacetime points $R \subseteq \mathbb{R}^d \times \mathbb{R}$. We say that Π is *zero-knowledge* if there exists a polynomial-time *simulator* outputting a quantum state ρ such that for all allowable spacetime points $\alpha \in R$, ρ is computationally indistinguishable from the joint state ρ_α of verifiers V at any given moment during the interaction with prover P located at α.

Just like there are many variants of zero-knowledge in cryptography (statistical vs. computational, semi-honest vs. malicious verifier, etc.), there can also be many variants of zero-knowledge position verification. In this paper we study the setting of *semi-honest* verifiers, where we assume that the verifiers honestly follow the protocol Π, but may try to learn more about the prover's true location by inspecting the prover's messages. We discuss ideas and barriers towards relaxing the semi-honest verifier model in Section 5.

Next we show that zero-knowledge position verification is achievable assuming the existence of one-way functions (OWF) and secure singleton position verification protocols satisfying the following conditions:

1. The protocol is one round: the verifiers send a message to the prover, who is supposed to

respond immediately.

2. The prover's message is classical, and the verifiers' accept/reject decision is a deterministic function of the prover's response, timing, and the verifier's shared randomness.

3. The protocol can securely verify any position contained within the convex hull of the verifiers' spatial positions.

We call a singleton position verification protocol satisfying this *nice*; see Definition 2.8 for a formal definition, and an intuition for these criteria. An example of a nice singleton position verification scheme is f-BB84. Other examples include the classical-verifier protocol of [LLQ22], or the random-oracle model protocol of [Unr14].

We now can state the main result of the paper.

Theorem 1.1 (Zero-knowledge position verification, informal)**.** *Suppose there exist post-quantum one-way functions and nice singleton position verification protocols which have position security against spoofers sharing at most E entangled qubits. Then for all finite $R \subseteq \mathbb{R}^d \times \mathbb{R}$, there is an (honest-verifier) zero-knowledge position verification protocol $\Pi_{ZK} = (P, V, R)$ with position security against spoofers sharing $E/2$ entangled qubits.*

In other words, any nice secure position verification protocol Π for proving *single* spacetime point can be "upgraded" (assuming OWF) to a *zero-knowledge* position verification protocol Π_{ZK} for any *finite set* of spacetime positions.

A formal version of this main result is stated and proven in Theorem 4.3.

Remark 1.2. As a default choice we assume the quantum model of position verification, where the best known protocols are secure against adversaries with unlimited computational power, but with a bounded amount of shared entanglement. However, since the position security of the zero-knowledge location proofs construction is directly inherited from the underlying singleton position verification protocol (plus security of the OWF against the verifiers)[2], our results hold in more general settings. For example, classical position verification is trivially possible if you assume there is a single adversarial spoofer; timing the messages of the prover will pinpoint its location. Plugging this simple positioning protocol into Theorem 1.1 yields an interesting example of a *purely classical* zero-knowledge location protocol secure against single-prover adversaries, assuming standard OWF exist. Similarly, our results also yield zero-knowledge location protocols in the classical-verifier, quantum-prover model of [LLQ22], where security is computational (see Section 3.3 for an in-depth explanation).

1.3 Technical overview

The key technical tool in our construction of zero-knowledge position verification is an object we call *position commitment*. This is a protocol that allows a prover to commit to its spacetime position (L, t) without initially revealing anything about it to the verifiers (i.e., the position commitment is *hiding*). Some time later, the prover can decide to *reveal* the pair (L, t) to the verifiers, who can perform a check to ensure that the prover was truly in the location L at time t (i.e., the position commitment is *binding*).[3]

[2] What our construction actually uses is secret-key encryption and bit commitment, which are equivalent to OWF.

[3] We call this second property "position binding" to mirror the binding property of cryptographic commitments. Note, however, that our notion is significantly stronger than conventional binding: the prover is not only bound to the choice of a single point (L, t) – it must have *physically occupied* (L, t).

We define and construct this primitive primarily as a building block for zero-knowledge position proofs, but we point out that position commitments by themselves already enable interesting applications. For example, an individual could continuously commit to their position at regular intervals throughout the day, which would establish a *private* record of their whereabouts. Later, if necessary, they could unseal part of this record in a courtroom, to convince a judge of their alibi.

Our first technical result is that position commitments can be constructed from one-way functions and nice singleton position verification protocols.

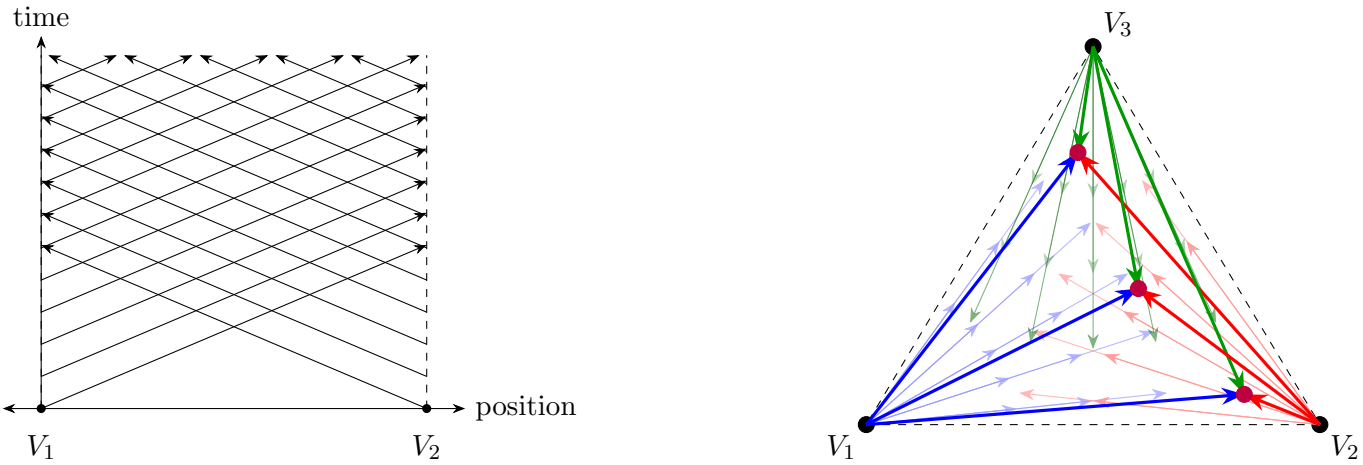

Figure 2: The position commitment described in Protocol 1 pictured for 1D (left), and 2D (right). In d spatial dimensions, intersections between $d+1$ different verifier messages represent spacetime points at which an honest prover might receive a set of position verification challenges, which they will respond to in an encrypted way. A few example points have been bolded on the right for clarity.

Theorem 1.3 (informal version of Lemma 3.3)**.** *Suppose there exist post-quantum one-way functions and nice singleton position verification protocols which have position security against spoofers sharing at most E entangled qubits. Then for every finite spacetime region S, there exists a position commitment protocol with completeness for all points in S, computational hiding against semi-honest verifiers, and statistical position binding against spoofers sharing $E/2$ qubits of entanglement.*

Constructing position commitments. We give a high level overview of our construction of position commitments. We'll outline what the honest verifiers V and honest prover P do.

Suppose prover P is planning to be in the spacetime location $(L^*, t^*) \in S$ where t^* is in the future. Without announcing (L^*, t^*) to the verifiers, P initiates a *session* that takes place between times $t_{\text{init}} < t_{\min}(S) < t_{\max}(S) < t_{\text{final}}$ wherein it interacts with V to commit to its hidden position (here, $t_{\min}(S)$ and $t_{\max}(S)$ are the earliest and latest times of points in S, respectively). In this session, the prover and the verifiers engage in a continuous stream of singleton position verification protocols, running enough instances to cover the entire space of committable points where P may be located. However, in order not to expose its location, P encrypts all of its messages under a secret key, which the verifiers will only learn when the prover decides it wants to reveal its commitment. The protocol is constructed such that no matter where P is, it can convincingly pretend to be

at all other points in S simultaneously, by sending dummy messages which meet those timing constraints. In order to prevent the prover from changing its mind about the contents of the encrypted messages by finding a different decryption key, the prover commits to the encryption secret key at the beginning of the protocol. The protocol runs in time $\text{poly}(\lambda, |S|)$. We illustrate the position commitment protocol for dimensions $d = 1$ and $d = 2$ in Figure 2.

Proving security of position commitments. It is not difficult to see that *hiding* security of the above construction follows from two facts: 1. all prover messages are indistinguishable assuming the security of secret-key encryption, and 2. the set of message timings received by the verifiers is kept static, regardless of the prover's true position.

Proving the *position binding* of this construction is, however, considerably less straightforward. The proof, a reduction to position verification soundness and classical commitment binding, is unlike typical reductions in the cryptography literature. Namely, there is a challenge in manipulating the *real-time* inputs and outputs of the adversary – it is unclear what "fast-forwarding" and "rewinding" might mean in a context where the adversary is distributed across concrete points in space and time. In our solution to this, the reduction is required to take *nonuniform quantum advice*.

From position commitments to zero-knowledge position verification. We now describe how position commitments can be used as a building block to construct zero-knowledge position proofs, which proves Theorem 1.1. The method we use for this extension is very similar to how, in the classical case, commitments are used to achieve zero-knowledge for general NP predicates. Once a prover has interacted with some verifiers to create the position commitment (which is, in our construction from nice position verification protocols, a classical string), it engages in a zero-knowledge proof protocol for some statement about the hidden position. In particular, the prover can express an NP-verifiable statement of the fact that it possesses an opening to the commitment such that the revealed point belongs to R, and employ the standard zero-knowledge proof system for NP languages. The soundness of this proof, combined with statistical position binding of our position commitments, gives us position security against dishonest provers.

Optimizing the position commitment protocol. We construct position commitments for a fairly general class of position verification protocols (all so-called *nice* protocols), which encompasses most of what has been explored in the literature. However, this generality comes with a cost to the efficiency of our protocols. In Section 3.3, we narrow our assumption further to protocols whose messages are fully classical, and whose verifier challenges can be generated independently per verifier, and show how this can be used to greatly optimize the computational requirements of our protocol.

1.4 A brief overview of quantum position verification

The study of quantum position verification (QPV) has blossomed in recent years. A number of protocols have been proposed [KMS11, BCF+14, Unr14, BCS22, LLQ22, ES25], some with provable security guarantees, and others' that are conjectured. Fascinatingly, unconditionally-secure position verification is not possible even in the quantum setting; it was shown by [BCF+14] that the so-called "instantaneous nonlocal quantum computation" protocol of Vaidman [Vai03] (subsequently improved by [BK11]) can be used to break any QPV protocol, provided that the

attackers can utilize exponential amounts of quantum entanglement. The central question in this area is to prove the security of a QPV protocol in the presence of adversaries that can only share polynomial amounts of entanglement. This challenge, in turn, is connected with deep questions in a number of seemingly-unrelated areas, ranging from *classical* lower bounds for information-theoretic cryptography [ABM+24, AKL+25, ACM25, GMOW25], the AdS/CFT correspondence in quantum gravity [May19], and conjectures about Banach spaces [JKPP22].

A recent interesting result by [KLC+24] showed that classical-verifier position verification protocols such as that of [LLQ22] are, in general, equivalent to protocols for *certifiable randomness*, hinting at deep connections between QPV and other areas of quantum cryptography.

1.5 Summary and future directions

Our concept of zero-knowledge position verification gives a solution towards achieving both privacy and provability for statements about one's physical location. We show that it can be achieved using one-way functions and (non-private) quantum position verification protocols. We view our contribution as taking the first steps in exploring the natural combination of privacy and proofs of position. We identify some interesting directions for future work.

Towards malicious-verifier security (full discussion in Section 5). We define and construct position commitments and zero-knowledge position verification protocols where the privacy condition holds with respect to *semi-honest* verifiers. We view this model as well-motivated by itself; for example, one can imagine that the verifiers are cell towers whose goal is to provide consistent position verification coverage to many customers over a wide area. Although the cell phone company may be interested in tracking their customers' whereabouts by examining communication logs, it is not sufficiently motivated to deviate from the protocol in a way that can be noticed by customers.

Can we obtain a stronger form of zero-knowledge position verification against malicious verifiers who deviate from the protocol? Unfortunately, we cannot hope to achieve zero-knowledge against arbitrary cheating verifiers; in Section 5 we present a generic attack where the verifiers can determine whether a prover is in a location L by actively sending directional messages towards L, but nowhere else. Whether or not the prover responds will leak whether the prover is in location L. We believe this is an interesting property, which is inherent to any protocol where the honest prover would respond to directional messages (either because the honest verifier behavior requires sending directional messages, or because it cannot distinguish between messages that are directional vs broadcast). On the other hand, we conjecture that in other settings, a more standard "HVZK-to-ZK" upgrading techniques should apply, and security against malicious verifiers can be added. That is:

Conjecture 1 (informal)**.** *Suppose we have a honest-verifier position verification protocol which requires the verifier to send only broadcast messages, and further suppose that the honest prover can tell whether a message it received was directional rather than broadcast. Then the protocol can be upgraded to have security against malicious verifiers.*

We note that position verification protocols that require only broadcast messages exist (e.g., ones where the communication is classical). Thus, we believe that formalizing our attack and proving the above conjecture is a promising avenue for future work.

Towards stronger position verification guarantees. When we carefully consider the soundness of position verification, we find that there is a subtle gap between the *intuitive* notion of position security and the *formal* definition presented here and in previous papers on position verification, starting from the earliest ones [CGMO09, BCF+14]. In the intuitive notion, a position verification protocol allows someone to prove the statement "I was at location X". However, in the formal definition, the protocol only guarantees that, of a coalition of colluding provers, *at least one* of the provers (rather than *the* prover) was at location X. It may be desirable to have a stronger guarantee that all of the "real" computation done to pass the protocol was done at the claimed location.

How do we define who the "real" prover or computation is (who is "I")? One potential direction is to leverage unclonability. Imagine a position verification protocol with the following structure:

1. At the beginning of time, the prover receives from the verifiers a quantum state $|\psi\rangle$, which is unclonable and has some cryptographic function. Let's say, for example, that $|\psi\rangle$ is a quantum signing token (like that of [BS23]).

2. During the position verification protocol, the prover is required to sign its responses with $|\psi\rangle$.

3. By an appropriate security reduction, we would then be able to claim, "the PV protocol passes if and only if $|\psi\rangle$ was in location X".

We note, however, that formalizing this security notion, with the goal of fully capturing our intuitive notion, still seems difficult. What does it mean for $|\psi\rangle$ to be *in* location X – what if it is distributed across a large error-correcting code? Questions like these are explored in a 2016 paper [HM16], but it is not immediately clear how to apply their results to this setting. An additional question is, under what assumptions can we say that $|\psi\rangle$ truly refers to a *unique* object in the universe, which has a canonical position that we can refer to? Is some version of this statement implied by unclonability alone?

Simplifying our nonuniform security reduction. In Section 3 we give a construction of a new position-based cryptographic primitive, which we call position commitments; we then prove the soundness of this construction by a reduction to secure position verification and classical commitment binding. The reduction, which is found in Lemma 3.3, operates in an atypical way and requires nonuniform quantum advice. The reasons for this are, in brief, as follows.

Our reduction has white-box access to a distributed cheating adversary for our position commitment scheme, and must use this to win in the standard position verification experiment (or break the statistical binding of Naor commitments). In the position commitment experiment, the cheating adversary receives a set of PV challenges, and outputs a set of *encrypted* PV responses, which will then be decrypted at a later time whenever the adversary chooses to reveal the secret key. A natural reduction strategy would be to feed this adversary the PV challenges, and quickly fast-forward the adversary until it 1. outputs the encrypted responses, and then later 2. outputs the decryption key – after this, the reduction can simply output the plaintext PV responses and win in the PV experiment.

However, we run into a problem: the distributed adversary might need to exchange messages between its various components, *which are spatially distant*. Therefore, it is physically impossible for our reduction to fast-forward a general adversary by any significant amount of time, as the reduction needs to respond instantaneously to its challenges and cannot wait for the exchanging

of messages at the speed of light. In our proof of Lemma 3.3 by endowing the reduction with a nonuniform quantum advice state, containing the secret decryption key that the adversary will eventually output. Can this be simplified, and the nonuniformity removed?

Towards more realistic models. Finally, the model we used in this paper (and which is consistent with prior work) is highly idealized, including precise, synchronized clocks, and instantaneous computation. Some of this seems inherent not only to the main ideas underlying existing protocols (based on timing of responses) but also to any real-world quantum information processing. Finding a way to relax these assumptions could help bring these protocols a step closer to practical realizability.

2 Preliminaries

2.1 Notation

We take λ to be the security parameter. We write PPT and QPT to denote probabilistic polynomial-time and quantum polynomial-time, respectively. We write $\mathrm{negl}(\lambda)$ to denote a negligible function in λ. For two algorithms P and V (implicitly, prover and verifier), we write $(P \leftrightarrows V) \in \{\text{accept}, \text{reject}\}$ to denote the interaction between P and V, where the accept/reject decision is output by V at the conclusion of the interaction. When the parties share a common input x, the interaction will be denoted $(P \leftrightarrows V)(x)$. V's view of a (classical-message) interaction with P, denoted $\mathrm{View}_V(P \leftrightarrows V)$, is defined as the full transcript of messages exchanged between P and V, as well as all random coins flipped by V, during the interaction. For quantum-message interactions between P and V, we define V's view at a given time τ, denoted $\mathrm{View}_V^\tau(P \leftrightarrows V)$, to be the quantum mixed state corresponding to V's internal state at time τ.

2.2 The model

We now describe the model of space, time, and interaction used in the paper.

Spacetime. We model space as $\mathbb{R}^d$ for some dimension d, and time as $\mathbb{R}$. We assume that all parties have access to a synchronized clock that reports the current time $t \in \mathbb{R}$. A *point in spacetime* is a pair $(L, t) \in \mathbb{R}^d \times \mathbb{R}$, combining a spatial point with a time. For the sake of simplicity, we assume that all parties are located at points $(L, t) \in \mathbb{Q}^d \times \mathbb{Q}$; this allows finite representations of their positions.

Each algorithm A is identified with a unique spatial point at all times t, called its *position*. We use the following notations interchangeably:

1. A occupies/is positioned at the (spacetime) point (L, t).
2. A's position at time t is L.
3. $L = \mathsf{loc}_t(A)$

For a spacetime region $R \subseteq \mathbb{R}^d \times \mathbb{R}$, we may write $\mathsf{loc}(A) \in R$ to mean that there exists a time t such that $(\mathsf{loc}_t(A), t) \in R$, and $\mathsf{loc}(A) \notin R$ for the negation of this.

Signals. We assume there are two types of signals: *directional* and *broadcast*. A directional signal is emitted from some location and travels along some ray in $\mathbb{R}^d$, and only spatial points located on the ray can receive the signal (e.g., a party can transmit a "laser beam"). A broadcast signal is emitted from some location and eventually can be received by all points in space (e.g., a party can send a "radio broadcast"). In general, a signal is a quantum state, although we often distinguish between the classical and quantum parts of a given signal (for example, a party may send a string x and a qubit $|\psi\rangle$). Only classical strings will be broadcast.

By convention, we assume that signals travel at a fixed speed of one unit of space per unit time (i.e., the "speed of light"). For example, if a directional signal is sent from the origin in the direction of some unit vector $\mathbf{m} \in \mathbb{R}^d$ at time 0, the signal arrives at point $t\mathbf{m} \in \mathbb{R}^d$ at every subsequent time $t > 0$. For broadcast signals, the previous statement is true for *all* unit vectors $\mathbf{m} \in \mathbb{R}^d$.

Computation and Interaction. We model each party as some quantum algorithm A that can receive and transmit classical and quantum information. Its classical inputs are: a security parameter 1^λ represented in unary, its current location L (expressed in binary), and the time t (expressed in binary). Its quantum inputs are a register containing its internal memory state, and a register representing the (quantum) messages that are received at precisely time t. If there are no messages received, then the message register is given some encoding of the $\perp$ symbol. The output of the algorithm A is a pair $(\texttt{mode}, \mathbf{m})$ and two quantum registers where

1. $\texttt{mode} \in \{\texttt{directional}, \texttt{broadcast}, \perp\}$ indicates whether the message is supposed to be directional, broadcast, or not sent at all,
2. The vector $\mathbf{m} \in \mathbb{R}^d$ specifies the direction if needed, and
3. The first quantum register is the party's internal memory state (to be passed to the same party in the next time step), and the second quantum register contains the message to be sent.

If $\texttt{mode} = \texttt{broadcast}$, then the state ρ is measured to obtain a classical string x and broadcast everywhere.

We assume that algorithms have two interfaces with the global clock: a discrete "tick" update on which they can run their loops, and a "real-time" interface with which they can, for example, send out a message at any $t \in \mathbb{R}$. Since we assume that each party has a unique position $L \in \mathbb{R}^d$ at each time $t \in \mathbb{R}$, we model its behavior at each clock tick t as computing the output $A(1^\lambda, L, t, \mathsf{E}, \mathsf{A})$ where E, A denote its internal memory and message registers, respectively; the message register is some well-defined concatenation of all the signals received at spacetime point (L, t). We will henceforth elide these inputs (save for the security parameter) as they are implicit from context.

By default, we assume that the computational complexity of the quantum algorithm A is polynomial in the security parameter λ. For simplicity, we separate the time it takes to compute $A(1^\lambda, L, t, \cdots)$ from the ticking of the global clock t. That is, we assume A computes its output in one time step.

In this paper, there are two types of parties: verifiers and provers. We focus on honest verifiers whose locations are always known to all parties. We discuss possible approaches to malicious verifiers in Section 5.

These modeling decisions were chosen to be consistent with the prior literature on position verification, while being as simple as possible in order to illustrate the essential ideas behind zero

knowledge position verification. Extending the model to handle inaccuracies, errors, unsynchronized clocks, etc. are fascinating directions for future work.

2.3 Cryptographic Notions

We recall the definitions of some basic cryptographic primitives that will be used in the paper.

Definition 2.1 (Computational Indistinguishability)**.** *The families of random variables* $\{X_\lambda\}_{\lambda\in\mathbb{N}}$ *and* $\{Y_\lambda\}_{\lambda\in\mathbb{N}}$ *are* ***computationally indistinguishable*** *if for every QPT distinguisher* D*, there exists a negligible function* negl *such that for all* $\lambda \in \mathbb{N}$*:*

$$\left|\Pr\left[D(1^\lambda, X_\lambda) = 1\right] - \Pr\left[D(1^\lambda, Y_\lambda) = 1\right]\right| \leq \mathrm{negl}(\lambda) \ .$$

For random variables X *and* Y*, where the dependence on* λ *is implicit from context, we denote the above with* $X \approx_c Y$*. Similarly, we say that families of quantum states* $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ *and* $\{\sigma_\lambda\}_{\lambda\in\mathbb{N}}$ *are computationally indistinguishable if they cannot be distinguished by QPT algorithms with more than negligible advantage.*

Definition 2.2 (Secret-Key Encryption)**.** *Let* $\mathcal{E} = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ *be a tuple of PPT algorithms, such that for all* $\lambda \in \mathbb{N}$ *and all* $x \in \{0,1\}^*$*,*

$$\Pr\left[\mathrm{Dec}(\mathrm{sk}, \mathrm{Enc}(\mathrm{sk}, x)) = x : \mathrm{sk} \leftarrow \mathrm{Gen}(1^\lambda)\right] = 1 \ .$$

We say $\mathcal{E}$ *is a* ***post-quantum encryption scheme*** *if for any QPT algorithm* A *with oracle access to* $\mathrm{Enc}(\mathrm{sk}, \cdot)$ *and which outputs two equal-size lists of plaintexts, the following holds. With* $(a_1 \dots a_n), (b_1 \dots b_n) \leftarrow A^{\mathrm{Enc}(\mathrm{sk},\cdot)}(1^\lambda)$ *and* $\mathrm{sk} \leftarrow \mathrm{Gen}(1^\lambda)$*,*

$$(\mathrm{Enc}(\mathrm{sk}, a_1) \dots \mathrm{Enc}(\mathrm{sk}, a_n)) \approx_c (\mathrm{Enc}(\mathrm{sk}, b_1) \dots \mathrm{Enc}(\mathrm{sk}, b_n)) \ .$$

Definition 2.3 (Commitment Scheme)**.** *A* ***commitment scheme*** $\mathcal{C}$ *is a pair* $(\mathrm{Setup}, \mathrm{Com})$ *of PPT algorithms with the following syntax:*

- $\mathrm{Setup}(1^\lambda)$ *outputs a random string* pp *(for "public parameters"), which we assume contains the security parameter* 1^λ*.*
- $\mathrm{Com}(\mathrm{pp}, x; r)$ *is a deterministic function of* (pp, x, r) *where* $r \in \{0,1\}^\lambda$*.*

We say that $\mathcal{C}$ *is* ***post-quantum computationally hiding*** *if for all QPT algorithms* A*, with* $\mathrm{pp} \leftarrow \mathrm{Setup}(1^\lambda)$ *and* $(x_0, x_1) \leftarrow A(1^\lambda, \mathrm{pp})$*, the following statement holds:*

$$\left\{\mathrm{Com}(\mathrm{pp}, x_0; r) : r \leftarrow \{0,1\}^\lambda\right\}_{\lambda\in\mathbb{N}} \approx_c \left\{\mathrm{Com}(\mathrm{pp}, x_1; r) : r \leftarrow \{0,1\}^\lambda\right\}_{\lambda\in\mathbb{N}} \ .$$

We say that $\mathcal{C}$ *is* ***statistically binding*** *if for all (computationally unbounded) algorithms* B*, there is a negligible function* negl *such that the following holds for all* $\lambda \in \mathbb{N}$*:*

$$\Pr\left[\begin{array}{c} x_1 \neq x_2 \\ \mathrm{Com}(\mathrm{pp}, x_1; r_1) = \mathrm{Com}(\mathrm{pp}, x_2; r_2) \end{array} : \begin{array}{c} \mathrm{pp} \leftarrow \mathrm{Setup}(1^\lambda) \\ (r_1, r_2, x_1, x_2) \leftarrow B(\mathrm{pp}) \end{array}\right] \leq \mathrm{negl}(\lambda) \ .$$

Definition 2.4 (Zero-Knowledge Proof)**.** *Let $L \subseteq \{0,1\}^*$ be a language. A pair (P, V) of classical PPT interactive algorithms is an **honest-verifier, (post-quantum) computational zero-knowledge proof system** for L with completeness error $c(\lambda)$, soundness error $s(\lambda)$ if the following hold.*

1. ***(Completeness)** For all $\lambda \in \mathbb{N}$ and for all $x \in L$, there exists a string w (called the witness) such that*
$$\Pr\left[(P(w) \leftrightarrows V)(1^\lambda, x) \text{ accepts}\right] \geq 1 - c(\lambda)$$
2. ***(Soundness)** For all $\lambda \in \mathbb{N}$ and $x \notin L$, for all provers P^*,*
$$\Pr\left[(P^* \leftrightarrows V)(1^\lambda, x) \text{ accepts}\right] \leq s(\lambda) \ .$$
3. ***(Honest-Verifier Post-Quantum Computational Zero Knowledge)** There exists a PPT simulator* Sim *such that for all $x \in L$, the following holds:*
$$\left\{\mathrm{Sim}(1^\lambda, x)\right\}_{\lambda\in\mathbb{N}} \approx_c \left\{\mathrm{View}_V(P(w) \leftrightarrows V)(1^\lambda, x)\right\}_{\lambda\in\mathbb{N}}$$
for all witnesses w that make the completeness condition hold.

Fact 2.5 (Existence of post-quantum encryption, commitments, and zero-knowledge proofs)**.** *Assuming the existence of post-quantum one-way functions, there exist the following:*

1. *Post-quantum secret-key encryption [KL14],*
2. *Post-quantum computationally hiding and statistically binding commitments [Nao91], and*
3. *Post-quantum honest-verifier computational zero-knowledge proof systems for any $L \in \mathsf{NP}$ with perfect completeness (i.e., zero completeness error) and negligible soundness [GMW91].*

Remark 2.6. Although the referenced constructions of secret-key encryption, commitments, and zero-knowledge proofs do not explicitly analyze quantum adversaries, it is straightforward to see that the computational security properties are inherited directly from the underlying one-way function. When the one-way functions are post-quantum secure, the security holds against QPT algorithms.

2.4 Position Verification

We present the formal definition of position verification protocols for arbitrary spacetime regions $R \subseteq \mathbb{R}^d \times \mathbb{R}$ (which generalizes the standard notion of position verification).

Definition 2.7 (Position Verification)**.** *Let $V = \{V_1, \ldots, V_k\}$ be a set of verifier algorithms. Let $R \subseteq \mathbb{R}^d \times \mathbb{R}$ be a set called the **allowable prover spacetime positions**, and $P = \{P_\alpha\}_{\alpha\in R}$ be a family of honest prover algorithms such that $\mathsf{loc}_t(P_\alpha) = L$ for each allowed spacetime point $\alpha = (L, t) \in R$. Then $\Pi = (P, V, R)$ is a **position verification protocol** with completeness error $c(\lambda)$ and position security $s(\lambda)$ against a class $\mathscr{C}$ of spoofers if the following are true.*

1. ***(Completeness)** For all allowed points $\alpha \in R$,*
$$\Pr\left[(P_\alpha \leftrightarrows V)(1^\lambda) \text{ accepts}\right] \geq 1 - c(\lambda).$$

2. ***(Position Security)*** *For all prover coalitions* $P^* = \{P_1^*, \ldots, P_m^*\} \in \mathscr{C}$ *where* $\mathsf{loc}(P_i^*) \notin R$ *for all* i*, it holds that*

$$\Pr\left[(P^* \leftrightarrows V)(1^\lambda) \text{ accepts}\right] \leq s(\lambda) \ .$$

If $|R| = 1$*, we call* Π *a* ***singleton*** *position verification protocol.*

We now introduce a special subclass of position verification protocols that will be needed for our construction of position commitments and zero-knowledge position proofs.

Definition 2.8 (Nice Protocols)**.** *We say that a family of singleton position verification protocols* $\{\Pi_i^{\text{single}}\}_{i \in I}$ *is* ***nice*** *if the following conditions are true:*

1. ***(One-shot, classical-response)*** *Each* $(P = \{P_{(L,t)}\}, V, R = \{(L,t)\}) = \Pi_i^{\text{single}} \in \Pi$ *has the following structure:*

   - *At the start of the protocol, the verifiers* $V_1, \ldots, V_k$*, using a shared random string* s*, jointly prepare a quantum state* $|\psi\rangle$ *(which might be entangled between all the verifiers, or could only be possessed by a subset of verifiers) and a set of classical messages* $(x_1, \ldots, x_k)$ *which is a deterministic function of* s*. Then, each verifier* V_i *sends a message containing their share of* $|\psi\rangle$*, along with* x_i*, to the purported prover at the spacetime point* (L, t)*.*
   - *When the honest prover* $P_{(L,t)}$ *receives the messages* $x_1, \ldots, x_k$ *as well as the state* $|\psi\rangle$*, it performs some measurement on* $|\psi\rangle$ *based on* $x_1, \ldots, x_k$*, and sends the classical measurement result* y *to all the verifiers.*
   - *Let* $t_1, \ldots, t_k$ *be the clock times that* y *is received by the verifiers* $V_1, \ldots, V_k$*, respectively. At the end of the protocol, the verifiers collectively compute some deterministic predicate* $W(s, y, t_1, \ldots, t_k)$ *and accept if and only if the output is* 1*.*

2. ***(Coverage)*** *For every finite spacetime region* $S \subset \mathbb{R}^d \times \mathbb{R}$*, there exist spatial points* $X_1^S, \ldots, X_k^S \in \mathbb{R}^d$ *whose convex hull contains* S*, and which additionally satisfy the following. For every point* $\alpha \in S$*, there is a protocol* $(P, V, R = \{\alpha\}) \in \Pi$ *in which* $V_1, \ldots, V_k$ *are positioned statically*[4] *at* $X_1^S, \ldots, X_k^S$*, respectively.*

For conciseness of notation, we will typically use the singular form (a nice position verification protocol) to mean a nice family of position verification protocols.

Although Definition 2.8 clearly constrains the space of all position verification protocols, we argue that it actually describes a very natural structure which occurs in many of the position verification schemes from prior literature. The first condition of Definition 2.8 describes this natural structure. As for the second condition, it has been widely noted by prior work that a necessary condition for secure position verification is that the target point be located within the *convex hull* of the verifiers' positions. It is also the case, in all prior position verification protocols we have come across, that this is a sufficient condition.

For example, the f-BB84 protocol described in the introduction, and illustrated in Figure 1, is structured in the form of a nice position verification protocol. We will state this more concretely.

[4]The fact that the verifiers are stationary is not used explicitly, only the fact that the positions at each time are publicly known. We do not, therefore, rule out the possibility of accomplishing our protocols with mobile verifiers (e.g. medium-earth orbit satellites, like GPS)

Remark 2.9 (f-BB84 is nice). For all bounded spacetime regions $S \subset \mathbb{R}^d \times \mathbb{R}$, let $X_1^S, \ldots, X_{d+1}^S \in \mathbb{R}^d$ be verifier positions whose convex hull contains S. Then, for each $\alpha \in S$, let (P_α, V_α) be the f-BB84 prover and verifier algorithms for α, respectively, where the verifiers are positioned at $X_1^S, \ldots, X_{d+1}^S$. Then $\{(P_\alpha, V_\alpha, \{\alpha\})\}_{S \subset \mathbb{R}^d \times \mathbb{R}, \alpha \in S}$ is nice family of singleton position verification protocols. The protocol is perfectly complete, and has position security $1 - \varepsilon$ against the class of adversaries who pre-share $O(\log n)$ qubits of entanglement, for some constant $\varepsilon > 0$ [BCS22]. Position security can be strengthened to vanish asymptotically via parallel repetition, which was shown in [ES25].

As mentioned in the introduction, Definition 2.8 also applies to the protocols of [LLQ22] and [Unr14].

3 Position Commitments

In this section we formally define position commitments, and present our main construction.

Definition 3.1 (Position Commitment)**.** *Let $V = \{V_1, \ldots, V_k\}$ be verifier algorithms, let $S \subset \mathbb{R}^d \times \mathbb{R}$ be a finite set called the committable points, and let $P = \{P_\alpha\}_{\alpha \in S}$ be a family of prover algorithms. Then, the tuple (P, V, S) is a* ***position commitment scheme*** *if it satisfies the properties below. The interaction between P and V is split into two phases,* Commit *and* Reveal*, which have the following syntax.*

1. *The input to the* Commit *phase is the security parameter 1^λ. The* Commit *phase takes place in some time window $[t_{\text{init}}, t_{\text{final}}]$, where $t_{\text{init}} \leq t_{\min}(S) \leq t_{\max}(S) \leq t_{\text{final}}$. At time t_{final}, the verifiers jointly output a state ρ, called the* ***commitment state****.*

2. *The input to the* Reveal *phase is the security parameter 1^λ, the commitment state ρ, and a claimed position $\alpha^* \in S$. The output of the* Reveal *phase is the verifiers' accept or reject decision.*

Let $\text{Commit}_{P \leftrightarrows V}$ *and* $\text{Reveal}_{P \leftrightarrows V}$ *denote the interaction between prover P and verifiers V in the* Commit *and* Reveal *phases, respectively. Let $\mathscr{C}$ be some class of spoofing provers. A position commitment scheme has* ***completeness error*** *$c(\lambda)$,* ***statistical position binding*** *$s(\lambda)$ against $\mathscr{C}$, and* ***computational honest-verifier hiding*** *if it satisfies the following properties:*

1. *(*Completeness*) For all $\lambda \in \mathbb{N}$, for all $\alpha \in S$,*

$$\Pr\left[\text{Reveal}_{P_\alpha \leftrightarrows V}(1^\lambda, \rho, \alpha) \text{ accepts} : \rho \leftarrow \text{Commit}_{P_\alpha \leftrightarrows V}(1^\lambda)\right] \geq 1 - c(\lambda) \ .$$

2. *(*Statistical Position Binding*) For all prover sets $P^* \in \mathscr{C}$, with probability at least $1 - s(\lambda)$ over $\rho \leftarrow \text{Commit}_{P^* \leftrightarrows V}(1^\lambda)$, there exists a spacetime point α_ρ such that two properties hold[5]:*

   (a) *Some prover in P^* occupied the point α_ρ.*

   (b) *For all other points $\alpha' \neq \alpha_\rho$, and all (computationally unbounded) algorithms A,*

$$\Pr\left[\text{Reveal}_{A \leftrightarrows V}(1^\lambda, \rho, \alpha') \text{ accepts}\right] = 0.$$

[5] In simpler terms, the position binding property essentially forces the adversary to (1) commit to a unique spacetime point, and (2) have actually occupied this point, in order to have any hope of convincing the verifiers in the Reveal phase.

3. *(*Computational Honest-Verifier Hiding*) There exists a QPT simulator* Sim *such that for all* $\alpha \in S$*, and all times* $t_{\text{init}} \leq \tau \leq t_{\text{final}}$*,*

$$\left\{\text{Sim}(1^\lambda, \tau)\right\}_{\lambda\in\mathbb{N}} \approx_c \left\{\text{View}_V^\tau(P_\alpha \leftrightarrows V)(1^\lambda)\right\}_{\lambda\in\mathbb{N}}.$$

Additionally, we say that a commitment scheme is ***mostly-classical*** *if it has the following two properties. (1) The output of the* Commit *phase is classical, and (2) the* Reveal *phase is a classical protocol where no random coins are flipped by the verifiers.*

Remark 3.2. Note that the position binding property serves two simultaneous roles, one which is like classical binding security of commitments, and one which is like the position security of position verification protocols. We could imagine separating position binding into a binding property and a position security property. An example of when we might want this is if a pair of provers at locations α_1 and α_2 run the position commitment scheme to create a commitment ρ, such that $\text{Reveal}(1^\lambda, \rho, \alpha_1)$ and $\text{Reveal}(1^\lambda, \rho, \alpha_2)$ both accept. This breaks the above definition of position binding, but in some sense the provers did not lie about their positions – there could be an application where we even *prefer* to give the provers this flexibility. We think it may be interesting for future work to consider variants of this, where the two roles are separated.

3.1 Constructing Secure Position Commitments

Lemma 3.3. *Assume there exists a nice family of singleton position verification protocols* $\{\Pi_\alpha = (P_\alpha, V_\alpha, \{\alpha\})\}_{\alpha\in S}$ *for some finite spacetime region* S*, with completeness* $c(\lambda)$ *and position security* $s(\lambda)$ *against spoofers who pre-share at most* $E(\lambda)$ *qubits of entanglement. Assume that post-quantum one-way functions exist. Then the construction in Protocol 1, instantiated with the aforementioned primitives, is secure under Definition 3.1 with completeness* $c(\lambda)$*, statistical position binding* $|S|(s(\lambda)+\text{negl}(\lambda))$ *against spoofers sharing* $E(\lambda)/2$ *bits of entanglement, and computational honest-verifier hiding. Additionally,* $\mathcal{C}$ *is mostly-classical.*

Proof. We first show that $\mathcal{C}$ is mostly-classical, which follows directly from the "one-shot, classical-response" property of the nice protocol Π, as well as the definitions of the honest prover and verifiers which do not introduce any quantum states, or flip any random coins in Reveal. Now we prove the rest of the claim:

Completeness: Take an honest prover P_α. At some point during Commit, V and P_α will execute the protocol Π_α, except that P_α's messages are encrypted. Since Π_α has the structure of a nice protocol, however, we can argue that by decrypting the prover's messages at reveal time and resuming verification, we reach the same state as an unmodified execution of Π_α. Thus the verifiers accept with probability $1 - c(\lambda)$.

Computational Honest-Verifier Hiding: For hiding, we observe that no matter where P is located, the transcript M of messages each verifier V_i during the Commit phase *always* consists of receiving the commitment c, and then a set of encrypted messages from the prover whose timings are consistent with Π_α, for every point $\alpha \in S$. This is made possible by the fact that P_α knows the precise time that each verifier expects each message in $\Pi_{\alpha'}$ for all $\alpha' \in S$, and has a sufficient head start on the verifiers to be able to send all of these messages in time. For any public parameters

Protocol 1 Encrypt-Then-Verify Position Commitment Protocol.

Let W_α be the verification predicate for Π_α, as described in Definition 2.8. Let $(\text{Setup}, \text{Com})$ be a classical commitment scheme, and let $(\text{Gen}, \text{Enc}, \text{Dec})$ be a secret-key encryption scheme (each with post-quantum security). We construct a position commitment scheme $\mathcal{C} = (P, V, S)$ as follows.

procedure COMMIT(1^λ)
  Some verifier V^* is chosen as a coordinator. At time t_{init}, V^* broadcasts $\text{pp} \leftarrow \text{Setup}_{\text{Com}}(1^\lambda)$.
  After receiving pp, P generates $\text{sk} \leftarrow \text{Gen}(1^\lambda)$ and $c \leftarrow \text{Com}(\text{pp}, \text{sk}, r)$ with $r \leftarrow \{0,1\}^\lambda$.
  Let T be the maximum time a message takes to travel between any point in S and any verifier V_i. P sends c to each verifier, with each message timed to arrive at $t_1 = t_{\text{init}} + 2T$.
  Immediately after sending c, P runs the following:
  Choose a time t^* such that $\mathsf{loc}_{t^*}(P) = L^*$ and $(L^*, t^*) \in S$.
  for each $\alpha \in S$ **in parallel do**
    if $\alpha = (L^*, t^*)$ **then**
      At time t^*, receive challenge $x_1 \dots x_k, |\psi\rangle$ from the verifiers.
      Measure $|\psi\rangle$ according to $x_1, \dots, x_k$ to produce y, then broadcast $(\alpha, \text{Enc}(\text{sk}, y))$.
    else (when $\alpha \neq (L^*, t^*)$)
      for each i in $1 \dots k$ **do**
        Compute the time t_i that V_i expects to receive the prover's response in Π_α.
        Send $(\alpha, \text{Enc}(\text{sk}, \perp))$ directionally to V_i, such that it arrives at time t_i.
      end for
    end if
  end for

  Meanwhile, $V_1, \dots, V_k$ jointly run the following program, starting at time t_1:
  for each $\alpha \in S$ **in parallel do**
    Take some shared randomness r_α and use it to prepare a challenge $|\psi\rangle, x_1, \dots, x_k$
    Each V_i sends x_i, and its respective portion of $|\psi\rangle$, to the spacetime point α.
    Each verifier timestamps and records all prover messages to their transcript.
  end for
  Let t_{final} be the latest arrival time of any prover message. Let M be the combined transcript of prover messages received by the verifiers, and let s be the verifiers' entire shared randomness.
  At t_{final}, the verifiers output $\rho = (\text{pp}, c, M, s)$
end procedure

procedure REVEAL($1^\lambda, \rho = (\text{pp}, c, M, s, (L^*, t^*))$)
  P sends (L^*, t^*), along with sk and r, to V^*.
  V^* checks $c = \text{Com}(\text{pp}, \text{sk}, r)$; reject if invalid.
  Verifiers create an empty list A of accepting positions.
  for $\alpha \in S$ **do**
    Each V_i extracts the prover message labeled $(\alpha, \dots)$ from their transcript of Commit, letting t_i be its timestamp, and decrypts it with $\text{Dec}(\text{sk}, \cdot)$. If any messages are $\perp$, or if the decryption differs between verifiers, **continue** to the next α. Otherwise, let y be the decryption.
    Verifiers compute $W_\alpha(s_\alpha, y, t_1, \dots, t_k)$. If it equals 1, then they add α to A.
  end for
  Accept if $A = \{(L^*, t^*)\}$, and reject otherwise.
end procedure

pp, we can define a simulated transcript M'_{pp} as follows: take $\text{sk} \leftarrow \text{Gen}(1^\lambda)$, $r \leftarrow \{0,1\}^n$, and $c \leftarrow \text{Com}(\text{pp}, \text{sk}, r)$. Add (t_1, c) to M'_{pp}, where $t_1 = t_{\text{init}} + 2T$ (recall that T is the maximum time to reach any verifier from any point in S). Then add "$V_i \leftarrow (\alpha, \text{Enc}(\text{sk}, \perp))$ @ t" to M'_{pp} for every time t where each verifier V_i would receive a message in Π_α for every $\alpha \in R$. Now, we define the simulator Sim to compute the following on input $(1^\lambda, \tau)$: Start simulating V, which outputs pp at time t_{init}. Construct the timestamped transcript M'_{pp} as defined above. Then, simulate V through time τ, while feeding it messages exactly according to M'_{pp}. Then, output V's internal state.

Once we are convinced that the simulated transcript matches the real transcript in terms of quantity and timing of messages, it follows straightforwardly from the hiding property of the classical commitment scheme, as well as the security of the secret key encryption scheme, that the output of the simulator is computationally indistinguishable from the verifiers' view during the honest commitment protocol.

Statistical Position Binding: Take a prover strategy P^* which pre-shares at most $E/2$ qubits of entanglement. Assume that P^* breaks statistical position binding with probability p – i.e. with probability p over $\rho \leftarrow \text{Commit}_{P^* \leftrightarrows V}$, one of conditions (a) and (b) in the definition is not satisfied. Because (Setup, Com) is statistically binding, P^* can only distinguish it from a perfectly binding scheme with negligible probability. Thus we can take an equivalent view which removes this event, and where we adjust all probabilities accordingly by a negligible amount.

So, we henceforth treat (Setup, Com) as perfectly binding, and say that P^* breaks statistical position binding with probability at least $p - \text{negl}(\lambda)$. Under this perfect binding, there cannot be two points α_1 and α_2 such that ρ can open to both, since decryption under sk is deterministic and due to the check for a unique point on line 35 of Protocol 1. This means that with probability $p - \text{negl}(\lambda)$, P^* breaks position security by *spoofing its position*. In other words, with probability $p - \text{negl}(\lambda)$ over $\rho \leftarrow \text{Commit}_{P^* \leftrightarrows V}$, there exists some Reveal strategy which succeeds in revealing ρ at a point that P^* never occupied. We can break this down into point-by-point cases: for any point $\alpha \in S$ occupied by no provers in P^*, let q_α be the probability that there exists some successful strategy for revealing to α. By a union bound over points in S, we can take some maximizing choice α^* such that $q^*_\alpha \geq (p - \text{negl}(\lambda))/|S|$. Let $q = (p - \text{negl}(\lambda))/|S|$ for simplicity of notation. To restate the above, we have found a point α^* which is not occupied by any prover in P^*, such that commitments generated by P^* can be opened to position α^* with probability at least q.

Now by averaging, choose some pp^* and c^* such that when we condition on $\text{pp} = \text{pp}^*$ and $c = c^*$ (the public parameters and commitment to the secret key, respectively) in the Commit phase, ρ can still be opened to α^* with at least the same probability q. Let sk, r be the unique values such that $c^* \leftarrow \text{Com}(\text{pp}, \text{sk}, r)$ – recall, we are treating Com as perfectly binding. Given this choice, let ω be the (possibly entangled) internal state across all provers in P^* just after the commitment c was announced in the Commit phase, conditioned on $c = c^*$ and $\text{pp} = \text{pp}^*$[6]. We now describe the reduction, which takes ω, sk as nonuniform advice:

For all i, let $\tilde{P}_i$ be a reduction algorithm positioned exactly as P^*_i was in the Commit phase of the original cheating strategy – that is, for all $t_{\text{init}} \leq t \leq t_{\text{final}}$, $\mathsf{loc}_t(\tilde{P}_i) = \mathsf{loc}_t(P^*_i)$. $\tilde{P}_i$ controls an instance of P^*_i, in the sense that it can read and write to P^*_i's internal state and can intercept messages which are sent and received by P^*_i. We now have $\tilde{P}$ participate in an execution of the vanilla position verification experiment Π_{α^*}, with verifiers $\tilde{V}$, whose positions are identical to the verifiers V from the commitment protocol. At the start of the experiment, the provers in $\tilde{P}$ set

[6]This can be thought of as a quantum mixed state across the combined registers of each prover.

the joint state across the registers of all of the simulated provers in P^* to the advice state ω. The verifiers execute Π_{α^*}, and the provers begin simulating the execution of the cheating strategy P^*. Whenever some prover $\tilde{P}_i$ detects that its simulation of P_i^* is sending a message (α, m) and $\alpha = \alpha^*$, it intercepts this and computes $m' = \mathrm{Dec}(\mathrm{sk}, m)$. $\tilde{P}_i$ then forwards m' to the intended verifier.

We now analyze the success probability of this reduction. In the commitment experiment, in order for P^* to have generated a commitment which can open to α^* after reaching the state ω, it must have sent a set of messages of the form $(\alpha^*, \mathrm{Enc}(\dots))$ during the Commit phase, whose decryptions under sk would cause the verifiers for Π_{α^*} to accept. This is clear from the verification process in the protocol, along with the fact that the provers had output c^* which is perfectly binding to sk. Now since $\tilde{P}$ sends exactly P^*'s messages but decrypted under sk, then $\tilde{V}$ must accept whenever P^* would have generated such a commitment. Thus, the reduction convinces the verifiers $\tilde{V}$ with exactly probability q.

Now, note that after we added the advice state ω to the cheating strategy $\tilde{P}$, the amount of preshared entanglement increased to be at most E qubits. Since we assumed that Π_{α^*} has position security $s(\lambda)$ against this class of prover, we have shown that our position commitment protocol has position binding $|S|(s(\lambda) + \mathrm{negl}(\lambda))$ against spoofers with $E/2$ qubits of preshared entanglement. □

3.2 Relationship to Other Quantum Cryptographic Primitives

Proposition 3.4. *Assume there exists a position commitment scheme $\mathcal{C} = (P, V, S)$ with completeness $c(\lambda)$, statistical position binding $s(\lambda)$ against some class $\mathscr{C}$ of adversaries, and computational hiding security. Then, the following hold:*

1. *There exists a (not necessarily nice) singleton quantum position verification protocol for each point in S, with completeness $c(\lambda)$ and with position security $s(\lambda)$ against $\mathscr{C}$.*
2. *If there are no computational limits on $\mathscr{C}$, then there exist quantum bit commitments with completeness $c(\lambda)$, $s(\lambda)$-statistical binding, and honest-verifier computational hiding against semi-honest receivers.*

Proof. We will start with point 1. The position verification protocol $\Pi = (P, V, \{(L, t)\})$ is defined as follows: P and V run the Commit phase of $\mathcal{C}$, and then proceed immediately to the Reveal phase. The hiding property is no longer important, but it is immediately clear that the completeness and position binding of $\mathcal{C}$ are inherited directly as the completeness and position security of Π, respectively.

We move on to point 2, constructing an interactive protocol for (semi-honest receiver) quantum commitments. Taking some arbitrary spatial point L, designate two fixed spacetime points $(L, t_0), (L, t_1) \in S$ to represent bits 0 and 1, respectively. To commit to $b \in \{0, 1\}$, the sender (who we will call P, since it takes the place of the prover) will engage the receiver (who we will call V, since it takes the place of the verifier) in a simulated version of the Commit phase of $\mathcal{C}$, where it commits to its position (L, t_b). The simulation will take place in the (untimed, un-positional) interactive-protocol setting as follows. There is a round of interaction for each timestep t of the Commit phase:

- V sends P any messages **which would arrive** at (L, t).
- P sends V any messages **which it would send** at time t.

The output of this protocol is the commitment state ρ. To reveal, the parties simply run Reveal in the straightforward way.

We now argue for the completeness and hiding properties. The reason for the asymmetry, where V sees P's messages slightly *ahead* of schedule, is because we assume V behaves semi-honestly and so cannot use this information to distort the protocol. Notice, then, that P's view at any time τ is identical to what it would have been in the protocol – this is true no matter which bit P commits to, since it's simulating a prover at the same spatial position in either case. Given that the prover's final view is identical to that of Commit, it also follows that the verifier's final view is identical to that of Commit. From this it is easy to see that the protocol has completeness $c(\lambda)$. Additionally, the verifier's view is trivially simulatable by removing the timestamps from the simulator provided by the honest-verifier computational hiding security of $\mathcal{C}$. The semi-honest-receiver computational hiding property follows.

The statistical binding property follows in a similar fashion. If it was violated, we could construct a prover for $\mathcal{C}$ which breaks position binding with the same probability, by sending exactly the messages that are sent in the simulation above. □

In the case of quantum commitments, luckily, we can leverage some nice equivalence results to strengthen the properties relative to what we showed above.

Theorem 3.5 ([Yan22])**.** *Any (interactive) quantum bit commitment scheme whose purification is semi-honest secure can be converted into a non-interactive quantum bit commitment scheme with perfect completeness, and whose hiding and binding properties are exactly preserved.*

In our case, the purified version would maintain hiding (since the prover keeps the purified secret key privately) and binding (purification cannot help break classical (post-quantum secure) commitments in any way), so we can apply the theorem.

Corollary 3.6. *Take the assumptions in Proposition 3.4. Then, there exists a non-interactive quantum bit commitment scheme with perfect completeness, $s(\lambda)$-statistical binding, and computational hiding.*

3.3 Optimizations for Concrete Constructions

In Section 3.1, we showed how to construct position commitments from any nice position verification protocol. However, our construction is quite computationally demanding on both the prover and verifier algorithms, which puts some strain on our modeling assumption of instantaneous computation. In particular, it seems challenging for the algorithms to run a position verification protocol for every point in the committable set *simultaneously*. We would prefer a solution which only requires a small, constant amount of computation per timestep.

We will now informally describe an optimized construction which can hopefully serve as evidence for the real-world implementability of position commitments and, subsequently, zero-knowledge position proofs. We will require a couple of additional properties from the underlying position verification protocol, on top of being nice: the verifiers' messages must be **1.** entirely classical, and **2.** generated independently of one another. We note that this is satisfied, for example, by the protocol of [LLQ22][7]. The first property is so that the messages can be broadcast, instead of

[7]The main protocol of [LLQ22] is only discussed in the 1-D setting, but they give another protocol in higher dimensions which is conjectured to be secure, and also satisfies the constraints of our optimization

directional (since there is no meaningful notion of "broadcasting" a quantum state). The second property allows the verifiers to build a mesh of challenges throughout the spacetime region, as pictured in Figure 3. Wherever a prover is, there will be some combination of independently sampled challenges arriving at (or very near) that spatial point at some time, which the prover can then respond to under encryption as before. As for the prover's task of pretending to be in all other points at once: observe that as long as the verifiers to receive at least one encrypted message at each timestep of the protocol, they cannot rule out the presence of a prover at any given position. On the other hand, a prover can easily induce such a state by sending each verifier one encrypted message per timestep (where, similarly to Protocol 1, it sends directional messages delayed appropriately so that all verifiers receive the first one and the last one at the same time). The optimizations are informally described below.

Protocol 2 (Optimized Encrypt-Then-Verify Scheme)**.** *We define this protocol informally by specifying the main loop of the provers and verifiers, and where everything outside of this loop proceeds in roughly the same way as Protocol 1.*

- *The verifiers do the following* at each timestep t*: each verifier V_i generates a challenge string $x_{i,t} \in \{0,1\}^n$, chosen from the verifiers' shared randomness, and broadcasts it.*
- *The honest prover at location L, at each timestep t, checks whether it received a set of verifier challenges $(x_1, \ldots, x_{d+1})$. If not, P sends* $\mathrm{Enc}(\perp)$ *to each verifier. If it did, then P computes some output $y(x_1, \ldots, x_{d+1})$ and sends* $\mathrm{Enc}(\mathrm{sk}, y)$ *to each verifier. The prover delays these messages appropriately so that they are received in lockstep by all verifiers simultaneously.*

We claim that Protocol 2 is complete for all spacetime points where the verifiers' broadcast messages intersect. We provide a basic visual intuition in Figure 3, and it is not hard to argue intuitively that, with a message broadcast at every timestep there is a fine mesh of committable points induced over the convex hull of the verifiers' positions. The arguments for hiding and binding of this commitment follow essentially in just the same way as Lemma 3.3 – the verifiers' view is always a constant stream of encrypted messages, and the prover is bound by its commitment to the secret key and by the security of the underlying position verification scheme. As for the optimization: by observing the algorithm for honest parties, it is clear that the *per-timestep* computational load on all parties in this protocol is a quantity which does not grow with the number of committable points $|S|$, and is polynomial in the security parameter.

4 Zero-knowledge Position Verification

As outlined in Section 1.2, the main goal of this paper is to construct secure position proofs which are zero-knowledge over some nontrivial region of spacetime, in the following sense:

Definition 4.1 (Zero-Knowledge Property for Position Verification)**.** *Let $\Pi = (P, V, R)$ be a position verification protocol, where P is a family $\{P_\alpha\}_{\alpha \in R}$ of honest prover algorithms. We say that Π has the* ***honest-verifier, computational zero-knowledge*** *property over R if the following is true. There exists a QPT simulator* Sim *such that for any $\alpha \in R$, and any time τ,*

$$\left\{\mathrm{Sim}(1^\lambda, \tau)\right\}_{\lambda \in \mathbb{N}} \approx_c \left\{\mathrm{View}_V^\tau(P_\alpha \leftrightarrows V)(1^\lambda)\right\}_{\lambda \in \mathbb{N}}$$

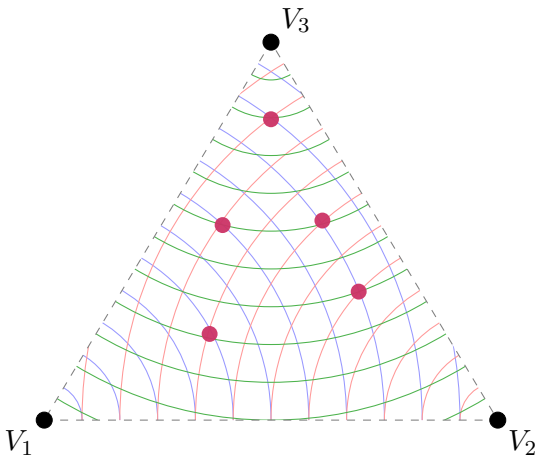

Figure 3: Illustration of the optimized position commitments in Protocol 2 for $d = 2$, with intersection points highlighted in red. When these arcs are as dense as one per timestep, the set of intersection points will be a fine-grained mesh.

We notice that the zero-knowledge property is very similar to the hiding property of position commitments. Naturally, we will see that we can construct zero-knowledge position proofs by proving statements about the potential reveal outcomes of a position commitment.

Lemma 4.2. *Suppose $\mathcal{C} = (P, V, S)$ is a mostly-classical position commitment scheme with completeness error $c(\lambda)$, statistical position binding $s(\lambda)$ against some prover class $\mathscr{C}$, and computational honest-verifier hiding. Then, for any nonempty region $R \subset S$, there exists a position verification protocol $\Pi = (P, V, R)$ with completeness $c(\lambda)$, position security $s(\lambda) + \mathrm{negl}(\lambda)$ against $\mathscr{C}$, and honest-verifier computational zero knowledge.*

Proof. Take such an $R \subset S$, and fix any $\alpha^* \in R$. Let $P_{\mathcal{C}}$ be some honest prover for $\mathcal{C}$ located at α^*. We now describe the zero-knowledge position verification protocol, by defining the behavior of the honest prover $P = P_{\alpha^*}$ and the honest verifiers V.

P and V first run $\rho \leftarrow \mathrm{Commit}_{\mathrm{P}\mathcal{C}\leftrightarrows\mathrm{V}}(1^\lambda)$, generating a classical commitment state which V then sends to the prover. P will then initiate a zero-knowledge proof that ρ belongs to an NP language Reveal_R, which we define below.

$$\rho \in \mathsf{Reveal}_R \iff \exists(\alpha, P^*) \; s.t. \; \alpha \in R \; \wedge \; P^* \text{ is a PPT alg.} \; \wedge \; \mathrm{Reveal}_{P^*\leftrightarrows V}(1^\lambda, \rho, \alpha) \text{ accepts}$$

We know that under our assumptions, there exists a post-quantum computational zero-knowledge proof protocol $\mathrm{ZK} = (P_{ZK}, V_{ZK})$ for Reveal_R with perfect completeness and negligible soundness error. So the protocol Π continues as follows: P and V run P_{ZK} and V_{ZK}, respectively, on common input ρ. P_{ZK} is additionally given a witness $w = (\alpha^*, P_{\mathrm{Reveal}})$, where P_{Reveal} is P's code for the Reveal phase. Afterwards, V accepts if and only if V_{ZK} accepts.

We claim that Π has completeness error $c(\lambda)$: this follows from the fact that $\mathcal{C}$ has completeness error $c(\lambda)$ and ZK has perfect completeness. Next, we argue for Π's position security: take a cheating prover coalition $P^* \in \mathscr{C}$ where nobody occupies position α^*. By the statistical position binding of $\mathcal{C}$, except with probability $s(\lambda)$ over $\rho \leftarrow \mathrm{Commit}_{\mathrm{P}^*\leftrightarrows\mathrm{V}}(1^\lambda)$, we have that $\mathrm{Reveal}_{A\leftrightarrows V}(1^\lambda, \rho, \alpha)$ rejects for all points α and all algorithms A. Thus, ρ only belongs to the language Reveal_R with

probability at most $s(\lambda)$. By the soundness of ZK, this means that V accepts with probability at most $s(\lambda) + \mathrm{negl}(\lambda)$.

Finally, we describe the zero-knowledge simulator Sim. Take $\mathrm{Sim}_{\mathcal{C}}$ to be the simulator guaranteed by the computational hiding property of the commitment scheme, and $\mathrm{Sim}_{\mathrm{ZK}}$ be the simulator for the protocol ZK. On input $(1^\lambda, \tau)$, Sim will first run $\sigma = \mathrm{Sim}_{\mathcal{C}}(1^\lambda, \min(t_{\mathrm{final}}, \tau))$. If $\tau \leq t_{\mathrm{final}}$, Sim outputs σ. Otherwise, take the commitment state ρ which is contained on the verifiers' output register in σ (and let σ' be the remainder of the verifiers' state). Next, Sim runs $\pi \leftarrow \mathrm{Sim}_{\mathrm{ZK}}(1^\lambda, \rho)$ and outputs (σ', π). As we have constructed Π, and by the assumed properties of $\mathrm{Sim}_{\mathcal{C}}$ and $\mathrm{Sim}_{\mathrm{ZK}}$, the output of $\mathrm{Sim}(1^\lambda, \tau)$ is computationally indistinguishable from $\mathrm{View}_V^\tau(P \leftrightarrows V)(1^\lambda)$. □

Theorem 4.3 (Main Result)**.** *Suppose there exist post-quantum one-way functions, and a nice position verification protocol with completeness $c(\lambda)$ and position security $s(\lambda)$ against spoofers sharing at most E entangled qubits. Then, for any finite region $R \subset \mathbb{R}^d \times \mathbb{R}$, there exists a position verification protocol $\Pi = (P, V, R)$ with completeness $c(\lambda)$, position security $|S|(s(\lambda) + \mathrm{negl}(\lambda))$ against spoofers sharing $E/2$ entangled qubits, and honest-verifier computational zero knowledge.*

Proof. Let $\mathcal{C} = (P, V, S)$ be the encrypt-then-verify position commitment scheme outlined in Section 3.1, instantiated with the above assumptions. Then, the existence of Π as claimed, follows from Lemma 4.2. □

5 Towards Malicious-Verifier Zero Knowledge

We have focused so far on the honest-verifier setting. We feel that this is a reasonable assumption in some settings, such as for example when the verifiers are cell towers; malicious behavior would likely be detectable by, or significantly impact quality of service to, a large customer base. However, in full generality, it would clearly be desirable to achieve zero-knowledge even against malicious verifiers. With this in mind, we will give some intuition for why the honest-verifier assumption may be difficult to remove, and then give some potential directions for relaxing it.

A General Attack on Privacy. If we naively try to work within a universe with malicious verifiers, we run into a simple, but annoyingly effective, "poke-around-in-the-dark" attack on privacy. This attack, for any positional protocol Π with good completeness, goes as follows: the verifiers *only* send messages along a particular ray $\mathbf{z} \in \mathbb{R}^d$, and then see whether they accept or reject at the conclusion of Π. The protocol will accept with high probability if and only if the honest prover is located along $\mathbf{z}$! Equivalently, we could imagine that they send messages *everywhere but* a particular direction, and now they will only reject if the prover's position was in that direction. The major obstacle in defending against these attacks in our model is that a party can only see messages which pass through their location; thus, no honest party can tell whether or not messages were sent directionally to other locations (much like the proverbial tree falling in a forest with no one around to hear it).

Thus we claim (informally) that for any protocol $\Pi = (P, V, R)$ where in order for completeness to hold, V must send directional messages to every point in R, the protocol cannot be zero-knowledge against malicious verifiers. This is due to the attack described above, where V can "deny service" to some zone of R, and learn whether the prover was in this zone based on the eventual decision to accept/reject.

Possible Directions As demonstrated above, it is clear that we cannot hope to achieve malicious-verifier privacy when the protocol hinges on the verifiers' use of directional messages. One might note, however, that our optimized protocol in Section 3.3 utilizes verifier messages which are sent using entirely classical, broadcast signals. Note that the prover, however, still has to send directional messages to the verifiers, in order to mask their position. However, if we can reasonably assume a model where verifiers only send broadcast messages, but provers can send either directional or broadcast messages, then we could hope to prove zero-knowledge against malicious verifiers. The broadcast-only would be reasonable to assume if, say, 1. the verifiers are low-tech cell towers without directional capability, or even if 2. the verifiers are arbitrary, but the prover can *differentiate* whether a signal it received was directional or broadcast – if the prover simply ignores all directional messages, then we essentially collapse to the broadcast-only model.

Another interesting area of investigation is to augment the model with more verifiers. We could consider having a redundant number of verifiers in each protocol, i.e. a considerable surplus to the $d + 1$ which are normally required, and use an honest-majority assumption to see what privacy guarantees can be achieved. To motivate this model, we can again use the example of cell towers and observe that points in densely populated areas are likely to be in range of dozens of cell towers at once – much more than the four that are required to perform 3-D position verification. We could imagine that with this surplus, the prover is free to randomize the choice of verifiers which it actually listens to, and thus with high probability decouple the accept/reject decision from the actions of any given verifier (or any minority coalition of dishonest verifiers). We note that in this model there is also an interesting twist to be considered – since each verifier is surrounded by other verifiers, we could ask it to play the role of *a prover*, and establish some property of its position to the rest of the network. An even simpler observation about this model is that the surrounding verifiers can audit the timings of its broadcast messages, which could be useful in the aforementioned discussion about achieving malicious-verifier zero-knowledge in the broadcast-only model.

Acknowledgments

Diagrams in this paper were made with the help of generative AI. HY is supported by AFOSR award FA9550-23-1-0363, NSF awards CCF-2530159, CCF-2144219, and CCF-2329939, and by the Sloan Foundation.

References

[59] *The Antarctic Treaty*. 402 U.N.T.S. 71; 12 U.S.T. 794; TIAS No. 4780; 19 I.L.M. 860 (1980). Opened for signature Dec. 1, 1959; entered into force June 23, 1961. Dec. 1959. URL: https://www.ats.aq/e/antarctictreaty.html (cit. on p. 2).

[ABM+24] Rene Allerstorfer, Harry Buhrman, Alex May, Florian Speelman, and Philip Verduyn Lunel. "Relating non-local quantum computation to information theoretic cryptography". In: *Quantum* 8 (2024), p. 1387 (cit. on p. 8).

[ACM25] Vahid R. Asadi, Eric Culf, and Alex May. *Rank lower bounds on non-local quantum computation*. 2025. arXiv: 2402.18647 [quant-ph]. URL: https://arxiv.org/abs/2402.18647 (cit. on p. 8).

[AKL+25] Vahid R. Asadi, Kohdai Kuroiwa, Debbie Leung, Alex May, Sabrina Pasterski, and Chris Waddell. "Conditional disclosure of secrets with quantum resources". In: *Quantum* 9 (Oct. 2025), p. 1885. ISSN: 2521-327X. DOI: 10.22331/q-2025-10-16-1885. URL: http://dx.doi.org/10.22331/q-2025-10-16-1885 (cit. on p. 8).

[BCF+14] Harry Buhrman, Nishanth Chandran, Serge Fehr, Ran Gelles, Vipul Goyal, Rafail Ostrovsky, and Christian Schaffner. "Position-based quantum cryptography: Impossibility and constructions". In: *SIAM Journal on Computing* 43.1 (2014), pp. 150–178 (cit. on pp. 1–3, 7, 9).

[BCS22] Andreas Bluhm, Matthias Christandl, and Florian Speelman. "A single-qubit position verification protocol that is secure against multi-qubit attacks". In: *Nature Physics* 18.6 (2022), pp. 623–626 (cit. on pp. 3, 4, 7, 15).

[BK11] Salman Beigi and Robert König. "Simplified instantaneous non-local quantum computation with applications to position-based cryptography". In: *New Journal of Physics* 13.9 (2011), p. 093036 (cit. on pp. 3, 7).

[BS23] Shalev Ben-David and Or Sattath. "Quantum Tokens for Digital Signatures". In: *Quantum* 7 (Jan. 2023), p. 901. ISSN: 2521-327X. DOI: 10.22331/q-2023-01-19-901. URL: http://dx.doi.org/10.22331/q-2023-01-19-901 (cit. on p. 9).

[CGMO09] Nishanth Chandran, Vipul Goyal, Ryan Moriarty, and Rafail Ostrovsky. "Position based cryptography". In: *Annual International Cryptology Conference*. Springer. 2009, pp. 391–407 (cit. on pp. 1, 2, 9).

[ES25] Llorenç Escolà-Farràs and Florian Speelman. *Quantum position verification in one shot: parallel repetition of the f-BB84 and f-routing protocols*. 2025. arXiv: 2503.09544 [quant-ph]. URL: https://arxiv.org/abs/2503.09544 (cit. on pp. 7, 15).

[GMOW25] Uma Girish, Alex May, Leo Orshansky, and Chris Waddell. *Comparing classical and quantum conditional disclosure of secrets*. 2025. arXiv: 2505.02939 [quant-ph]. URL: https://arxiv.org/abs/2505.02939 (cit. on p. 8).

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. "Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems". In: *J. ACM* 38.3 (July 1991), pp. 690–728. ISSN: 0004-5411. DOI: 10.1145/116825.116852. URL: https://doi.org/10.1145/116825.116852 (cit. on p. 13).

[GW98] Eran Gabber and Avishai Wool. "How to prove where you are: Tracking the location of customer equipment". In: *Proceedings of the 5th ACM Conference on Computer and Communications Security*. 1998, pp. 142–149 (cit. on p. 2).

[HM16] Patrick Hayden and Alex May. "Summoning information in spacetime, or where and when can a qubit be?" In: *Journal of Physics A: Mathematical and Theoretical* 49.17 (Mar. 2016), p. 175304. ISSN: 1751-8121. DOI: 10.1088/1751-8113/49/17/175304. URL: http://dx.doi.org/10.1088/1751-8113/49/17/175304 (cit. on p. 9).

[JKPP22] Marius Junge, Aleksander M Kubicki, Carlos Palazuelos, and David Pérez-García. "Geometry of Banach spaces: a new route towards Position Based Cryptography". In: *Communications in Mathematical Physics* 394.2 (2022), pp. 625–678 (cit. on p. 8).

[KL14] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography, Second Edition*. 2nd. Chapman & Hall/CRC, 2014. ISBN: 1466570261 (cit. on p. 13).

[KLC+24] Fatih Kaleoglu, Minzhao Liu, Kaushik Chakraborty, David Cui, Omar Amer, Marco Pistoia, and Charles Lim. *On the Equivalence between Classical Position Verification and Certified Randomness*. Cryptology ePrint Archive, Paper 2024/1726. 2024. URL: https://eprint.iacr.org/2024/1726 (cit. on p. 8).

[KMS11] Adrian Kent, William J Munro, and Timothy P Spiller. "Quantum tagging: Authenticating location via quantum information and relativistic signaling constraints". In: *Physical Review A—Atomic, Molecular, and Optical Physics* 84.1 (2011), p. 012326 (cit. on pp. 1, 2, 7).

[LLQ22] Jiahui Liu, Qipeng Liu, and Luowen Qian. "Beating Classical Impossibility of Position Verification". In: *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. 2022, pp. 100–1 (cit. on pp. 3, 5, 7, 8, 15, 20).

[Mal16] Robert Malaney. "The quantum car". In: *IEEE Wireless Communications Letters* 5.6 (2016), pp. 624–627 (cit. on p. 2).

[May19] Alex May. "Quantum tasks in holography". In: *Journal of High Energy Physics* 2019.10 (2019), pp. 1–39 (cit. on p. 8).

[Nao91] Moni Naor. "Bit commitment using pseudorandomness". In: *J. Cryptol.* 4.2 (Jan. 1991), pp. 151–158. ISSN: 0933-2790. DOI: 10.1007/BF00196774. URL: https://doi.org/10.1007/BF00196774 (cit. on p. 13).

[Pre25] Associated Press. *Biden, Trump and Macron bodyguards foil possible Strava security risk*. Accessed: 2025-09-27. Sept. 2025. URL: https://apnews.com/article/biden-trump-macron-bodyguards-security-strava-0a48afca09c7aa74d703e72833dcaf72 (cit. on p. 2).

[Unr14] Dominique Unruh. "Quantum position verification in the random oracle model". In: *Annual Cryptology Conference*. Springer. 2014, pp. 1–18 (cit. on pp. 3, 5, 7, 15).

[Vai03] Lev Vaidman. "Instantaneous measurement of nonlocal variables". In: *Physical review letters* 90.1 (2003), p. 010402 (cit. on pp. 3, 7).

[Yan22] Jun Yan. "General Properties of Quantum Bit Commitments (Extended Abstract)". In: *Advances in Cryptology – ASIACRYPT 2022*. Ed. by Shweta Agrawal and Dongdai Lin. Cham: Springer Nature Switzerland, 2022, pp. 628–657. ISBN: 978-3-031-22972-5 (cit. on p. 20).