Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Security evaluation of quantum distance-bounding protocols via semidefinite programming

Kevin Bogner

kevin.bogner@kuleuven.be

KU Leuven

Aysajan Abidin

KU Leuven

Dave Singelée

KU Leuven

Bart Preneel

KU Leuven

Article

Keywords:

Posted Date: September 15th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10377329/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Security evaluation of quantum distance-bounding protocols via semidefinite programming

Kevin Bogner [1,*], Aysajan Abidin [1], Dave Singelée [2], Bart Preneel [1]

[1]COSIC, KU Leuven, Leuven, Belgium
[2]DistriNet, KU Leuven, Leuven, Belgium
*Corresponding author: kevin.bogner@kuleuven.be

Abstract

Quantum distance-bounding (QDB) protocols use a timed fast phase of quantum communication to check that a prover is genuine and physically nearby. Comparing protocols is difficult because optimal attacks depend on protocol structure, so we isolate the fast phase and study one-round distance-fraud and mafia-fraud games. For discrete-variable QDB, semidefinite programs represent every allowed strategy in these games; each mafia-fraud value comes with an attack achieving it and a certificate that no attack does better. This contrasts with quantum position verification, where the attack splits between two separated parties and the optimization is nonconvex. The best one-round distance-fraud attack succeeds with probability 1/2 for all discrete-variable protocols studied, while mafia-fraud values differ; two of the four benchmarked protocols receive their first one-round results, and the other two receive stronger attacks than previously reported. For continuous-variable QDB, we report estimates from a calibrated Gaussian attack model.

1 Introduction

Distance-bounding (DB) protocols let a verifier check that a prover knows the right secret key and that it is physically nearby [BC94, HK05]. Think of a car (the verifier) checking that the genuine key fob (the prover) is actually near the car, and not in the owner's home while an attacker relays its signals. To this end, a DB protocol combines untimed setup messages (the slow phase) with timed challenge-response rounds (the fast phase). The round-trip time of each response limits how far away the prover can be. In quantum distance bounding (QDB), the fast phase uses quantum communication, so the prover must receive, process, and answer fresh quantum data within a strict time budget. This has led to a growing family of QDB protocols, including prepare-and-measure, entanglement-based, and continuous-variable QDB (CV-QDB) constructions [AMSP17, Abi19, AES25, BSA24, BAS25].

As in classical DB, the main attack classes are distance fraud (DF), mafia fraud (MF), and terrorist fraud (TF). In distance fraud, a dishonest prover that holds a valid key but sits outside the allowed distance tries to appear closer than it is. In mafia fraud, an attacker without the key sits between an honest verifier and an honest prover and relays or manipulates their messages; the relay attack on keyless car entry sketched above is the classic example. In terrorist fraud, a dishonest prover helps a nearby accomplice make the verifier accept, without handing over its long-term key.

A central difficulty, made explicit by the QDB security framework of [BASP26], is that QDB protocols are not easy to compare on common ground. Their fast phases reveal different information at different times, so the optimal attack can change substantially from one design to another. Consequently, much of the literature focuses on protocol-specific attacks, which are useful within a single protocol but do not give a uniform comparison across QDB variants.

We close this gap by using the QDB security framework of [BASP26], which formalizes the DF and MF games and the attacker's capabilities. From it, we derive one-round fast-phase games and optimize them via semidefinite programming (SDP). Throughout the paper, the *value* of such a game is the highest success probability that any allowed attacker strategy can achieve in that single round. For discrete-variable QDB (DV-QDB) protocols, these games have exact SDP representations; for Gaussian CV-QDB we use a calibrated Gaussian estimator.

This exact solvability is the main conceptual point of the paper, and it is best seen against quantum position verification (QPV) [KMS11, BCF$^+$14], the closest well-studied setting, in which verifiers use timed quantum messages to check a prover's location. A QPV attack is mounted by two separated parties that intercept the verifiers' messages and must coordinate under timing constraints. Optimizing such a split attack is in general a nonconvex problem, so QPV analyses rely on relaxations matched by explicit attacks. The QDB MF adversary is also a cooperating pair, but in the one-round game the pair collapses to a single sequential strategy: first query the honest prover, then answer the verifier's challenge (Section 2.3). The resulting strategy set has an exact comb-SDP representation, so the SDP optimum equals the one-round game value rather than a relaxation bound. The DV-QDB games studied here thus offer a position-verification-like setting in which one-round attack values admit exact convex formulations.

For DV-QDB, the reduced DF and MF games become SDPs. In the DF game, the attacker simply prepares a quantum state, which gives a state SDP. In the MF game, the attacker acts in two steps: it first queries the honest prover before the timed challenge arrives (a *pre-ask*) and then answers the verifier's fresh challenge; such two-step strategies are described by two-slot combs; this gives a comb SDP. Both strategy sets have exact SDP representations, so the SDP optima equal the one-round game values rather than relaxation bounds. Each MF value is certified by a matching primal and dual pair in exact arithmetic: an explicit attack together with a matching upper-bound certificate of the same value (Section 4.4). For Gaussian CV-QDB, exact SDPs are out of reach; we instead report success-probability estimates from a calibrated Gaussian estimator. Both sets of results are summarized in Table 1.

DB was introduced by Brands and Chaum [BC94] and later adapted into practical designs for radio-frequency identification applications [HK05]. Later work introduced the standard attack classes DF, MF, and TF, together with a large body of protocol designs and analyses; a convenient overview is given in [ABB$^+$18]. Modeling TF is harder than DF or MF, and the literature has yet to agree on a single definition of TF resistance [FO13, Vau13].

Early QDB work proposed a qubit-based fast phase with a final classical MAC; we call this protocol QDB 2017 [AMSP17]. QDB 2019 [Abi19] refined this by authenticating the prover through its quantum measurement-and-preparation response, removing the MAC. This design difference sets the scope of our benchmark (Section 2.4). More recent work extended QDB to entanglement-based and continuous-variable settings: a mutual entanglement-based protocol we call Mutual QDB [AES25], an E91-inspired variant we call E91 QDB [BSA24], and a continuous-variable protocol we call CV-QDB [BAS25].

The work methodologically closest to ours is a cryptanalysis of early QDB protocols [Ver22], which sharpened several attack analyses, studied TF countermeasures and their limitations, and used SDP to optimize measurements in protocol-specific attacks. We extend this idea to a different scope: rather than applying SDP inside individual attacks, we use it to capture the entire one-round strategy space. The analysis rests on two foundations. First, the QDB security framework of [BASP26] formalizes the attacker's capabilities. Second, from [MSTPTR18] we take the causal viewpoint that a party's response can only depend on information that has had time to physically reach it before it answers. Instead of tracking explicit times and locations, we therefore only track which information is available to each party at the moment it must respond. From these, we derive one-round DF and MF games that we solve exactly for the DV-QDB protocols and estimate for CV-QDB.

The closest related setting is QPV, in which verifiers use timed quantum messages to check a prover's location [KMS11, BCF$^+$14]. Unconditional QPV is impossible against attackers that share unlimited entanglement [BCF$^+$14], so QPV security is studied against bounded attackers. One-round QPV games have been studied in depth. The monogamy-of-entanglement game of [TFKW13] gives closed-form attack values for BB84-style QPV when the attackers share no entanglement beforehand, [BCS22] analyzes single-qubit QPV under multi-qubit attacks, and [EFS23] bounds attack values for lossy QPV with SDP relaxations that are matched by explicit attacks. Continuous-variable QPV has also been analyzed [AEFR$^+$23, EFRA$^+$24]. QDB differs from QPV in two ways that change the attack model. First, the QDB prover authenticates itself with a shared secret key, and the hidden protocol variables in our games come from that key material. Second, the QDB MF adversaries may query the honest prover before the timed challenge (the pre-ask), and in the one-round game the cooperating pair acts as a single sequential strategy (Section 2.3), whereas a QPV attack is mounted by two separated parties that intercept the verifiers' messages and must coordinate under timing constraints using pre-shared entanglement.

2 Results

2.1 One-round fast-phase games

We use the security framework of Section 4.1 to determine what the adversary can do in the one-round DF and MF games, in particular what information it may use and when. Following [MSTPTR18], we track this information rather than explicit times and locations. For DV-QDB, these games reduce to exact SDPs. The DF game becomes a state SDP, since the distant prover only prepares a quantum response state. The SDP finds the state that maximizes the verifier's acceptance probability. The MF adversary, by contrast, is a cooperating pair that we treat as a single two-step strategy. It pre-asks the honest prover, keeps the returned quantum system in memory, then processes it together with the verifier's challenge to produce its response. This strategy is a two-slot comb, so the MF game becomes a comb SDP that maximizes the verifier's acceptance probability.

For the DV reductions that follow, we describe a one-round game instance by a finite set of branches Ω, each equally likely, so a branch has weight $1/|\Omega|$. A branch $\omega \in \Omega$ is one possible setting of the round's classical variables. Crucially, some of these variables must stay hidden from the adversary. Otherwise the adversary is able to produce the honest prover's response, which the verifier always accepts by completeness (Definition 1). On branch ω, the verifier sends the challenge state ρ_ω and receives the returned state σ (honest or adversarial). The accept operator Π_ω, the verifier's check, is determined by the branch. The verifier accepts with probability $\mathrm{Tr}(\Pi_\omega \sigma)$, which measures how close σ is to the response it expects on this branch. The bound $0 \preceq \Pi_\omega \preceq I$, with I the identity matrix, is what makes this a genuine probability between 0 and 1. Without it, Π_ω would be an arbitrary matrix rather than a real measurement.

The CV-QDB protocol is handled separately in Section 2.8.

Our SDPs model only a single fast round, which is enough for the DF and MF games studied here. We do not formulate a TF SDP, because TF security is about what happens across multiple runs: whether a malicious prover's help can be reused in a fresh execution. A single-round SDP cannot capture that.

2.2 Distance-fraud reduction to SDP

We reduce the DF experiment of Definition 2 to a single round. Following [MSTPTR18], let $v(\omega)$ denote the DF response-time view, the information available to the distant prover $\mathcal{P}^\star$ at the moment it must respond. Typically the view contains the secret key and everything from the slow phase, but not the fast-phase challenge state. In distance fraud, $\mathcal{P}^\star$ lies outside the

distance bound, so it must answer before the challenge state ρ_ω could even reach it. Its response therefore cannot depend on ρ_ω, and the returned state depends only on $v(\omega)$. This restriction is formalized in [BASP26, BMV15]. The optimal DF success probability is therefore

$$p^\star_{\mathrm{DF}} = \max_{\{\sigma_v\}} \frac{1}{|\Omega|} \sum_{\omega\in\Omega} \mathrm{Tr}\big(\Pi_\omega \sigma_{v(\omega)}\big) \tag{1}$$

subject to

$$\sigma_v \succeq 0, \qquad \mathrm{Tr}(\sigma_v) = 1 \quad \text{for every view } v.$$

Here $\{\sigma_v\}$ is the prover's strategy, with σ_v the state it returns when it has view v. The two constraints make every σ_v a valid quantum state, positive semidefinite and with trace one, so the prover could actually prepare it. On branch ω, the prover has view $v = v(\omega)$ and returns $\sigma_{v(\omega)}$, which the verifier accepts with probability $\mathrm{Tr}(\Pi_\omega \sigma_{v(\omega)})$. Averaging this over all branches and choosing the best response states gives the optimal DF success probability $p^\star_{\mathrm{DF}}$.

Lemma 1 (Reduced distance-fraud value)**.** *For any reduced one-round DF instance of a DV-QDB protocol in the model above, the optimal DF value is given by Equation* (1)*.*

Proof. Fix a response-time view v. By the timing restriction, the returned system cannot depend on the challenge state ρ_ω. It is therefore described by some density operator σ_v. On branch ω, the verifier accepts the returned state with probability $\mathrm{Tr}(\Pi_\omega \sigma_{v(\omega)})$; averaging these acceptance probabilities over the equally likely branches gives the attack's success probability, which is exactly the objective of Equation (1). Conversely, any feasible family $\{\sigma_v\}$ is physically realizable, since on observing view v the prover can prepare σ_v. Optimizing over feasible density operators therefore yields the optimal DF value. □

2.3 Mafia-fraud reduction to SDP

We reduce the MF experiment of Definition 3 to a single round. In that experiment, the adversary is the cooperating pair $(\mathcal{A}_1, \mathcal{A}_2)$, located near the verifier and prover, respectively. In the one-round SDP game we replace the pair $(\mathcal{A}_1, \mathcal{A}_2)$ by a single adversary S that acts in two steps and uses the same strategy on every branch. Formally, S is a branch-independent two-slot comb.

The first step is the pre-ask, the learning phase of Definition 3. The adversary chooses a probe system Q and sends it to the honest prover, which returns a response R. This reply maps the input Q to the output R, so we describe it by a quantum channel

$$\mathcal{N}_\omega : \mathcal{L}(Q) \to \mathcal{L}(R),$$

the most general physical process taking one quantum system to another; here $\mathcal{L}(Q)$ denotes the matrices acting on register Q. The channel is branch-dependent because the honest prover uses protocol variables that differ from branch to branch. The adversary, by contrast, does not know ω, so the same comb S is used on every branch. The second step mirrors the challenge session of Definition 3. The verifier sends the challenge state ρ_ω on a register C (a system carrying one message), and the adversary returns its output on a register O. This output depends on the challenge ρ_ω together with the pre-ask systems Q and R. The comb S is defined on the fixed registers Q, R, C, O, and the branch ω selects which challenge state arrives on C. Figure 1 shows the comb strategy S and its interactions with the prover and verifier.

To turn the game into an SDP, we now put the comb S and the channels in matrix form, using the unnormalized Choi convention

$$J(\mathcal{E}) = \sum_{i,j} |i\rangle\langle j| \otimes \mathcal{E}(|i\rangle\langle j|)$$

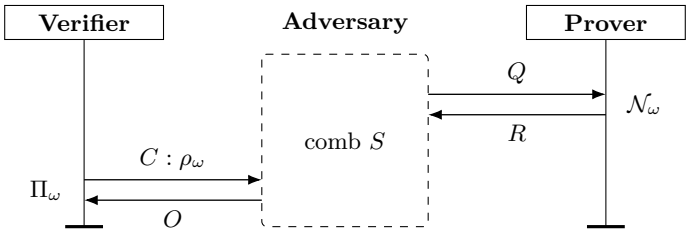


Figure 1: Reduced mafia-fraud SDP game. The adversary's comb S first interacts with the prover via the pre-ask channel $\mathcal{N}_\omega$, then answers the verifier's challenge ρ_ω. The branch-dependent triple $(\mathcal{N}_\omega, \rho_\omega, \Pi_\omega)$, with Π_ω the verifier's accept operator, combines into a single tester operator W_ω that captures everything outside the adversary's control on branch ω. The acceptance probability is then $\mathrm{Tr}(W_\omega S)$, a linear function of the comb S; this linearity is what makes the optimization an SDP.

with respect to the input computational basis. This represents every quantum channel as a single positive semidefinite matrix on the tensor product of its input and output spaces, and it turns the comb S into an operator on $Q \otimes R \otimes C \otimes O$. The MF acceptance probability then becomes a linear function of S, which is exactly what an SDP optimizes. The comb-normalization constraints on S are imposed in the SDP below.

For each branch ω, the three branch-dependent objects $(\mathcal{N}_\omega, \rho_\omega, \Pi_\omega)$ combine into a single tester operator W_ω. The tester is the protocol-side counterpart of the adversary's comb. It packages everything outside the adversary's control on branch ω, and pairs with S via the trace so that $\mathrm{Tr}(W_\omega S)$ is exactly the acceptance probability. With the Choi convention above, the tester reads

$$W_\omega = J(\mathcal{N}_\omega)^{\mathsf{T}}_{QR} \otimes (\rho_\omega)^{\mathsf{T}}_{C} \otimes \Pi_{\omega,O},$$

where transposes are taken in the computational basis of each register.

Thus, the optimal MF success probability is

$$p^\star_{\mathrm{MF}} = \max_{S,\eta} \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \mathrm{Tr}(W_\omega S), \tag{2}$$

subject to

$$S \succeq 0, \qquad \mathrm{Tr}_O(S) = \eta \otimes I_{RC}, \qquad \eta \succeq 0, \qquad \mathrm{Tr}(\eta) = 1.$$

Here η is the probe quantum state that the adversary prepares on register Q and sends to the honest prover in the pre-ask interaction, and Tr_O is the partial trace over O: it discards register O and returns the state of the remaining registers. The constraints above are the standard comb normalization conditions; they ensure that S describes a physically realizable sequential strategy in the order Q, R, C, O.

To prove each MF value, we use a companion dual SDP whose feasible points upper-bound $p^\star_{\mathrm{MF}}$. For every reported instance, explicit primal and dual witnesses have the same algebraic objective value, which pins $p^\star_{\mathrm{MF}}$ to that value. Their feasibility and objective equality are rechecked in exact arithmetic, so the proof does not rely on the solver. Details are in Section 4.4.

Lemma 2 (Reduced mafia-fraud value)**.** *For any reduced one-round MF instance of a DV-QDB protocol in the model above, the optimal MF value is given by Equation* (2)*.*

Proof. An MF adversary first prepares a probe, receives the honest-prover reply, later receives the verifier challenge, and then outputs its response. Such sequential strategies are exactly deterministic combs on the registers Q, R, C, O satisfying the constraints in Equation (2). For each branch ω, the honest-prover channel, challenge state, and accept operator combine into the tester W_ω, and pairing it with the comb S gives the branch acceptance probability $\mathrm{Tr}(W_\omega S)$. The reverse also holds. Every feasible S corresponds to a real adversary strategy [GW07, CDP09]. Averaging over branches gives Equation (2). □

Table 1: Results for the reduced one-round fast-phase games under the assumptions of Section 4.3; for the DV-QDB protocols, each entry is the best success probability of the corresponding one-round attack. The prior mafia-fraud attack column lists previously reported per-round fast-phase attack values when a directly comparable value is available. – marks entries with no comparable prior per-round value; * marks values exceeding previously reported per-round fast-phase attacks; † marks values not previously reported to our knowledge. The displayed DV-QDB decimals are rounded from exact one-round values computed via SDP; the MF values are certified by matching primal and dual witnesses in exact arithmetic (Section 4.4). The CV-QDB entries assume zero added response noise; the DF entry is the exact optimum within the restricted Gaussian model, and the MF entry is a finite-grid Gaussian estimate. QDB 2017 [AMSP17], whose fast phase is bound by a final MAC, falls outside this benchmark and is treated in Section 4.5.

Protocol	Distance-fraud result	Prior mafia-fraud attack	Mafia-fraud result
Discrete-variable QDB protocols			
QDB 2019 [Abi19]	0.5000	0.875 [Ver22]	0.9045^{*}
Mutual QDB [AES25]	0.5000	0.625 [AES25]	0.7500^{*}
E91 QDB [BSA24]	$0.5000^{\dagger}$	–	$0.9332^{\dagger}$
Continuous-variable QDB protocol			
CV-QDB [BAS25]	$0.3592^{\dagger}$	–	$0.9138^{\dagger}$

2.4 Protocol instantiations and results

We now instantiate the one-round DF and MF SDPs of Section 2.1 for each QDB protocol. We treat the prepare-and-measure protocol QDB 2019 [Abi19], the entanglement-based protocols (Mutual QDB [AES25], E91 QDB [BSA24]), and the CV protocol [BAS25]. We list only the SDP inputs; each protocol subsection shows its protocol-flow diagram. For each DV-QDB protocol, Table 2 lists the branch set, the DF response-time view, the verifier challenge state, the acceptance operator, the honest-prover pre-ask channel, and the resulting one-round values. These values should be read under the assumptions stated in the Methods (Section 4.3).

One early protocol is deliberately absent from this benchmark. QDB 2017 [AMSP17] follows the classical design of Brands and Chaum [BC94]: its timed fast phase only checks that the prover is close, and a final MAC over the fast-phase transcript checks that it is genuine. The one-round game strips that MAC, and with it the very mechanism that provides MF resistance, so the resulting value ($p^{\star}_{\mathrm{MF}} = 1$) reflects the missing MAC rather than the protocol. The same holds for any QDB protocol that binds its fast phase with a later authenticated message. The benchmark therefore covers the protocols whose fast phase itself authenticates the prover; Section 4.5 details the QDB 2017 reduction and its fast-phase-only values.

Throughout the protocol subsections, we use the following notation:

- a, b denote fast-phase preparation or measurement bases, c a challenge bit, m the verifier's measurement outcome in the entanglement-based and CV protocols, and r a masking value.
- The accept operator Π_ω and channel $\mathcal{N}_\omega$ carry the full branch ω as a subscript. For each protocol we instead index them by the relevant variables, writing for instance $\mathcal{N}_a$ if the channel depends only on a, and $\mathcal{N}_{a,b}$ if it depends on both a and b.
- For the BB84-style prepare-and-measure bases, we write $|u\rangle_a := H^a|u\rangle$ for $u, a \in \{0, 1\}$, where H is the Hadamard transform. Thus $a = 0$ gives the computational basis and $a = 1$ the Hadamard basis.
- For E91 QDB, the verifier and prover settings a_v and a_p each select one of three measurement bases (X, W, Z for the verifier and W, Z, V for the prover, defined in the E91 QDB subsection below), rather than a BB84 basis. The response basis b is still BB84-style.

Table 2: DV-QDB reduced one-round SDP instantiations. All bit variables are uniformly distributed, and all sums over the prover's measurement outcome t are over $\{0,1\}$. In the $\rho_\omega \to \Pi_\omega$ column, a ket $|\psi\rangle$ stands for $|\psi\rangle\langle\psi|$ (a pure state on the left of the arrow, an accept projector on the right). For E91 QDB, the row covers only the value-check pairs $(a_v, a_p) \in \{(1,0),(2,1)\}$; the other settings are used for CHSH testing or left unchecked.

Protocol	ω; $v(\omega)$	$\rho_\omega \to \Pi_\omega$	$\mathcal{N}_\omega(\rho)$	$(p^\star_{\mathrm{DF}}, p^\star_{\mathrm{MF}})$
QDB 2019	(a,b,c); (a,b)	$\lvert c\rangle_a \to \lvert c\rangle_b$	$\mathcal{N}_{a,b}(\rho) = \sum_t \langle t\rvert_a \rho \lvert t\rangle_a \lvert t\rangle_b\langle t\rvert$	(0.5000, 0.9045)
Mutual QDB	(a,b,r,m); (a,b,r)	$\lvert\chi_{a,m}\rangle \to \lvert m\oplus r\rangle_b$	$\mathcal{N}_{a,b,r}(\rho) = \sum_t \langle t\rvert_a \rho \lvert t\rangle_a \lvert t\oplus r\rangle_b\langle t\oplus r\rvert$	(0.5000, 0.7500)
E91 QDB	(a_v,a_p,b,m); (a_p,b)	$\lvert\chi_{a_v,m}\rangle \to \lvert\tilde{m}\rangle_b$	$\mathcal{N}_{a_p,b}(\rho) = \sum_t \langle t\rvert_{a_p} \rho \lvert t\rangle_{a_p} \lvert t\rangle_b\langle t\rvert$	(0.5000, 0.9332)

- For the entanglement-based protocols, $|\chi_{a,m}\rangle$ is the reduced-game challenge state. When the verifier measures its half of the entangled pair in basis a and gets outcome m, the prover's half collapses to $|\chi_{a,m}\rangle$. For the perfectly correlated $|\Phi^+\rangle$ pairs of Mutual QDB, this is $|m\rangle_a$. For the perfectly anti-correlated singlet pairs of E91 QDB, it is the a_v-basis state orthogonal to $|m\rangle_{a_v}$.

Two patterns stand out from the DV-QDB values in Table 2. First, the DF value is 1/2 for every DV-QDB protocol considered here, so DF does not distinguish them. Second, MF provides the more meaningful comparison. A protocol has a high MF value when the adversary can extract information from the pre-ask that helps it answer the verifier's later challenge, and a lower value when it cannot. The DV MF values are SDP-certified by matching primal (lower-bound) and dual (upper-bound) witnesses; details are in Section 4.4.

2.5 QDB 2019

QDB 2019 [Abi19], the most studied QDB protocol, uses independent bases for the verifier's challenge a and the prover's response b. The protocol flow is shown in Figure 2. The instantiation in Table 2 gives $p^\star_{\mathrm{MF}} \approx 0.9045$ (certified in Section 4.4).

The previously reported per-round attack [Ver22] attains $p^\star_{\mathrm{MF}} = 0.875$ by reducing the pre-ask phase to a classical estimate of the relation between a and b. Our SDP attack is stronger. Instead of collapsing the pre-ask to such an estimate, the adversary keeps its quantum state intact and processes it jointly with the later challenge.

2.6 Mutual QDB

Mutual QDB [AES25] is an entanglement-based protocol in which the verifier and prover authenticate each other. The verifier distributes EPR pairs, and the prover returns measurement outcomes masked by a committed value r. The protocol flow is shown in Figure 3. We analyze only the first timed return leg, since our model covers a single fast round. The second leg and the commitment opening are out of scope. The instantiation in Table 2 gives $p^\star_{\mathrm{MF}} = 0.75$.

The previously reported per-round attack [AES25] attains $p^\star_{\mathrm{MF}} = 0.625$. Our SDP attack is stronger, and its value follows from a simple basis-matching strategy. The adversary pre-asks the prover with $|\hat{m}\rangle_{\hat{a}}$ and stores the returned state. If $\hat{a} = a$, the stored response is $|\hat{m}\oplus r\rangle_b$. The later verifier challenge is the entangled-half state carrying the fresh outcome m in basis a, so the adversary can measure it in basis $\hat{a}$ to learn m. It then flips the stored response iff $m \neq \hat{m}$, using a Pauli Y operation, which flips the logical bit in both BB84 bases up to phase, and obtains $|m\oplus r\rangle_b$. If $\hat{a} \neq a$, the same procedure is uncorrelated with the fresh outcome and succeeds with probability 0.5. This gives $0.5\cdot 1 + 0.5\cdot 0.5 = 0.75$, matching the SDP value.

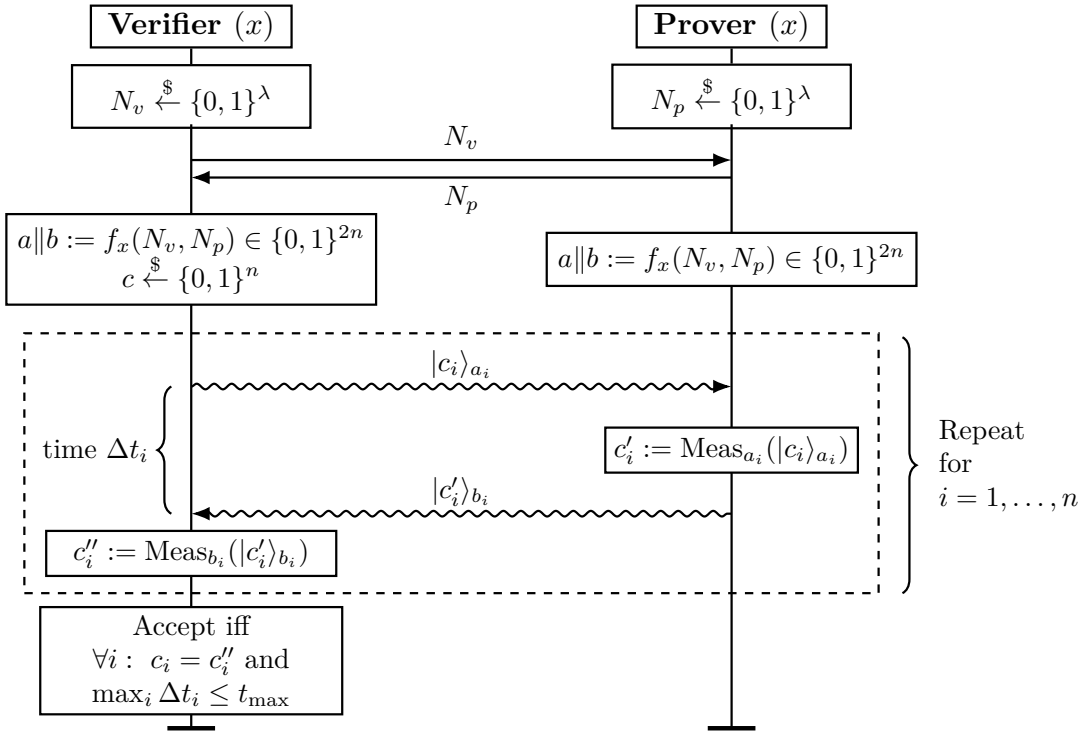


Figure 2: Overview of the QDB 2019 protocol [Abi19]. Before the fast rounds, the parties exchange nonces and derive the PRF outputs a and b. In each fast round, the verifier prepares $|c_i\rangle_{a_i}$, and the prover measures it in basis a_i and re-prepares the outcome in basis b_i. After all rounds, the verifier accepts only if all recovered bits match the originals and every round-trip time Δt_i stays below $t_{\max}$.

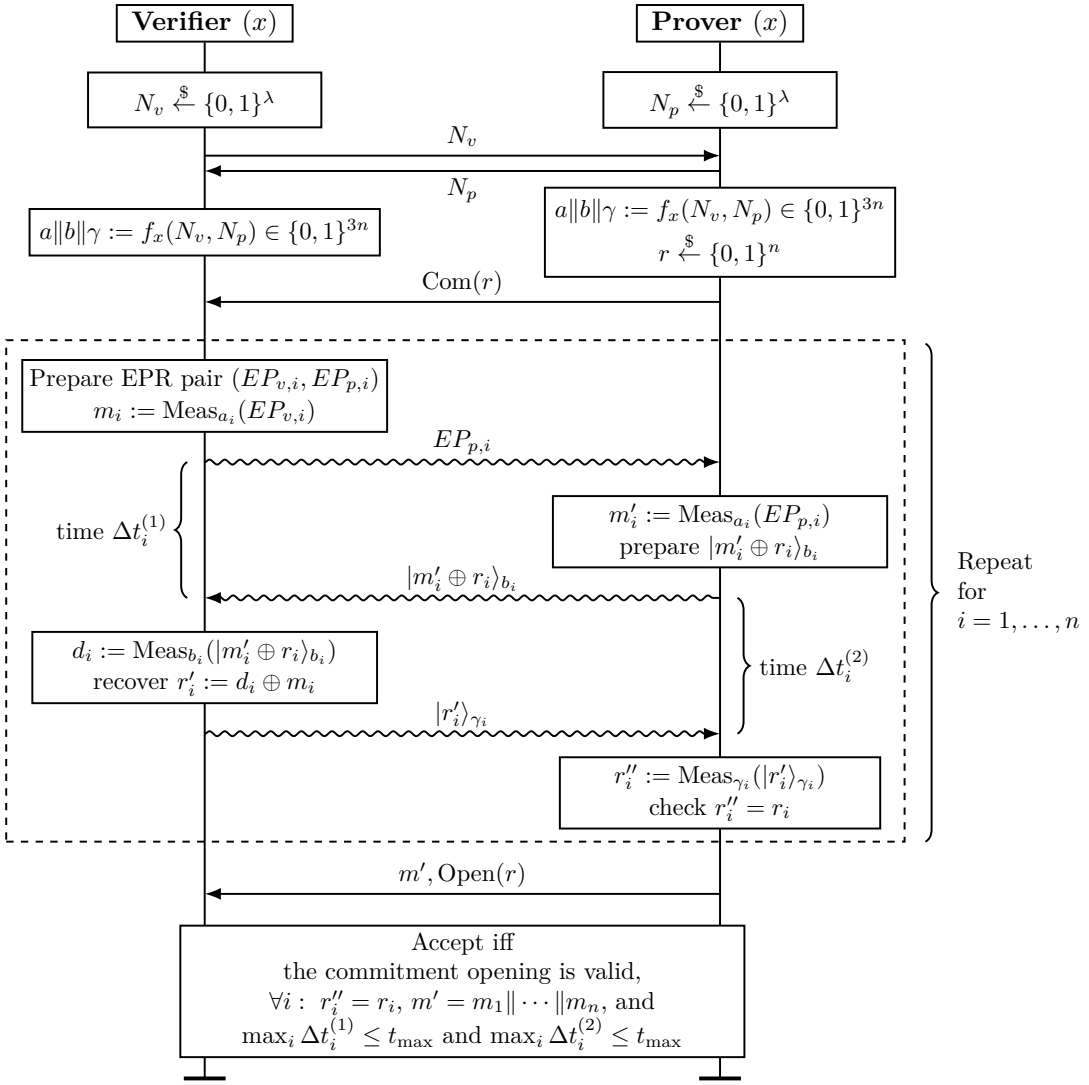


Figure 3: Overview of the Mutual QDB protocol [AES25]. Before the fast rounds, the parties exchange nonces and derive the PRF outputs a, b, and γ, and the prover commits to a random mask string r. In each fast round, the verifier prepares an EPR pair, measures its own half in basis a_i, and sends the prover half $EP_{p,i}$. The prover measures in the same basis and returns $|m'_i \oplus r_i\rangle_{b_i}$. The verifier then recovers the mask bit and sends back $|r'_i\rangle_{\gamma_i}$. After all rounds, the prover opens the commitment and reveals m'. The verifier accepts only if the commitment opening is valid, the revealed measurement string is consistent with its recorded outcomes, and every first-leg round-trip time stays below $t_{\max}$. The prover accepts only if its committed mask bits come back unchanged and every second-leg round-trip time stays below $t_{\max}$.

2.7 E91 QDB

E91 QDB [BSA24] is a Bell-inequality-based entanglement protocol. The verifier prepares an entangled pair, measures its own half in setting a_v, and sends the other half to the prover. The prover measures it in setting a_p and returns the outcome. Settings $0, 1, 2$ select the bases X, W, Z for the verifier and W, Z, V for the prover, where W and V are the eigenbases of the observables $(Z + X)/\sqrt{2}$ and $(Z - X)/\sqrt{2}$. Their Bloch-sphere directions lie at $+\pi/4$ and $-\pi/4$ from Z, so their basis vectors are rotated by $\pm\pi/8$ from the computational-basis vectors. The two matching-basis pairs $(a_v, a_p) \in \{(1, 0), (2, 1)\}$ are then exactly the rounds in which verifier and prover measure in the same basis (W and Z, respectively), and these are used for value checks. The other settings serve a CHSH non-locality test or are unchecked. The protocol flow is shown in Figure 4. We analyze only the value-check pairs. The instantiation in Table 2 gives $p^{\star}_{\mathrm{MF}} \approx 0.9332$. To our knowledge, both the DF value ($p^{\star}_{\mathrm{DF}} = 0.5$) and the MF value reported here are the first per-round values for this protocol.

For the value-check pairs, we use the singlet anti-correlation convention of the SDP instantiation. The target outcome is $\tilde{m}_i = 1 - m_i$, with m_i the verifier's measurement outcome. The earlier value-matching convention of [BSA24] sets $\tilde{m}_i = m_i$. The two conventions are equivalent up to a relabeling of the prover's classical bit.

The high MF value comes from the pre-ask interaction. The prover measures the adversary's probe in basis a_p and re-encodes the outcome in the response basis b, so the returned state carries strong information about both bases. The adversary processes this state jointly with the fresh challenge state in basis a_v to produce the accepted response.

2.8 Continuous-variable QDB

The CV-QDB protocol we consider is the Gaussian protocol of [BAS25], shown in Figure 5. Unlike the DV-QDB protocols, its acceptance test is threshold-based. The verifier accepts a round when the returned value lies within a tolerance δ_{cv} of its reference value. In the DV protocols the binary response lets a distant prover succeed at least half the time, so the DF value never drops below 0.5. Here the returned value must instead land close to a continuous target, so a distant prover often misses and the DF value can fall below 0.5. A tighter δ_{cv} lowers it further. An exact optimum is out of reach here. The CV state space is infinite-dimensional, so the attack optimization does not reduce to a finite-dimensional SDP as it does for the DV protocols. We therefore restrict to Gaussian attacks and report the estimators $p^{\mathrm{CV}}_{\mathrm{DF}}$ and $p^{\mathrm{CV}}_{\mathrm{MF}}$. To our knowledge, these are the first per-round DF and MF estimates for this protocol.

The DV-QDB values are computed in a noiseless setting. To make the CV estimator comparable, we calibrate the parameters to an honest one-round completeness of about 0.99. The squeezing r_{sq} controls the strength of the EPR correlation between the verifier's and prover's modes, so a larger value ties the honest response more tightly to the reference and raises completeness. The $\hbar = 2$ convention fixes the vacuum variance at 1, so the quadratures and the acceptance threshold are expressed in units of the vacuum (shot) noise. At this operating point we use $r_{\mathrm{sq}} = 1.2$ and $\delta_{\mathrm{cv}} = 1.1$, with zero added noise in the measured response quadrature. Figure 6 shows how completeness and the DF and MF estimates respond as these two parameters vary around the operating point. Panel (a) sweeps the threshold and panel (b) the squeezing.

2.8.1 CV distance fraud

In the DF game, the distant prover must answer before the challenge mode arrives, so its response cannot depend on the verifier's reference value. For each branch (a, b), let m_i be the verifier's reference homodyne value and let $\mathrm{Var}_{\mathrm{resp}}(b)$ be the smallest response variance the prover can achieve in basis b. Because the response and the reference are independent, the error variance is

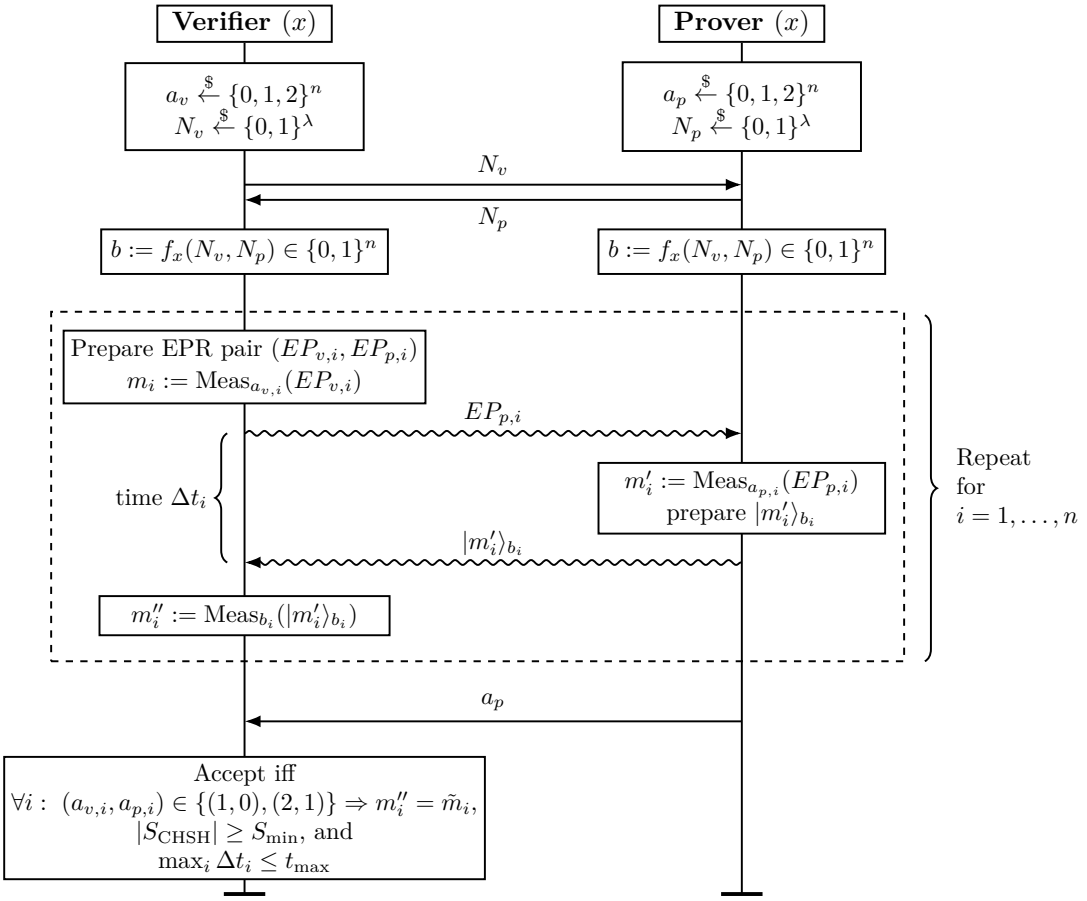


Figure 4: Overview of the E91 QDB protocol [BSA24]. Before the fast rounds, the parties choose the measurement-setting strings a_v and a_p, exchange nonces, and derive the response-basis string b. In each fast round, the verifier prepares an entangled pair, measures its own half in setting $a_{v,i}$, and sends the prover half $EP_{p,i}$. The prover measures in setting $a_{p,i}$ and returns $|m'_i\rangle_{b_i}$. After all rounds, the prover reveals a_p. The verifier accepts only if the logical target $\tilde{m}_i$ matches on the retained setting pairs $(1,0)$ and $(2,1)$, the sign-adjusted CHSH quantity satisfies $|S_{\mathrm{CHSH}}| \geq S_{\min}$ on $(0,0)$, $(0,2)$, $(2,0)$, and $(2,2)$, and every round-trip time Δt_i stays below $t_{\max}$. The two conventions for the logical target $\tilde{m}_i$ are stated in Section 2.7.

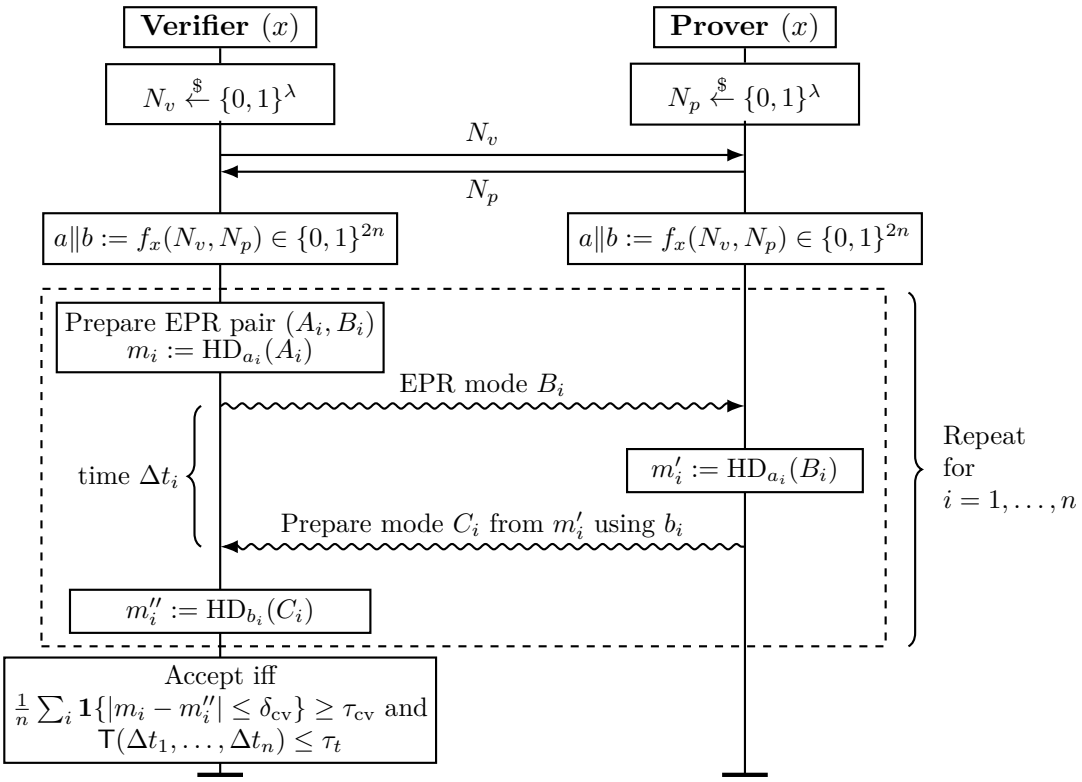


Figure 5: Overview of the Gaussian CV-QDB protocol [BAS25]. Before the fast rounds, the parties exchange nonces and derive the PRF outputs a and b. In each fast round, the verifier prepares an EPR pair, measures one mode in basis a_i, and sends the other mode to the prover. The prover measures that mode in basis a_i and prepares a response mode from the outcome using b_i. After all rounds, the verifier accepts only if the returned quadrature lies within tolerance δ_{cv} in a sufficient fraction of rounds and the round-trip times pass the timing check $\mathsf{T}(\Delta t_1, \ldots, \Delta t_n) \le \tau_t$.

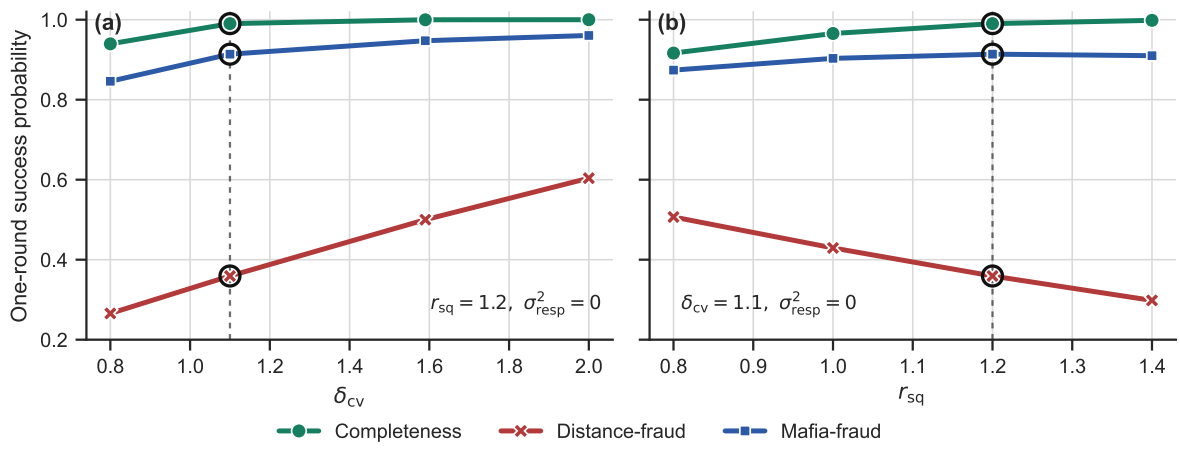


Figure 6: CV-QDB calibration sweeps with zero added response noise $\sigma^2_{\text{resp}} = 0$. Panel (a) varies the threshold at fixed $r_{\text{sq}} = 1.2$, while panel (b) varies the squeezing at fixed $\delta_{\text{cv}} = 1.1$. The mafia-fraud probabilities shown are finite-grid Gaussian estimates recomputed for each parameter point. Black outlines identify the calibrated operating point used in Table 1.

the sum of the two,

$$\text{MSE}^{\text{DF}}_{a,b} = \text{Var}(m_i) + \text{Var}_{\text{resp}}(b).$$

A round passes when the returned value lies within δ_{cv} of the reference. With a Gaussian error, each branch passes with an error-function probability, and averaging over the four equally likely branches gives

$$p^{\text{CV}}_{\text{DF}} = \frac{1}{4}\sum_{a,b} \text{erf}\left(\frac{\delta_{\text{cv}}}{\sqrt{2\,\text{MSE}^{\text{DF}}_{a,b}}}\right).$$

At the calibrated operating point this gives $p^{\text{CV}}_{\text{DF}} \approx 0.3592$.

2.8.2 CV mafia fraud

The CV MF model has the same two-slot structure as the DV MF, a pre-ask interaction followed by the verifier's challenge. The adversary sends a Gaussian probe to the honest prover and measures the returned mode. It then combines this pre-ask information with a measurement of the verifier's later challenge mode to estimate the verifier's target value. Unlike the DV MF, the CV version has no exact comb SDP. A Gaussian attack is fixed by a few continuous parameters, so we scan a finite grid of these parameters and keep the strongest attack. We add no response noise, an attacker-favorable choice that can only raise the estimated attack success. We bound the probe energy by $N_{\text{pre}} \leq 2$, since an unbounded probe could read out the hidden basis and pass trivially. At the calibrated operating point this gives $p^{\text{CV}}_{\text{MF}}(N_{\text{pre}} \leq 2) \approx 0.9138$. This is a finite-grid Gaussian estimate, not a certified optimum over all finite-energy attacks. The grid and estimator details are in Section 4.6.

3 Discussion

We introduced a uniform one-round fast-phase framework for QDB and showed that the DV-QDB DF and MF strategy sets have exact semidefinite-program representations, a state SDP for DF and a two-slot comb SDP for MF. Unlike in QPV, where attack optimization is nonconvex and analyses rely on relaxations, these SDPs represent every allowed strategy, so their optima equal

the DV game values rather than relaxation bounds. This gives a single way to compare protocols across families, rather than a separate attack tailored to each one. For QDB 2019 and the first timed return leg of Mutual QDB, the SDP yields higher one-round MF values than previously reported. For E91 QDB, we report the first per-round DF and MF values; for CV-QDB, the first per-round estimates.

For every DV-QDB protocol, $p^{\star}_{\mathrm{DF}} = 0.5$, so DF does not distinguish between them. The MF values in the benchmark instead range from 0.75 to 0.9332. What matters for MF resistance is not whether a protocol is prepare-and-measure or entanglement-based, but whether an early interaction with the prover can be converted into a valid response to a fresh verifier challenge. Protocols in the style of Brands and Chaum, such as QDB 2017, whose MF resistance comes from a final MAC rather than from the fast phase itself, fall outside this comparison (Section 4.5).

The CV-QDB results point in the same direction, with one caveat. Because acceptance is threshold-based, the estimates $p^{\mathrm{CV}}_{\mathrm{DF}}$ and $p^{\mathrm{CV}}_{\mathrm{MF}}$ depend on the chosen squeezing, threshold, energy bound, grid, estimator class, and noise model.

These results concern reduced one-round fast-phase games, not full-protocol fraud probabilities. A full-protocol adversary attacks all n rounds with a single strategy. It can keep quantum memory between rounds, adapt to earlier rounds, and interleave its pre-asks and responses, so the n-round fraud probability cannot simply be assumed to be the one-round value raised to the power n. How repeated-game values relate to one-round values is the parallel-repetition question, which is subtle even for the most-studied games. The repeated value can decay more slowly than the n-th power [Raz11], and for games with entangled players the known general bounds are weaker still [Yue16]. The QDB security framework [BASP26] provides the multi-round side of the analysis; one route there is a concentration bound that tolerates adaptive adversaries and turns a per-round success cap, if one holds conditionally on the adversary's history, into a soundness bound for threshold acceptance. The exact DV one-round values reported here are the natural per-round constants for such a multi-round lifting, and for QDB 2019 the stronger attack of Table 1 updates the per-round value used there. Whether these one-round values cap the adversary's success in every round of a full execution is a protocol-by-protocol question beyond our one-round scope.

A natural next step is to feed these per-round results into a secure-ranging-rate framework for QDB, analogous to a secure key rate in quantum key distribution. Such a framework would combine $p^{\star}_{\mathrm{DF}}$, $p^{\star}_{\mathrm{MF}}$, $p^{\mathrm{CV}}_{\mathrm{DF}}$, and $p^{\mathrm{CV}}_{\mathrm{MF}}$ with honest completeness, timing statistics, multi-round thresholds, post-fast-phase checks, and the threat model to quantify the rate at which secure ranging decisions can be certified.

4 Methods

Our analysis rests on two standard ingredients, recalled in this section. The first is the QDB security framework [BASP26], which sets the rules of the attacks we study. The second is semidefinite programming, the convex-optimization tool we use to solve the resulting games.

4.1 QDB security framework

We first recall the completeness definition from the QDB security framework [BASP26]. Completeness means that an honest prover within the distance bound B of the verifier is accepted with overwhelming probability.

Definition 1 (Completeness)**.** If an honest prover $\mathcal{P}$ is located within the distance bound at distance $d \le B$ from the verifier $\mathcal{V}$, where $r_{\mathcal{V}}$ denotes the verifier's internal randomness, then

$$\Pr\Big[(\mathcal{V}(x, r_{\mathcal{V}}) \;\leftrightarrow\; \mathcal{P}(x)) \text{ accepts}\Big] \;\ge\; 1 - \mathrm{negl}(\lambda).$$

The framework also defines the DF and MF experiments, which the reduced games of Sections 2.2 and 2.3 build on.

Definition 2 (Distance-fraud experiment)**.**

1. Setup. The verifier $\mathcal{V}$ and a dishonest prover $\mathcal{P}^\star$ share a long-term secret key x. The dishonest prover $\mathcal{P}^\star$ is located outside the distance bound at a distance $d > B$ from $\mathcal{V}$.

2. Challenge session. $\mathcal{V}$ and $\mathcal{P}^\star$ engage in a complete execution of the QDB protocol, which includes n fast rounds subject to the distance bound check enforced by $\mathcal{V}$.

3. Output. $\mathcal{V}$ outputs $\mathrm{Out}_\mathcal{V} \in \{0, 1\}$.

Definition 3 (Mafia-fraud experiment)**.**

1. Setup. Verifier $\mathcal{V}$ and honest prover $\mathcal{P}$ share a long-term secret key x. Two QPT adversaries, $\mathcal{A}_1$ (co-located with $\mathcal{V}$) and $\mathcal{A}_2$ (co-located with $\mathcal{P}$), share a timing-constrained authenticated classical and quantum channel.

2. Learning phase (Pre-ask). The adversaries may initiate and control any polynomial number of auxiliary executions of the QDB protocol between $\mathcal{V}$ and $\mathcal{P}$: in each such execution, every classical and quantum message between the honest parties passes through $(\mathcal{A}_1, \mathcal{A}_2)$, who may relay, delay, drop, modify, or inject messages arbitrarily (subject only to the timing constraints). At the end of this phase, $(\mathcal{A}_1, \mathcal{A}_2)$ retain the entire classical transcript and their joint quantum state.

3. Challenge session. The adversaries $(\mathcal{A}_1, \mathcal{A}_2)$ interact with $\mathcal{V}$ in a full execution of the QDB protocol. Simultaneously, they may interact with the honest $\mathcal{P}$ (who uses the correct long-term key x and is at distance $d > B$). $\mathcal{V}$ enforces the distance bound B.

4. Output. $\mathcal{V}$ outputs $\mathrm{Out}_\mathcal{V} \in \{0, 1\}$.

Each experiment is a game whose output is 1 when the verifier accepts and 0 otherwise. The adversary's DF or MF advantage is its probability of making the verifier accept. A QDB protocol is secure against the corresponding fraud if this advantage is negligible in the security parameter λ for every QPT adversary that respects the location and timing restrictions. Here λ sets the length of the protocol's secret values (the nonces are λ-bit strings) and the strength of its PRF and MAC. The full protocol runs n fast rounds, and security must hold even against an adversary that attacks all of them and runs any polynomial number of extra sessions.

4.2 Semidefinite programs

SDP methods are standard tools for optimizing quantum strategies [GW07, CDP09] and have been used before to analyze attacks on QDB protocols [Ver22]. For background, see [BV04] on convex optimization and [Wat18, SC23] on SDPs in quantum information, where a typical use is to compute the best success probability that any quantum strategy can achieve in a given task. We use SDPs in exactly this way: the strategies we optimize over are attacks, so the computed value bounds what any attacker can do.

The quantities our programs optimize are probabilities, so we need matrix tools that can never produce a negative number. A matrix X is *Hermitian* when it equals its own conjugate transpose; its eigenvalues are then real. If in addition no eigenvalue is negative, X is *positive semidefinite* (PSD), written $X \succeq 0$; equivalently, $\langle\psi|X|\psi\rangle \geq 0$ for every vector $|\psi\rangle$. Hermitian matrices are thus the matrix analogue of real numbers, and PSD matrices the analogue of nonnegative numbers. Our strategy variables will always be PSD matrices, precisely to enforce this nonnegativity. Comparisons are defined as for numbers: for Hermitian A and B, we write

$A \succeq B$ when $A - B \succeq 0$. The *trace* $\mathrm{Tr}(M)$ is the sum of the diagonal entries of M; the trace of a product, $\mathrm{Tr}(AB)$, pairs the entries of A and B and is the matrix analogue of the dot product of two vectors. One fact is used repeatedly, the analogue of the rule that a product of two nonnegative numbers is nonnegative: if $A \succeq 0$ and $B \succeq 0$, then $\mathrm{Tr}(AB) \geq 0$. In quantum theory, this fact guarantees that measuring a state never produces a negative probability.

An SDP maximizes a linear function of a single PSD matrix variable, subject to linear constraints:

$$\max_{X \succeq 0} \ \mathrm{Tr}(CX) \qquad \text{subject to} \qquad \mathrm{Tr}(A_i X) = b_i \quad \text{for } i = 1, \ldots, m, \tag{3}$$

where the Hermitian matrices $C, A_1, \ldots, A_m$ and the numbers $b_1, \ldots, b_m$ are fixed data and the PSD matrix X is the variable. In our programs the variable X is the attack strategy, the one thing the adversary is free to choose, and the fixed data describe the protocol under attack. The function being maximized, called the *objective*, is then the probability that the attack succeeds. The constraints, together with $X \succeq 0$, are the conditions that make X a physical strategy, one that quantum mechanics allows the adversary to implement: the linear constraints normalize X so that its outcome probabilities sum to one, and $X \succeq 0$ keeps them nonnegative. Both the objective and the constraints are linear in X, because X enters them only through traces against fixed matrices.

Our SDPs will not always look exactly like Equation (3): some minimize instead of maximize, some have inequality constraints instead of equalities, and some use several matrix variables at once. Standard rewritings turn each of these variations into the form above, so we state every program in whatever form is most natural. Every SDP is *convex*. A general optimization problem can have local optima: solutions better than all nearby alternatives but worse than the true optimum. A numerical solver improves its answer step by step, so it can get stuck at such a point and report it as the best. In a convex problem every local optimum is the global one, so this failure mode disappears.

Quantum problems map to SDPs directly, because the basic objects of quantum theory are PSD matrices with linear side conditions. A state is a density matrix σ ($\sigma \succeq 0$, $\mathrm{Tr}(\sigma) = 1$). A measurement with outcomes i is a set of matrices $\{E_i\}$, called a POVM, with each $E_i \succeq 0$ and $\sum_i E_i = I$; outcome i occurs on state σ with probability $\mathrm{Tr}(E_i\sigma)$. These numbers behave as probabilities must: each is nonnegative by the trace fact above, and because $\sum_i E_i = I$, they sum to $\mathrm{Tr}(\sigma) = 1$. A channel, and more generally a multi-step strategy, is again a PSD matrix with linear conditions, in the Choi form introduced in Section 2.3. Probabilities like $\mathrm{Tr}(E_i\sigma)$ are traces of products and therefore linear in each object separately: fix the measurement and the probability is linear in the state; fix the state and it is linear in the measurement. Linear has a concrete meaning here: prepare an equal mixture of two states, and each outcome probability is the average of the two separate probabilities. This is exactly the linearity an SDP needs.

Modeling a problem as an SDP therefore takes three steps: identify what the optimizing party freely chooses, while everything the protocol fixes becomes constant data; write each free choice as a PSD variable together with the linear conditions that make it physical; and check that the success probability is linear in these variables. Only the last step can fail. It holds whenever a single party optimizes, but when two separated parties optimize jointly, their choices multiply inside the objective and the problem is in general nonconvex. That is the QPV obstacle discussed in Section 1: there too the optimizing party is the adversary, but a QPV adversary is a pair of attackers at two separated locations, exactly the situation where choices multiply. Our MF adversary is also a cooperating pair, yet our games avoid the obstacle, because in the one-round game the pair collapses to a single sequential strategy.

We now carry out these three steps on a simple example: minimum-error discrimination of two states [Hel69]. The task is a toy version of mafia fraud: the adversary holds a challenge state that encodes a bit it does not know and must guess that bit. A referee flips a fair coin $b \in \{0, 1\}$ and sends the qubit state ρ_b, where $\rho_0 = |0\rangle\langle 0|$ and $\rho_1 = |+\rangle\langle +|$ with $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$;

the receiver measures the qubit and guesses b. The only free choice is the measurement, a two-outcome POVM (E_0, E_1) with outcome i meaning guess $b = i$, and the success probability is linear in (E_0, E_1). The optimal guessing probability is therefore the SDP value

$$p^\star = \max_{E_0, E_1} \ \tfrac{1}{2}\operatorname{Tr}(E_0\rho_0) + \tfrac{1}{2}\operatorname{Tr}(E_1\rho_1) \quad \text{subject to} \quad E_0 \succeq 0,\ E_1 \succeq 0,\ E_0 + E_1 = I. \tag{4}$$

Every *feasible* point, a pair (E_0, E_1) satisfying the constraints, is a measurement the receiver can actually perform, so feasible points are strategies, and each one lower-bounds $p^\star$. For these two states the optimum is $p^\star = \cos^2(\pi/8) \approx 0.854$, achieved by measuring in the basis that makes an angle of $\pi/8$ with each of the two states [Hel69].

An upper bound on $p^\star$ must hold for every measurement at once; this is what the dual delivers. Take any Hermitian Y with $Y \succeq \frac{1}{2}\rho_0$ and $Y \succeq \frac{1}{2}\rho_1$. For every feasible (E_0, E_1),

$$\tfrac{1}{2}\operatorname{Tr}(E_0\rho_0) + \tfrac{1}{2}\operatorname{Tr}(E_1\rho_1) \ \le\ \operatorname{Tr}(E_0 Y) + \operatorname{Tr}(E_1 Y) \ =\ \operatorname{Tr}\big((E_0 + E_1)Y\big) \ =\ \operatorname{Tr}(Y),$$

where the inequality applies the trace fact above to E_i and $Y - \frac{1}{2}\rho_i$. Each such Y is therefore a certificate: no measurement whatsoever succeeds with probability above $\operatorname{Tr}(Y)$. Minimizing $\operatorname{Tr}(Y)$ over all such Y is itself an SDP, called the *dual*; the original maximization Equation (4) is the *primal*. The bound $p^\star \le \operatorname{Tr}(Y)$ is called *weak duality*. The primal searches over strategies and pushes the value up from below; the dual searches over certificates and presses it down from above. In the example the two meet: the best certificate also has value $\cos^2(\pi/8)$. This is not special to the example: by *strong duality*, the two values coincide for every SDP that satisfies a mild extra condition (Slater's condition), as all SDPs in this paper do [BV04]. The dual is constructed mechanically from the primal (here Y corresponds to the constraint $E_0 + E_1 = I$), and Section 4.4 states the dual of our MF program, built in the same way.

In practice, standard software first supplies candidate primal and dual solutions. One states the primal in a few lines of code, essentially as written in Equation (4), and a solver computes the optimum, to a given accuracy, in time polynomial in the matrix dimension and the number of constraints. Two features of the solver output matter here. First, the solver returns a primal and a dual solution together, so every computed value arrives with a candidate optimal strategy and a candidate certificate. Second, the output is floating point and satisfies the constraints only up to a small tolerance, so on its own it is numerical evidence, not a proof. For the MF values, we reconstruct explicit algebraic primal and dual witnesses and verify them in exact arithmetic. The verifier rebuilds the SDP tester and checks that both witnesses are feasible and have the same objective value, without calling the solver or reading its floating-point output (Section 4.4). The SDPs in this paper are small by solver standards: the largest variable, the MF strategy matrix of Section 2.3, is a 16×16 matrix.

Our DV-QDB fast-round games fit this three-step pattern. The classical data in a round (the challenge bits, key bits, basis choices, and measurement outcomes) take only finitely many values, and the quantum messages are qubits rather than infinite-dimensional systems, so the whole game can be written with finite-size matrices. Only the adversary's side is free. In DF it chooses the response states it returns; in MF it chooses one two-step strategy, captured by a single PSD matrix whose normalization conditions are linear. The verifier's acceptance probability is linear in these variables. The DF and MF games of Sections 2.2 and 2.3 are therefore SDPs. For the MF values, Section 4.4 gives a strategy that achieves each value and an exact certificate that nothing does better.

4.3 Assumptions and scope

The results should be read under the following assumptions and scope restrictions:

- One-round analysis. We extract a single timed fast round from the full QDB execution and omit correlations across rounds. The reported quantities concern one-round fast-phase

games, not full-protocol fraud probabilities. Section 3 discusses how the DV values and CV estimates relate to security over all n rounds. Later protocol checks and implementation-level attacks are outside these games.

- Ideal randomness. We model the secret values derived from the pseudorandom function (PRF) in the slow phase as independent uniform bits. This idealization assumes that nonces are fresh and that the PRF is secure against quantum polynomial-time (QPT) adversaries, i.e., efficient quantum attackers.
- Timing model. We use the timing constraints of the QDB security framework [BASP26]. In DF, the fast response cannot depend on information from the verifier's actual challenge. In MF, the response may use information obtained from the honest prover before the timed verifier challenge.
- Benchmark scope. The benchmark covers protocols whose fast phase itself authenticates the prover. QDB 2017 [AMSP17] instead follows the design of Brands and Chaum [BC94]: its fast phase only checks proximity, and a final message authentication code (MAC) authenticates the session. The one-round game strips that MAC, so we treat QDB 2017 separately in Section 4.5.
- Protocol-specific reductions. For the Mutual QDB protocol [AES25], we report only the first of its two timed exchanges (the first timed return leg). For the E91 QDB protocol [BSA24], we report only the rounds in which verifier and prover measure in the same basis and the outcome is checked (the value-check branches).
- CV-QDB estimator. For Gaussian CV-QDB [BAS25], we use the restricted Gaussian estimator of Section 2.8 with zero added response noise, matching the DV-QDB analysis and making the attacker as powerful as possible. The reported MF result is an estimate from a finite parameter grid, not a certified optimum; the DF result is the exact optimum within this restricted Gaussian family.
- Terrorist fraud. TF security is about whether a malicious prover's help can be reused across protocol runs, so it falls outside our one-round analysis (see Section 2.1).

4.4 DV mafia-fraud certificate checks

To check the reported MF values without trusting the SDP solver, we certify each one with a matching pair of bounds. The primal SDP in Equation (2) maximizes the acceptance probability over all attack combs, so every feasible comb S is an explicit attack, and its acceptance probability is a lower bound on $p^{\star}_{\mathrm{MF}}$. The upper bound needs the dual.

Write the averaged tester as

$$\bar{W} := \frac{1}{|\Omega|} \sum_{\omega \in \Omega} W_{\omega},$$

the mean of the per-branch acceptance operators over the branch set Ω. With a Hermitian variable Y on $Q \otimes R \otimes C$ and a scalar μ, the dual is

$$\min_{Y,\mu} \ \mu$$

subject to

$$Y \otimes I_O \succeq \bar{W}, \qquad \mu I_Q \succeq \mathrm{Tr}_{RC}(Y), \qquad Y = Y^{\dagger}, \quad \mu \in \mathbb{R}.$$

The first constraint forces Y to dominate the averaged tester, and the second caps $\mathrm{Tr}_{RC}(Y)$ on the probe register Q by μ. Together they give $\mathrm{Tr}(\bar{W}S) \leq \mu$ for every comb S. No attack then accepts with probability above μ, so any dual-feasible (Y, μ) upper-bounds $p^{\star}_{\mathrm{MF}}$.

A feasible primal and a feasible dual bound $p^\star_{\mathrm{MF}}$ from below and above. If their objective values are the same number v, weak duality gives $v \le p^\star_{\mathrm{MF}} \le v$, and hence $p^\star_{\mathrm{MF}} = v$. The artifact supplies such a pair for every row of Table 3.

The exact verifier independently rebuilds $\bar{W}$ from the finite branch description of each protocol. It checks

$$S \succeq 0, \quad \mathrm{Tr}_O(S) = \eta \otimes I_{RC}, \quad \eta \succeq 0, \quad \mathrm{Tr}(\eta) = 1$$

for the primal witness, both dual slack inequalities above, and the exact identity $\mathrm{Tr}(\bar{W}S) = \mu$. Matrix positivity is proved by exact LDL decompositions. QDB 2017 and Mutual QDB use rational arithmetic, QDB 2019 uses $\mathbb{Q}(\sqrt{5})$, and E91 QDB uses $\mathbb{Q}(\sqrt{2}, \sqrt{24+17\sqrt{2}})$. In the last field, algebraic signs are decided with rational square-root enclosures. The verifier neither calls an SDP solver nor reads the stored floating-point certificates.

Table 3: Exact DV MF values. For each row, the artifact verifies a feasible primal and dual witness over the stated field with identical objective value.

Protocol	Exact $p^\star_{\mathrm{MF}}$ value	Certificate field
QDB 2017	1	$\mathbb{Q}$
QDB 2019	$(5+\sqrt{5})/8 = \sin^2(2\pi/5)$	$\mathbb{Q}(\sqrt{5})$
Mutual QDB, first leg	3/4	$\mathbb{Q}$
E91 QDB retained	$\left(8+\sqrt{24+17\sqrt{2}}\right)/16$	$\mathbb{Q}(\sqrt{2}, \sqrt{24+17\sqrt{2}})$

We also model all DV SDPs in CVXPY and solve them with SCS and CLARABEL. A separate NumPy check rebuilds each tester and reports the residuals, slack eigenvalues, and duality gap of the stored solver output. These floating-point results are a reproducibility check, not the proof of exact feasibility. They are shown in Table 4; the exact certificate verifier is run with `uv run qdb-benchmark exact-certificate --protocol all`.

Table 4: NumPy diagnostics for the stored floating-point DV MF solver output. The objective column is the stored primal objective. The final column is the minimum over the two dual slack spectra. Its small negative values reflect finite solver tolerance and show why this output is not itself a certificate; the exact proof is summarized in Table 3.

Protocol	Objective	Duality gap	Comb residual	Minimum dual slack eigenvalue
QDB 2017	0.999999992	3.57×10^{-9}	1.11×10^{-16}	-3.26×10^{-9}
QDB 2019	0.904508487	3.37×10^{-9}	1.82×10^{-13}	-2.02×10^{-9}
Mutual QDB, first leg	0.749999996	2.76×10^{-9}	1.84×10^{-16}	-1.43×10^{-9}
E91 QDB retained	0.933200437	2.55×10^{-11}	1.81×10^{-11}	-2.46×10^{-9}

4.5 QDB 2017 one-round values

QDB 2017 [AMSP17] derives the fast-phase basis string from a PRF and authenticates the session with a final MAC over the fast-phase transcript; the protocol flow is shown in Figure 7. The fast phase confirms that the prover is close (ranging), and the MAC that it is genuine (authentication). As explained in Section 2.4, the one-round fast-phase game strips the MAC, so the benchmark does not apply to this protocol. We report its fast-phase-only values here for completeness.

In the notation of Section 2.4, the reduced one-round instantiation is as follows. The branch is $\omega = (a, c)$, with DF response-time view $v(\omega) = a$. The challenge state and accept projector are $\rho_\omega = \Pi_\omega = |c\rangle_a\langle c|$. The honest-prover pre-ask channel is $\mathcal{N}_a(\rho) = \sum_t \langle t|_a \rho |t\rangle_a \, |t\rangle_a\langle t|$ with t summed over $\{0, 1\}$: the same measure-and-re-prepare map as QDB 2019, but with the response

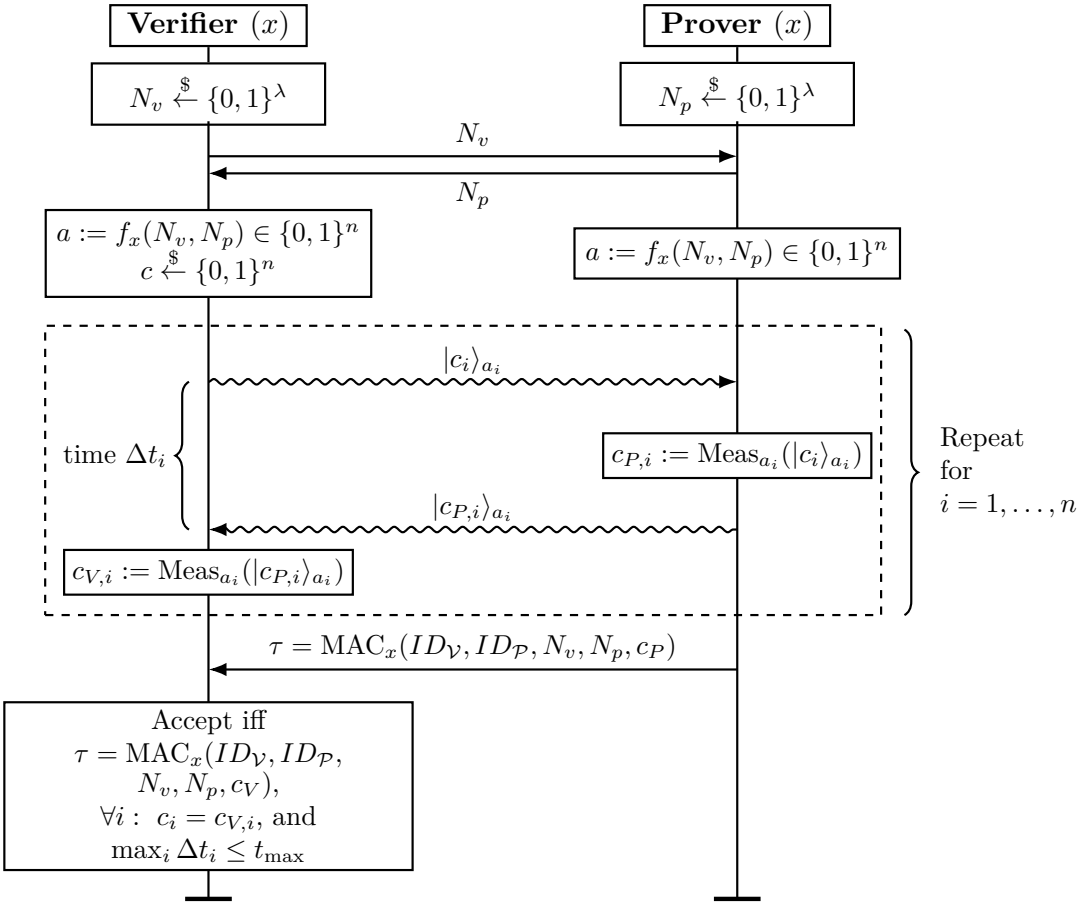


Figure 7: Overview of the QDB 2017 protocol [AMSP17]. Before the fast rounds, the parties exchange nonces and derive the PRF output a. In each fast round, the verifier prepares $|c_i\rangle_{a_i}$, and the prover measures it in basis a_i and re-prepares the outcome in the same basis. After all rounds, the prover sends $\tau = \mathrm{MAC}_x(ID_\mathcal{V}, ID_\mathcal{P}, N_v, N_p, c_P)$. The verifier accepts only if τ equals the MAC recomputed on c_V, all recovered bits match the originals, and every round-trip time Δt_i stays below $t_{\max}$.

basis equal to the challenge basis. The one-round values are $p^\star_{\mathrm{DF}} = 0.5$ and $p^\star_{\mathrm{MF}} = 1$; the MF value is certified in Section 4.4.

Without the authentication the MAC provides, the adversary does not even need the prover: it reflects the challenge state straight back to the verifier, which accepts with probability 1. This reflection attack is already noted in [AMSP17], and the full protocol uses the MAC to catch it. The value $p^\star_{\mathrm{MF}} = 1$ therefore reflects the missing authentication, not a weakness of the full protocol. It does show that ranging and authentication cannot be separated in a DB protocol: a fast phase that only ranges provides no MF resistance on its own.

4.6 CV-QDB finite-grid Gaussian MF estimator details

This section gives the grid and estimator details behind the restricted Gaussian MF estimator of Section 2.8, computed with zero added response noise. A Gaussian attack depends on only a few continuous parameters, such as the probe amplitude and the measurement angles. We restrict each one to a finite set of values and optimize over the resulting grid. Refining the grid may expose stronger attacks.

At a high level, the attack has three steps. First, the adversary sends a Gaussian probe to the honest prover and measures the returned mode, obtaining a pre-ask value Z_ϕ that leaks

information about the hidden bases. Second, when the verifier's challenge mode arrives, the adversary picks a measurement angle θ from Z_ϕ and reads off the outcome Y_θ. Third, it turns Y_θ into a linear estimate of the verifier's reference quadrature and returns that as its response. The grid searches over the parameters of the first two steps for the estimate that passes most often.

Concretely, the adversary sends a Gaussian probe mode Q. The prover measures the quadrature selected by a and returns a response mode whose b-quadrature contains $g_a U_a$, where $U_a = X_Q$ for $a = 0$, $U_a = P_Q$ for $a = 1$, and $g_0 = 1, g_1 = -1$. The adversary measures this returned mode at angle ϕ, obtaining Z_ϕ. Later, after receiving the verifier's actual challenge mode, it chooses a challenge-measurement angle $\theta(Z_\phi)$, obtains $Y_{\theta(Z_\phi)}$, and prepares a response mode with quadrature estimates $\widehat{M}_0$ and $\widehat{M}_1$. We write M_a for the verifier's reference quadrature in basis a. The response estimates are scored with zero added noise in the measured response quadrature (an attacker-favorable relaxation, not a certified finite-energy output-mode model).

Ideally, the adversary would maximize its average chance of landing within δ_{cv} of the verifier's reference quadrature,

$$\frac{1}{4}\sum_{a,b}\Pr\left[|M_a - \widehat{M}_b(Z_\phi, Y_{\theta(Z_\phi)})| \le \delta_{\mathrm{cv}}\right],$$

over the energy-constrained Gaussian probe, the pre-ask measurement, the adaptive challenge measurement, and the Gaussian response estimators. To our knowledge, no technique solves this continuous nonconvex optimization exactly, so we report the finite-grid estimate $p_{\mathrm{MF}}^{\mathrm{CV}}$ instead, with the grid and energy bound below.

The reported grid fixes the pre-ask probe to an unsqueezed displaced Gaussian state. Its displacement angle is $\pi/4$ and its amplitude runs from 0 to 3.0 in steps of 0.1, subject to $N_{\mathrm{pre}} \le 2$. The pre-ask angle ϕ and the challenge angle θ each range over 181 points uniformly spaced in $[0, 2\pi)$, and the pre-ask outcome Z_ϕ over a 1001-point quadrature grid. The measured response quadrature carries zero added noise in each response basis b, matching the noiseless DV upper bound, while the inactive quadrature is assigned vacuum variance $\hbar/2 = 1$. The pre-ask energy uses the $\hbar = 2$ convention,

$$N_{\mathrm{pre}} = \frac{\mathrm{Var}(X_Q) + \mathrm{Var}(P_Q) + \mathbb{E}[X_Q]^2 + \mathbb{E}[P_Q]^2}{2\hbar} - \frac{1}{2}.$$

The Z_ϕ-grid is uniform on

$$\left[\min_{a,b}(\mu_{ab} - 8\sqrt{v_{ab}}), \max_{a,b}(\mu_{ab} + 8\sqrt{v_{ab}})\right],$$

where μ_{ab} and v_{ab} are the mean and variance of Z_ϕ for the branch labeled by (a, b), and the integral over Z_ϕ is evaluated by the trapezoidal rule. For each pre-ask outcome $Z_\phi = z$, let $q_{ab}(z) = \Pr[a, b \mid Z_\phi = z]$ be the posterior probability that the hidden bases are (a, b) given z. For a candidate challenge angle θ, let Y_θ be the measured challenge quadrature. The adversary does not know the hidden bases, so it estimates M_a from Y_θ by linear regression and weights the regression coefficient by the posterior over a,

$$\widehat{M}_b = \kappa_b(z, \theta) Y_\theta, \qquad \kappa_b(z, \theta) = \frac{\sum_a q_{ab}(z)\,\mathrm{Cov}(M_a, Y_\theta)/\,\mathrm{Var}(Y_\theta)}{\sum_a q_{ab}(z)},$$

with $\kappa_b = 0$ when the denominator is zero. The conditional error variance for branch (a, b), written V_{err}, is

$$\mathrm{Var}(M_a) + \kappa_b^2\,\mathrm{Var}(Y_\theta) - 2\kappa_b\,\mathrm{Cov}(M_a, Y_\theta) + \mathrm{Var}_{\mathrm{resp}}(b).$$

The branch then passes with probability

$$\Pr(|\Delta| \le \delta_{\mathrm{cv}}) = \mathrm{erf}\left(\frac{\delta_{\mathrm{cv}}}{\sqrt{2V_{\mathrm{err}}}}\right).$$

A larger $\mathrm{Var}_{\mathrm{resp}}(b)$ therefore raises V_{err} and lowers this probability for a fixed estimator, so setting the response noise to zero is attacker-favorable within this estimator model. It is not, however, a physical finite-energy constraint. If the adversary must output a single CV mode while the hidden basis b is uncertain, its two response quadratures jointly obey a covariance uncertainty bound. We do not optimize over nonlinear estimators or over physical two-quadrature response covariance constraints. With this restricted grid, we obtain the zero-response-noise estimator

$$p_{\mathrm{MF}}^{\mathrm{CV}}(N_{\mathrm{pre}} \leq 2) \approx 0.913815.$$

4.7 Use of AI assistants

AI assistants were used under the authors' direction to help develop the benchmark code and to revise the manuscript text. All reported numerical results were checked by the authors, and the DV MF values are additionally verified by the exact-arithmetic certificate checker in the released artifact.

Data availability

All data supporting this study are available at `https://github.com/kevinbogner/qdb-benchmark` and archived at `https://doi.org/10.5281/zenodo.21361445` [Bog26].

Code availability

All code used to produce the reported results, including the exact-arithmetic certificate verifier, is available at `https://github.com/kevinbogner/qdb-benchmark` and archived at `https://doi.org/10.5281/zenodo.21361445` [Bog26].

Acknowledgements

This work was supported by CyberSecurity Research Flanders with reference number VR20192203, and by the European Union through the Horizon Europe project *Quantum Secure Networks Partnership* (QSNP, grant agreement No. 101114043) and the Connecting Europe Facility project BENELUX-QCI (grant agreement No. 101249520). The funders played no role in the design of the study, the analysis and interpretation of the results, or the writing of the manuscript.

Author contributions

K.B. developed the one-round games and their SDP reductions, implemented the benchmark code and the exact certificates, and wrote the manuscript. A.A., D.S., and B.P. supervised the work, contributed to the evaluation and interpretation of the results throughout multiple iterations, and revised the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare no financial or non-financial competing interests.

References

[ABB+18] Gildas Avoine, Muhammed Ali Bingöl, Ioana Boureanu, Srdjan Čapkun, Gerhard Hancke, Süleyman Kardaş, Chong Hee Kim, Cédric Lauradoux, Benjamin Martin,

Jorge Munilla, et al., *Security of distance-bounding: A survey*, ACM Computing Surveys (CSUR) **51** (2018), no. 5, 1–33.

[Abi19] Aysajan Abidin, *Quantum distance bounding*, Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, 2019, pp. 233–238.

[AEFR+23] Rene Allerstorfer, Llorenç Escolà-Farràs, Arpan Akash Ray, Boris Škorić, Florian Speelman, and Philip Verduyn Lunel, *Security of a continuous-variable based quantum position verification protocol*, arXiv:2308.04166, 2023.

[AES25] Aysajan Abidin, Karim Eldefrawy, and Dave Singelée, *Entanglement-based mutual quantum distance bounding*, Cyber Security, Cryptology, and Machine Learning: 8th International Symposium, CSCML 2024, Be'er Sheva, Israel, December 19–20, 2024, Proceedings, Lecture Notes in Computer Science, vol. 15349, Springer, 2025, pp. 219–235.

[AMSP17] Aysajan Abidin, Eduard Marin, Dave Singelée, and Bart Preneel, *Towards quantum distance bounding protocols*, Radio Frequency Identification and IoT Security: 12th International Workshop, RFIDSec 2016, Hong Kong, China, November 30–December 2, 2016, Revised Selected Papers, Springer, 2017, pp. 151–162.

[BAS25] Kevin Bogner, Aysajan Abidin, and Dave Singelée, *Continuous variable quantum distance bounding*, IEEE INFOCOM 2025-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2025, pp. 1–6.

[BASP26] Kevin Bogner, Aysajan Abidin, Dave Singelée, and Bart Preneel, *Security framework for quantum distance-bounding*, Quantum Information Processing **25** (2026), no. 5, Paper No. 162.

[BC94] Stefan Brands and David Chaum, *Distance-bounding protocols*, Advances in Cryptology - EUROCRYPT '93 (Berlin, Heidelberg) (Tor Helleseth, ed.), Lecture Notes in Computer Science, vol. 765, Springer, 1994, pp. 344–359.

[BCF+14] Harry Buhrman, Nishanth Chandran, Serge Fehr, Ran Gelles, Vipul Goyal, Rafail Ostrovsky, and Christian Schaffner, *Position-based quantum cryptography: Impossibility and constructions*, SIAM Journal on Computing **43** (2014), no. 1, 150–178.

[BCS22] Andreas Bluhm, Matthias Christandl, and Florian Speelman, *A single-qubit position verification protocol that is secure against multi-qubit attacks*, Nature Physics **18** (2022), no. 6, 623–626.

[BMV15] Ioana Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay, *Practical and provably secure distance-bounding*, Journal of Computer Security **23** (2015), no. 2, 229–257.

[Bog26] Kevin Bogner, *qdb-benchmark: code and data artifact for* Security evaluation of quantum distance-bounding protocols via semidefinite programming, Zenodo, 2026, Version 1.0.1, `https://doi.org/10.5281/zenodo.21361445`.

[BSA24] Kevin Bogner, Dave Singelée, and Aysajan Abidin, *Entangled states and Bell's inequality: A new approach to quantum distance bounding*, 2024 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2024, pp. 1–6.

[BV04] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.

[CDP09] Giulio Chiribella, Giacomo Mauro D'Ariano, and Paolo Perinotti, *Theoretical framework for quantum networks*, Physical Review A **80** (2009), no. 2, 022339.

[EFRA+24] Llorenç Escolà-Farràs, Arpan Akash Ray, Rene Allerstorfer, Boris Škorić, and Florian Speelman, *Continuous-variable quantum position verification secure against entangled attackers*, Physical Review A **110** (2024), no. 6, 062605.

[EFS23] Llorenç Escolà-Farràs and Florian Speelman, *Single-qubit loss-tolerant quantum position verification protocol secure against entangled attackers*, Physical Review Letters **131** (2023), no. 14, 140802.

[FO13] Marc Fischlin and Cristina Onete, *Terrorism in distance bounding: Modeling terrorist-fraud resistance*, Applied Cryptography and Network Security: 11th International Conference, ACNS 2013, Banff, AB, Canada, June 25–28, 2013. Proceedings, Springer, 2013, pp. 414–431.

[GW07] Gus Gutoski and John Watrous, *Toward a general theory of quantum games*, Proceedings of the 39th Annual ACM Symposium on Theory of Computing, ACM, 2007, pp. 565–574.

[Hel69] Carl W. Helstrom, *Quantum detection and estimation theory*, Journal of Statistical Physics **1** (1969), no. 2, 231–252.

[HK05] Gerhard P. Hancke and Markus G. Kuhn, *An RFID distance bounding protocol*, Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SecureComm 2005), 2005, pp. 67–73.

[KMS11] Adrian Kent, William J. Munro, and Timothy P. Spiller, *Quantum tagging: Authenticating location via quantum information and relativistic signaling constraints*, Physical Review A **84** (2011), no. 1, 012326.

[MSTPTR18] Sjouke Mauw, Zach Smith, Jorge Toro-Pozo, and Rolando Trujillo-Rasua, *Distance-bounding protocols: Verification without time and location*, 2018 IEEE Symposium on Security and Privacy (SP), IEEE, 2018, pp. 549–566.

[Raz11] Ran Raz, *A counterexample to strong parallel repetition*, SIAM Journal on Computing **40** (2011), no. 3, 771–777.

[SC23] Paul Skrzypczyk and Daniel Cavalcanti, *Semidefinite programming in quantum information science*, IOP Publishing, 2023.

[TFKW13] Marco Tomamichel, Serge Fehr, Jędrzej Kaniewski, and Stephanie Wehner, *A monogamy-of-entanglement game with applications to device-independent quantum cryptography*, New Journal of Physics **15** (2013), no. 10, 103002.

[Vau13] Serge Vaudenay, *On modeling terrorist frauds: Addressing collusion in distance bounding protocols*, International Conference on Provable Security, Springer, 2013, pp. 1–20.

[Ver22] Sebastian Reynaldo Verschoor, *Quantum information in security protocols*, Ph.D. thesis, University of Waterloo, 2022.

[Wat18] John Watrous, *The theory of quantum information*, Cambridge University Press, 2018.

[Yue16] Henry Yuen, *A parallel repetition theorem for all entangled games*, 43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016), Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2016, pp. 77:1–77:13.