Electronic Colloquium on Computational Complexity, Report No. 167 (2024)

Space-bounded quantum interactive proof systems

François Le Gall*[1], Yupan Liu†[1], Harumichi Nishimura‡[2], and Qisheng Wang§[3,1]

[1]Graduate School of Mathematics, Nagoya University
[2]Graduate School of Informatics, Nagoya University
[3]School of Informatics, University of Edinburgh

Abstract

We introduce two models of space-bounded quantum interactive proof systems, QIPL and $\mathsf{QIP_UL}$. The $\mathsf{QIP_UL}$ model, a space-bounded variant of quantum interactive proofs (QIP) introduced by Watrous (CC 2003) and Kitaev and Watrous (STOC 2000), restricts verifier actions to unitary circuits. In contrast, QIPL allows logarithmically many intermediate measurements per verifier action (with a high-concentration condition on *yes* instances), making it the weakest model that encompasses the classical model of Condon and Ladner (JCSS 1995).

We characterize the computational power of QIPL and $\mathsf{QIP_UL}$. When the message number m is polynomially bounded, $\mathsf{QIP_UL} \subsetneq \mathsf{QIPL}$ unless $\mathsf{P} = \mathsf{NP}$:

- QIPL exactly characterizes NP.
- $\mathsf{QIP_UL}$ is contained in P and contains $\mathsf{SAC}^1 \cup \mathsf{BQL}$, where SAC^1 denotes problems solvable by classical logarithmic-depth, semi-unbounded fan-in circuits.

However, this distinction vanishes when m is constant. Our results further indicate that intermediate measurements uniquely impact space-bounded quantum interactive proofs, unlike in space-bounded quantum computation, where $\mathsf{BQL} = \mathsf{BQ_UL}$.

We also introduce space-bounded unitary quantum statistical zero-knowledge ($\mathsf{QSZK_UL}$), a specific form of $\mathsf{QIP_UL}$ proof systems with statistical zero-knowledge against any verifier. This class is a space-bounded variant of quantum statistical zero-knowledge (QSZK) defined by Watrous (SICOMP 2009). We prove that $\mathsf{QSZK_UL} = \mathsf{BQL}$, implying that the statistical zero-knowledge property negates the computational advantage typically gained from the interaction.

*Email: legall@math.nagoya-u.ac.jp
†Email: yupan.liu.e6@math.nagoya-u.ac.jp
‡Email: hnishimura@i.nagoya-u.ac.jp
§Email: QishengWang1994@gmail.com

ISSN 1433-8092

Contents

1 Introduction

Recent advancements in quantum computation with a limited number of qubits have been achieved from both theoretical and experimental perspectives. Theoretical work began in the late 1990s, focusing on feasible models of quantum computation operating under space restrictions, where the circuit acts on $O(\log n)$ qubits and consists of $\mathrm{poly}(n)$ elementary gates [Wat99, Wat03a]. These models, referred to as quantum logspace, were later shown during the 2010s to offer a quadratic space advantage for certain problems over the best known classical algorithms [TS13, FL18], which saturates the classical simulation bound. In recent years, this area has gained increased attention, particularly in eliminating intermediate measurements in these models [FR21, GRZ21], and through further developments [GR22, Zha24]. Motivated by these achievements in quantum logspace, we are interested in exploring the power of the quantum interactive proof systems where the verifier is restricted to quantum logspace.

To put it simply, in a single-prover (quantum) interactive proof system for a promise problem $(\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$, a computationally weak (possibly quantum) *verifier* interacts with a computationally all-powerful but untrusted *prover*. In quantum scenarios, the prover and verifier may share entanglement during their interactions. Given an input $x \in \mathcal{I}_{\text{yes}} \cup \mathcal{I}_{\text{no}}$, the prover claims that $x \in \mathcal{I}_{\text{yes}}$, but the verifier does not simply accept this claim. Instead, an interactive protocol is initiated, after which the verifier either "accepts" or "rejects" the claim. The protocol has completeness parameter c, meaning that if x is in $\mathcal{I}_{\text{yes}}$ and the prover honestly follows the protocol, the verifier accepts with probability at least c. The protocol has soundness parameter s, meaning that if x is in $\mathcal{I}_{\text{no}}$ then the verifier accepts with probability at most s, regardless of whether the prover follows the protocol. Typically, an interactive protocol for $(\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$ has completeness $c = 2/3$ and soundness $s = 1/3$.

Interactive proof systems with time-bounded verifier. The exploration of classical interactive proof systems (IP) was initiated in the 1980s [Bab85, GMR89]. In these proof systems, the verifier is typically bounded by polynomial time, and $\mathsf{IP}[m]$ represents interactive protocols involving m messages during interactions. Particularly, when the verifier's messages are merely random bits, these *public-coin* proof systems are known as *Arthur-Merlin proof systems* [Bab85]. Shortly thereafter, it was established that any constant-message IP protocol can be parallelized to two messages,[1] and thus $\mathsf{IP}[O(1)]$ is contained in the second level of the polynomial-time hierarchy [Bab85, GS86]. However, IP protocols with a polynomial number of messages have been shown to be exceptionally powerful, as demonstrated by the seminal result $\mathsf{IP} = \mathsf{PSPACE}$ [LFKN92, Sha92]. Consequently, IP protocols with a polynomial number of messages generally cannot be parallelized to a constant number of messages unless the polynomial-time hierarchy collapses.[2]

About fifteen years after the introduction of interactive proof systems (and a model of quantum computation), the study of quantum interactive proof systems (QIP) began [Wat03b]. Remarkably, any QIP protocol with a polynomial number of messages can be parallelized to three messages [KW00]. A quantum Arthur-Merlin proof system was subsequently introduced in [MW05], and any three-message QIP protocol can be transformed into this form (QMAM). By the late 2000s, the computational power of QIP was fully characterized: The celebrated result $\mathsf{QIP} = \mathsf{PSPACE}$ [JJUW11] established that QIP is not more powerful than IP as long as the gap $c - s$ is at least polynomially small. However, when the gap $c - s$ is double-exponentially small, this variant of QIP is precisely characterized by EXP [IKW12]. In the late 2010s, another quantum counterpart of the Arthur-Merlin proof system was considered in [KLGN19], where the verifier's message is either random bits or halves of EPR pairs, leading to a quadrichotomy theorem that classifies the corresponding QIP protocols.

[1] The resulting proof system is a two-message Arthur-Merlin proof system, denoted by AM.

[2] The assumption that the polynomial-time hierarchy does not collapse generalizes the conjecture that $\mathsf{P} \subsetneq \mathsf{NP}$.

Interactive proof systems with space-bounded verifier. The investigation of (classical) interactive proof systems with space-bounded verifiers started in the late 1980s [DS92, Con91], alongside research on time-bounded verifiers. Notably, by using the fingerprinting lemma [Lip90], Condon and Ladner [CL95] showed that the class of (private-coin) classical interactive proof systems with logarithmic-space verifiers using a logarithmic number of random bits exactly characterizes NP. In parallel, public-coin space-bounded classical interactive proofs were explored in the early 1990s [For89, FL93, Con92]. By around 2010, it was established that such space-bounded protocols with a polynomial number of public coins precisely characterize P [GKR15]. More recently, the efficiency of such space-bounded protocols has been further improved [CR23].

Although research has been conducted on quantum interactive proofs where the verifier uses quantum finite automata [NY09, NY15, Yak13], analogous to classical work [DS92], to our knowledge no prior work has addressed space-bounded counterparts of quantum interactive proofs that align with the circuit-based model defined in [KW00, Wat03b]. In the case *without interaction*, space-bounded quantum Merlin-Arthur proof systems have been studied recently. When the verifier has *direct access* to an $O(\log n)$-qubit message, meaning it can process the message directly in its workspace qubits, this variant (QMAL) is as weak as BQL [FKL+16, FR21]. However, when the (unitary) verifier has online access to a $\mathrm{poly}(n)$-qubit message, where each qubit in the message state is read-once, this variant is as strong as QMA [GR23].[3]

It is important to note that online and direct access to messages during interactions makes no difference for time-bounded interactive or Merlin-Arthur-type proof systems, whether classical or quantum. This distinction arises from the nature of space-bounded computation.[4]

1.1 Main results

Definitions of QIPL and $\mathsf{QIP_UL}$. We introduce *space-bounded quantum interactive proof systems* and their unitary variant, denoted as QIPL and $\mathsf{QIP_UL}$, respectively. In these proof systems, the verifier V operates in quantum logspace and has direct access to messages during interaction with the prover P. Specifically, in a $2l$-turn (message) space-bounded quantum interactive proof system for a promise problem $(\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$, this proof system $P \rightleftharpoons V$ consists of the prover's private register Q, the message register M, and the verifier's private register W. Both M and W are of size $O(\log n)$, with M being accessible to both the prover and the verifier.[5]

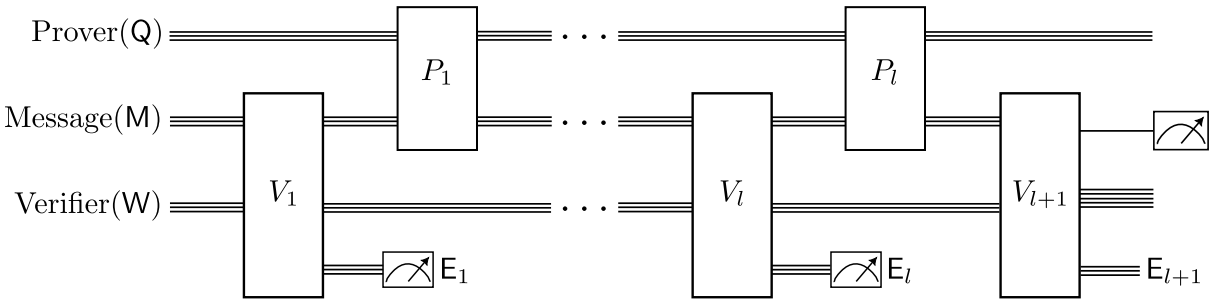

Figure 1.1: A $2l$-turn single-prover space-bounded quantum interactive proof system (QIPL).

The verifier V maps an input $x \in \mathcal{I}_{\mathrm{yes}} \cup \mathcal{I}_{\mathrm{no}}$ to a sequence $(V_1, \cdots, V_{l+1})$, with V_j for $j \in [l]$

[3] A space-bounded Merlin-Arthur-type proof system, where the verifier operates with classical logspace, $O(\log n)$ random bits, and has *online access* to a $\mathrm{poly}(n)$-bit message, exactly characterizes NP [Lip90]. An exponentially up-scaled quantum counterpart, with *classical* messages, was also considered in [GR23], where the variant with unitary quantum *polynomial*-space verifier (implicitly allowing $\mathrm{poly}(n)$ random bits) precisely corresponds to NEXP.

[4] For a detailed discussion in classical non-deterministic settings, see [Gol08, Section 5.3.1], where the term "direct access" is called "offline access".

[5] Our definitions of QIPL and $\mathsf{QIP_UL}$ can be straightforwardly extended to the corresponding proof systems with an odd number of messages, as shown in Figure 4.1.

representing the verifier's actions at the $(2j-1)$-th turn, and V_{l+1} representing the verifier's action just before the final measurement. The primary difference between QIPL and $\mathsf{QIP_UL}$ proof systems lies in the verifier's action V_j for $j \in [l]$:

- In QIPL proof systems, each V_j is an *almost-unitary* quantum circuit that includes $O(\log n)$ intermediate measurements in the computational basis. For convenience, we apply the principle of deferred measurements (e.g., [NC10, Section 4.4]), transforming the circuit that implements V_j into an *isometric* quantum circuit with a newly introduced environment register E_j,[6] which is measured at the end of that turn, with the measurement outcome denoted by u_j, as illustrated in Figure 1.1. Furthermore, each environment register E_j remains private to the verifier and becomes inaccessible after the round that starts with the verifier's j-th action.
- In $\mathsf{QIP_UL}$ proof systems, each V_j is a unitary quantum circuit.

The prover's actions can be similarly described by unitary quantum circuits. A proof system $P \rightleftharpoons V$ is said to *accept* if, after the verifier performs V_{l+1} and measures the designated output qubit in the computational basis, the outcome is 1. Additionally, we require *a strong notion of uniformity* for the verifier's mapping: the description of the sequence $(V_1, \cdots, V_{l+1})$ must be computable by a single deterministic logspace Turing machine.[7] Lastly, for QIPL proof systems, we impose an additional restriction on *yes* instances: the distribution of intermediate measurement outcomes $u = (u_1, \cdots, u_l)$, conditioned on acceptance, must be *highly concentrated*. More precisely, let $\omega(V)|^u$ be the contribution of u to $\omega(V)$, where $\omega(V)$ is the maximum acceptance probability of $P \rightleftharpoons V$. Then, there must exist a u^* such that $\omega(V)|^{u^*} \geq c(n)$.

We denote m-turn space-bounded quantum interactive proof systems with completeness c and soundness s as $\mathsf{QIPL}_m[c,s]$, and their unitary variant as $\mathsf{QIP_UL}_m[c,s]$. In particular, we adopt the following notations, which naturally extend to $\mathsf{QIP_UL}$:

$$\mathsf{QIPL}_m := \mathsf{QIPL}_m[2/3, 1/3] \text{ and } \mathsf{QIPL} := \cup_{1 \leq m \leq \mathrm{poly}(n)} \mathsf{QIPL}_m.$$

In *constant*-turn scenarios, it is crucial to emphasize that the proof systems $\mathsf{QIPL}_{O(1)}[c,s]$ and $\mathsf{QIP_UL}_{O(1)}[c,s]$ can directly simulate each other, as the environment registers $\mathsf{E}_1, \cdots, \mathsf{E}_{O(1)}$ collectively holds $O(\log n)$ qubits.[8] Therefore, for simplicity, we define $\mathsf{QIPL}_{O(1)}[c,s]$ proof systems in which the verifier's actions are implemented by *unitary* quantum circuits.

Space-bounded (unitary) quantum interactive proofs. Our first theorem serves as a quantum analog of the classical work by Condon and Ladner [CL95]:[9]

Theorem 1.1 (Informal of Theorem 3.1)**.** $\mathsf{QIPL} = \mathsf{NP}$.

Interestingly, Theorem 1.1 suggests that the QIPL model can be viewed as the *weakest* model that encompasses space-bounded (private-coin) classical interactive proofs, as considered in [CL95]. Our definition of QIPL is motivated by the goal of defining a quantum counterpart that includes these classical proof systems, ensuring that soundness against classical messages

[6] An isometric quantum circuit utilizes $O(\log n)$ ancillary gates, with each ancillary gate introducing an ancillary qubit $|0\rangle$. For further details, please refer to Definition 2.8.

[7] A weaker notion of uniformity only requires that the description of each V_j can be individually computed by a deterministic logspace Turing machine. It is important to note that these distinctions do not arise in the time-bounded setting, as the composition of a polynomial number of deterministic polynomial-time Turing machines can be treated as a single deterministic polynomial-time Turing machine.

[8] This equivalence follows directly from the principle of deferred measurements. However, for constant-turn space-bounded quantum interactive proofs, allowing each verifier action to involve polynomially many intermediate measurements might increase the proof system's power beyond the unitary case. This is because current techniques for proving results such as $\mathsf{BQL} = \mathsf{BQ_UL}$ [FR21, GRZ21, GR22] do not directly apply in this context.

[9] More specifically, the NP containment in Theorem 1.1 holds for any $\mathsf{QIPL}_m[c,s]$ proof system satisfying $c(n) - s(n) \geq \mathrm{poly}(n)$ and $1 \leq m(n) \leq \mathrm{poly}(n)$.

also holds for quantum messages. This guarantee is typically achieved by measuring the prover's quantum messages and treating the outcomes as classical messages (e.g., [AN02, Claim 1]).

However, space-bounded *unitary* quantum interactive proofs ($\mathsf{QIP_UL}$), which denote the most natural space-bounded counterpart to quantum interactive proofs as defined in [KW00, Wat03a], do not directly achieve the stated soundness guarantee. Hence, $\mathsf{QIP_UL}$ may be computationally weaker than QIPL. Our second theorem characterizes the computational power of $\mathsf{QIP_UL}$:

Theorem 1.2 (Informal of Theorem 3.3 and Theorem 4.2)**.** *The following holds*:

$$\mathsf{SAC}^1 \cup \mathsf{BQL} \subseteq \mathsf{QIP_UL} \subseteq \cup_{c(n)-s(n)\geq 1/\operatorname{poly}(n)} \mathsf{QIPL}_{O(1)}[c,s] \subseteq \mathsf{P}.$$

Theorems 1.1 and 1.2 suggest that $\mathsf{QIP_UL}$ is indeed *weaker* than QIPL unless $\mathsf{P} = \mathsf{NP}$. Interestingly, this distinction from the unitary case arises even when each verifier action is slightly more powerful than a unitary quantum circuit. It is also noteworthy that the class SAC^1 is equivalent to LOGCFL [Ven91], which contains NL and is contained in AC^1.[10] Our third theorem, meanwhile, focuses on space-bounded quantum interactive proof systems with a constant number of messages:

Theorem 1.3 (Informal of Theorem 4.3)**.** *For any* $c(n) - s(n) \geq \Omega(1)$, $\mathsf{QIPL}_{O(1)}[c,s] \subseteq \mathsf{NC}$.

To compare with time-bounded classical or quantum interactive proofs, we summarize our three theorems in Table 1. Notably, our two models of space-bounded quantum interactive proofs, QIPL and $\mathsf{QIP_UL}$, demonstrate behavior that is distinct from both:

- For (time-bounded) classical interactive proofs, all proof systems with $m \leq O(1)$ (the regime of the last row in Table 1) are contained in the second level of the polynomial-time hierarchy [Bab85, GS86], whereas the class of proof systems with $m = \operatorname{poly}(n)$ (the regime of the second and third rows in Table 1) exactly characterizes PSPACE [LFKN92, Sha92].
- For (time-bounded) quantum interactive proofs, all proof systems with parameters listed in Table 1 precisely capture PSPACE [Wat03b, KW00, JJUW11].

	Models	Constant gap $c(n) - s(n) \geq \Omega(1)$	Polynomial small gap $c(n) - s(n) \geq 1/\operatorname{poly}(n)$
The number of messages: $m(n) = \operatorname{poly}(n)$	QIPL	NP Theorem 1.1	NP Theorem 1.1
The number of messages: $m(n) = \operatorname{poly}(n)$	$\mathsf{QIP_UL}$	contains $\mathsf{SAC}^1 \cup \mathsf{BQL}$ & in P Theorem 1.2	contains $\mathsf{SAC}^1 \cup \mathsf{BQL}$ & in P Theorem 1.2
The number of messages: $3 \leq m(n) \leq O(1)$	QIPL & $\mathsf{QIP_UL}$	in NC Theorem 1.3	contains $\mathsf{SAC}^1 \cup \mathsf{BQL}$ & in P Theorem 1.2

Table 1: The computational power of QIPL and $\mathsf{QIP_UL}$ with different parameters.

Space-bounded unitary quantum statistical zero-knowledge. We also introduce *(honest-verifier) space-bounded unitary quantum statistical zero-knowledge*, denoted as $\mathsf{QSZK_UL_{HV}}$. This term refers to a specific form of space-bounded quantum proofs that possess statistical zero-knowledge against an honest verifier. Specifically, a space-bounded unitary quantum interactive proof system possesses this zero-knowledge property if there exists a quantum logspace simulator that approximates the snapshot states ("the verifier's view") on the registers M and W after each turn of this proof system, where each state approximation must be very close ("indistinguishable") to the corresponding snapshot state with respect to the trace distance.

[10] For more details on the computational power of SAC^1 and related complexity classes, see Section 2.4.

Our definition $\mathsf{QSZK_UL_{HV}}$ serves as a space-bounded variant of honest-verifier (unitary) quantum statistical zero-knowledge, denoted by $\mathsf{QSZK_{HV}}$, as introduced in [Wat02]. Our fourth theorem establishes that the statistical zero-knowledge property completely negates the computational advantage typically gained through the interaction:

Theorem 1.4 (Informal of Theorem 5.2)**.** $\mathsf{QSZK_UL} = \mathsf{QSZK_UL_{HV}} = \mathsf{BQL}$.

In addition to $\mathsf{QSZK_UL_{HV}}$, we can define $\mathsf{QSZK_UL}$ in line with [Wat09b], particularly considering space-bounded unitary quantum statistical zero-knowledge against *any verifier* (rather than an honest verifier). Following this definition, $\mathsf{BQL} \subseteq \mathsf{QSZK_UL} \subseteq \mathsf{QSZK_UL_{HV}}$. Interestingly, Theorem 1.4 serves as a direct space-bounded counterpart to $\mathsf{QSZK} = \mathsf{QSZK_{HV}}$ [Wat09b].

The intuition behind Theorem 1.4 is that the snapshot states after each turn capture all the essential information in the proof system, such as allowing optimal prover strategies to be "recovered" from these states [MY23, Section 7]. In space-bounded scenarios, space-efficient quantum singular value transformation [LGLW23] enables fully utilizing this information.

Finally, we emphasize that our consideration of this zero-knowledge property is purely complexity-theoretic. A full comparison with other notions of (statistical) zero-knowledge is beyond this scope. For more on classical and quantum statistical zero-knowledge, see [Vad99] and [VW16, Chapter 5].

1.2 Proof techniques

Our proof techniques are mostly inspired by established methods for standard quantum interactive proofs, while the nature of QIPL and $\mathsf{QIP_UL}$ necessitates certain adaptations of these techniques. We will highlight the challenges that arise and briefly explain how we address them.

1.2.1 Upper bounds for QIPL and $\mathsf{QIP_UL}$

$\mathsf{QIPL_{O(1)}} \subseteq \mathsf{P}$**.** We establish this inclusion using a semi-definite program (SDP) for a given $\mathsf{QIPL_{O(1)}}$ proof system, adapted from the SDP formulation for QIP in [VW16, Wat16]. Together with the turn-halving lemma, specifically Theorem 1.5(3), this inclusion implies that $\mathsf{QIP_UL} \subseteq \mathsf{P}$.

Consider a $(2l)$-turn $\mathsf{QIPL_{O(1)}}$ proof system $P \rightleftharpoons V$, where $l \leq O(1)$. Let $\rho_{\mathtt{M}_j\mathtt{W}_j}$ and $\rho_{\mathtt{M}'_j\mathtt{W}_j}$, for $j \in [l]$, denote snapshot states in the register M and W after the $(2j-1)$-st turn and the $(2j)$-th turn in $P \rightleftharpoons V$, respectively, as illustrated in Figure 3.1. The variables in this SDP correspond to these snapshot states after each prover's action, particularly $\rho_{\mathtt{M}'_j\mathtt{W}_j}$ for $j \in [l]$, while the objective function is the maximum acceptance probability $\omega(V)$ of $P \rightleftharpoons V$. Since the verifier's actions are *unitary* circuits, these variables can be treated independently. Hence, the SDP program mainly consists of two types of constraints, assuming that all variables are valid quantum states:

(i) Verifier's actions only operate on the registers M and W:

$$\rho_{\mathtt{M}_j\mathtt{W}_j} = V_j \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}} V_j^\dagger \text{ for } j \in \{2, \cdots, l\}, \text{ and } \rho_{\mathtt{M}_1\mathtt{W}_1} = V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathsf{MW}} V_1^\dagger.$$

(ii) Prover's actions do not change the verifier's private register:

$$\mathrm{Tr}_{\mathtt{M}_j}(\rho_{\mathtt{M}_j\mathtt{W}_j}) = \mathrm{Tr}_{\mathtt{M}'_j}(\rho_{\mathtt{M}'_j\mathtt{W}_j}) \text{ for } j \in [l]. \tag{1.1}$$

Since the variables in this SDP collectively hold $O(\log n)$ qubits, a standard SDP solver (e.g., [GM12]) provides a deterministic polynomial-time algorithm for approximately solving it.

$\mathsf{QIPL} \subseteq \mathsf{NP}$**.** We now extend the above SDP formulation to l-round QIPL proof systems, where the verifier's j-th action V_j is an *almost-unitary* quantum circuit that allows $O(\log n)$ intermediate measurements. We treat V_j as an isometric quantum circuit, introducing a new environment register E_j that is measured at the end of the turn, with the outcome denoted by u_j.

Note that $\omega(V)|^u$ denotes the contribution of the measurement outcomes $u = (u_1, \cdots, u_l)$ to the maximum acceptance probability $\omega(V)$. Since $\omega(V)$ can be expressed as a sum over all $\omega(V)|^u$, the soundness condition $\omega(V) \leq s(n)$ implies that each $\omega(V)|^u \leq s(n)$. With the completeness condition ensuring the existence of u^* such that $\omega(V)|^{u^*} \geq c(n)$, we can focus on a specific measurement outcome u and the *unnormalized* snapshot states $\rho_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathtt{E}_j}$ after measuring $\mathtt{E}_j$ in this SDP formulation. Building on the SDP formulation of $\mathsf{QIPL}_{\mathrm{O}(1)}$ proof systems, we obtain a family of SDP programs depending on the measurement outcomes $\{u\}$. Given a specific u, the SDP program includes two types of constraints:

(i') $\rho_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathtt{E}_j} = \big(I_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathtt{E}_j}\big) V_j \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}} V_j^\dagger$ for $j \in \{2, \cdots, l\}$, and

$\rho_{\mathtt{M}_1\mathtt{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathtt{E}_1} = \big(I_{\mathtt{M}_1\mathtt{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathtt{E}_1}\big) V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathsf{MW}} V_1^\dagger$.

(ii') $\mathrm{Tr}_{\mathtt{M}_j}(\rho_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathtt{E}_j}) = \mathrm{Tr}_{\mathtt{M}'_j}(\rho_{\mathtt{M}'_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathtt{E}_j})$ for $j \in [l]$.

Next, we explain the NP containment. The classical witness w consists of an l-tuple u, indicating a specific SDP program, and a feasible solution $(\rho_{\mathtt{M}'_1\mathtt{W}_1}, \cdots, \rho_{\mathtt{M}'_l\mathtt{W}_l})$ to this SDP program. This solution can be represented by l square matrices of dimension $\mathrm{poly}(n)$, thus having polynomial size. The verification procedure involves checking (1) whether the solution encoded in w satisfies these SDP constraints based on u; and (2) whether $\omega(V)|^u \geq c(n)$. All these checks can be verified using basic matrix operations in deterministic polynomial time.

1.2.2 Basic properties for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$

We begin by outlining three basic properties of space-bounded (unitary) quantum interactive proof systems, which are dependent on the parameters $c(n)$, $s(n)$, and $m(n)$:

Theorem 1.5 (Properties for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, informal of Theorem 3.2 and Lemma 4.5)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\mathrm{poly}(n)$, and $1 \leq m(n) \leq \mathrm{poly}(n)$. Then, it holds that*:

(1) **Closure under perfect completeness**.

$\mathsf{QIPL}_m[c, s] \subseteq \mathsf{QIPL}_{m+2}[1, 1 - (c-s)^2/2]$ *and* $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m+2}[1, 1 - (c-s)^2/2]$.

(2) **Error reduction**. *For any polynomial $k(n)$,*

$$\mathsf{QIPL}_m[c, s] \subseteq \mathsf{QIPL}_{m'}\left[1, 2^{-k}\right] \text{ and } \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m'}\left[1, 2^{-k}\right].$$

Here, $m' := O\big(km/\log \frac{1}{1-(c-s)^2/2}\big)$.

(3) **Parallelization**. $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{4m+1}[1, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{2m+1}[1, (1+\sqrt{s})/2]$.

Achieving perfect completeness for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof systems, particularly Theorem 1.5(1), can be adapted from the techniques used in QIP proof systems [VW16, Section 4.2.1] (or [KW00, Section 3]) by adding two additional turns. However, there are important subtleties to consider when establishing the other properties in Theorem 1.5.

Error reduction via sequential repetition. Since each message is of size $O(\log n)$, error reduction via *parallel repetition* does not apply to QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ when the gap $c - s$ is polynomially small, regardless of the number of messages.[11] Alternatively, error reduction via *sequential repetition* requires that the registers M and W (the "workspace") must be in the all-zero state ("cleaned") before each execution of the original proof systems. While this is trivial for QIP proof systems, it poses a challenge for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof systems because the (almost-)unitary quantum logspace verifier cannot achieve this on its own.

[11] Still, error reduction via parallel repetition works for QIPL when the gap $c - s \geq \Omega(1)$; see Lemma 4.4.

To establish Theorem 1.5(2), our solution is to have *the prover "clean" the workspace* while ensuring that the prover behaves honestly. This is achieved through the following proof system: The verifier applies a multiple-controlled adder before each proof system execution, with the adder being activated only when the control qubits are all zero. The verifier then measures the register that the adder acts on and accepts if (1) the workspace is "cleaned" for each execution and (2) *all* outcomes of the original proof system executions are acceptance.

Parallelization and strict uniformity condition for the verifier's mapping. The original parallelization technique proposed in [KW00, Section 4] applies only to $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ (also QIPL) proof systems with a constant number of messages. This limitation stems from the requirement that the prover sends the snapshot states for all m turns in a single message. As m increases, the size of this message grows to $O(m \log n)$, which becomes $\omega(\log n)$ when $m = \omega(1)$.

To overcome this issue, we adapt the technique from [KKMV09, Section 4], a "dequantized" version of the original approach that fully utilizes the *reversibility* of the verifier's actions. Instead of sending all snapshot states in one message, the new verifier performs the original verifier's action or its reverse at any turn in a single action. Specifically, when applying this method to a $(4m+1)$-turn $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof system $P \rightleftharpoons V$, the prover starts by sending only the snapshot state after the $(2m+1)$-st turn. The verifier then chooses $b \in \{0,1\}$ uniformly at random: if $b = 0$, the verifier continues to interact with the prover according to $P \rightleftharpoons V$, keeping the acceptance condition unchanged; while if $b = 1$, the verifier executes $P \rightleftharpoons V$ in reverse, and finally accepts if its private qubits are all zero. This proof system, which halves the number of turns, is referred to as the *turn-halving lemma*, as detailed in Theorem 1.5(3).

Next, we establish Theorem 1.2 by applying the turn-halving lemma $O(\log n)$ times.[12] Specifically, any $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof system with a polynomial number of messages can be parallelized to three messages,[13] while the gap $c - s$ of the resulting proof system becomes polynomially small. However, this reasoning poses a challenge: the resulting verifier must know all original verifier actions, necessitating a strong notion of uniformity for the verifier's mapping in our definition of $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$. In addition, to prove Theorem 1.3, we adopt a similar approach to that used for QIP, particularly $\mathsf{QIP}[3] \subseteq \mathsf{QMAM}$ [MW05], which inspired the turn-halving lemma [KKMV09, Section 4], and an exponentially down-scaling version of the work [JJUW11].

1.2.3 Lower bounds for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$

$\mathsf{NP} \subseteq \mathsf{QIPL}$. This inclusion draws inspiration from the interactive proof system in [CL95, Lemma 2] and presents a challenge in adapting this proof system to the QIPL setting.

We start by outlining this QIPL proof system for 3-SAT. Consider a 3-SAT formula

$$\phi = C_1 \vee C_2 \vee C_3 = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_4 \vee \neg x_2 \vee x_3) \wedge (x_4 \vee \neg x_1 \vee \neg x_3)$$

with $k = 3$ clauses and $n = 4$ variables. An assignment α of ϕ assigns each variable x_j for $j \in [n]$ a value α_j of either $\top$ (true) or $\bot$ (false). To verify whether ϕ is satisfied by the assignment α, we encode $\phi(\alpha)$ as $\mathrm{Enc}(\phi(\alpha))$, consisting of $3k$ triples (l, i, v), where l denotes the literal (either x_j or $\neg x_j$), i represents the i-th clause, and v denotes the value assigned to l. The prover's actions are divided into two phases:

(i) Consistency Check (for variables). The prover sends one by one all the triples (l, i, v) in $\mathrm{Enc}(\phi(\alpha))$, ordered by the variable $\mathrm{var}(l)$ corresponding to the literal l;

[12] An operation based on r random bits can be simulated by a corresponding unitary controlled by the state $|+\rangle^{\otimes r}$, where $|+\rangle := \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Thus, simulating $O(\log n)$ random bits across all turns of the proof system requires $O(\log n)$ ancillary qubits in total, which is feasible for the unitary quantum logspace verifier in $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$.

[13] Although the turn-halving lemma does not directly apply to QIPL proof systems, a similar reasoning works for its reversible generalization $\mathsf{QIPL}^{\diamond}$, reducing a constant number of messages to three.

(ii) Satisfiability Check (for clauses). For each $i \in \{1, \ldots, k\}$, the prover sends the three triples (l_1, i, v_1), (l_2, i, v_2), and (l_3, i, v_3) in $\mathrm{Enc}(\phi(\alpha))$.

The verifier's actions are as follows. To prevent the prover from entangling with the verifier and revealing the private coins, the verifier measures the received messages in the computational basis at the beginning of each action, interpreting the measurement outcomes as the prover's messages. Therefore, it suffices to establish soundness *against classical messages*.

We now focus on this specific proof system. In Phase (i), the verifier checks whether the assigned values to the same variable are consistent. Since the verifier's actions are almost-unitary circuits and *cannot discard information*, this seems challenging. Our solution is that the verifier keeps only the current and the previous triples, returning the previous triple to the prover in the next turn. In Phase (ii), the verifier checks whether each batch of three triples is satisfied and returns them immediately. Lastly, to ensure that the multisets of triples from Phase (i) and (ii) are identical, the verifier computes the "fingerprint" of these multisets,[14] triple by triple, and compares the fingerprints from both phases at the end. The verifier accepts if all checks succeed.

Using the fingerprinting lemma [Lip90], we prove the correctness of this proof system, showing that 3-SAT $\in \mathsf{QIPL}_{8k}[1, 1/3]$. Interestingly, when combined with the inclusion $\mathsf{QIPL} \subseteq \mathsf{NP}$, this protocol suggests an alternative approach to (indirect) error reduction for QIPL.

$\mathsf{SAC}^1 \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}$. This inclusion is inspired by the interactive proof system in [For89, Section 3.4]. By using error reduction for $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, specifically Theorem 1.5(2), it remains to demonstrate that $\mathsf{SAC}^1 \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}[1, 1 - 1/\operatorname{poly}(n)]$. A Boolean circuit is defined as a (uniform) SAC^1 circuit C if it is an $O(\log n)$-depth Boolean circuit that employs unbounded fan-in OR gates, bounded fan-in AND gates, and negation gates at the input level.

The interactive proof system for evaluating the circuit C starts at its top gate. If the gate is an OR, the prover selects a child gate; if it's an AND, the verifier flips a coin to select one. This process repeats until reaching an input x_i or its negation, with the verifier accepting if $x_i = 1$ or $x_i = 0$, respectively. Since the computational paths in C do not interfere, extending soundness against classical messages, following directly from [For89, Section 3.4], to quantum messages can be done by measuring the registers M and W in the computational basis at the end of the verifier's last turn. Finally, given that C has $O(\log n)$ depth, implementing the verifier's actions requires only $O(\log n)$ ancillary qubits, which is indeed achievable by a unitary verifier.

1.2.4 The equivalence of $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}$ and BQL

We demonstrate Theorem 1.4 by introducing a $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-complete problem:

Theorem 1.6 (Informal of Theorem 5.3)**.** IndivProdQSD *is* $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$*-complete.*

We begin by informally defining the promise problem Individual Product State Distinguishability, denoted by IndivProdQSD$[k(n), \alpha(n), \delta(n)]$, where the parameters satisfy $\alpha(n) - k(n) \cdot \delta(n) \geq 1/\operatorname{poly}(n)$ and $1 \leq k(n) \leq \operatorname{poly}(n)$. This problem considers two k-tuples of $O(\log n)$-qubit quantum states, denoted by $\sigma_1, \cdots, \sigma_k$ and $\sigma'_1, \cdots, \sigma'_k$, where the purifications of these states can be prepared by corresponding polynomial-size unitary quantum circuits acting on $O(\log n)$ qubits. For *yes* instances, these two k-tuples are "globally" far, satisfying

$$\mathrm{T}\big(\sigma_1 \otimes \cdots \otimes \sigma_k, \sigma'_1 \otimes \cdots \otimes \sigma'_k\big) \geq \alpha. \tag{1.2}$$

While for *no* instances, each pair of corresponding states in these k-tuples are close, satisfying

$$\forall j \in [k], \quad \mathrm{T}\big(\sigma_j, \sigma'_j\big) \leq \delta. \tag{1.3}$$

Then we show that (1) the complement of IndivProdQSD, $\overline{\text{IndivProdQSD}}$, is $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-hard; and (2) IndivProdQSD is in BQL, which is contained in $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$ by definition.

[14] See Section 2.4 for the definition of the fingerprint of a multiset. The computation of each fingerprint requires $O(\log n)$ random bits, which can be simulated in a QIPL proof system; see Footnote 12 for details.

$\overline{\text{IndivProdQSD}}$ **is $\mathsf{QSZK_UL_{HV}}$-hard.** The hardness proof draws inspiration from [Wat02, Section 5]. Consider a $\mathsf{QSZK_UL_{HV}}[2k, c, s]$ proof system, denoted by $\mathcal{B}$. The logspace-bounded simulator $S_{\mathcal{B}}$ produces good state approximations ξ_j and ξ'_j of the snapshot states $\rho_{\mathtt{M}_j\mathtt{W}_j}$ and $\rho_{\mathtt{M}'_j\mathtt{W}_j}$ after the $(2j-1)$-st turn and the $(2j)$-th turn in $\mathcal{B}$, respectively, satisfying $\xi_j \approx_\delta \rho_{\mathtt{M}_j\mathtt{W}_j}$ and $\xi'_j \approx_\delta \rho_{\mathtt{M}'_j\mathtt{W}_j}$, where $\delta_{\mathcal{B}}(n)$ is a negligible function.

Since the verifier's actions are unitary and the verifier is honest, it suffices to check that the prover's actions do not change the verifier's private register, corresponding to the type (ii) constraints Equation (1.1) in the SDP formulation for QIPL proof systems. For convenience, let $\sigma_j \coloneqq \mathrm{Tr}_{\mathtt{M}_j}(\xi_j)$ and $\sigma'_j \coloneqq \mathrm{Tr}_{\mathtt{M}'_j}(\xi'_j)$ for $j \in [k]$. We then establish $\mathsf{QSZKL_{HV}}$ hardness as follows:

- For *yes* instances, the message-wise closeness condition of the simulator $S_{\mathcal{B}}$ implies Equation (1.3) with $\delta(n) \coloneqq 2\delta_{\mathcal{B}}(n)$.
- For *no* instances, the simulator $S_{\mathcal{B}}$ produces the snapshot state before the final measurement, which accepts with probability $c(n)$ for all instances, while the proof system accepts with probability at most $s(n)$. The inconsistency between the simulator's state approximations and the snapshot states yields Equation (1.2) with $\alpha(n) \coloneqq (\sqrt{c} - \sqrt{s})^2/4(l-1)$.

$\text{IndivProdQSD} \in \mathsf{BQL}$**.** Since it holds that $\mathsf{BQL} = \mathsf{QMAL}$ [FKL+16, FR21], it suffices to establish that $\text{IndivProdQSD} \in \mathsf{QMAL}$. By applying an averaging argument in combination with Equation (1.2), we derive the following:

$$\sum_{j\in[k]} \mathrm{T}\big(\sigma_j, \sigma'_j\big) \geq \mathrm{T}\big(\sigma_1 \otimes \cdots \otimes \sigma_k, \sigma'_1 \otimes \cdots \otimes \sigma'_k\big) \geq \alpha \quad \Rightarrow \quad \exists j \in [k] \text{ s.t. } \mathrm{T}\big(\sigma_j, \sigma'_j\big) \geq \frac{\alpha}{k}. \tag{1.4}$$

The QMAL protocol works as follows: (1) The prover sends an index $i \in [k]$ to the verifier; and (2) The verifier accepts if $\mathrm{Tr}(\sigma_i, \sigma'_i) \geq \alpha/k$ and rejects if $\mathrm{Tr}(\sigma_i, \sigma'_i) \leq \delta$, in accordance with Equation (1.4) and Equation (1.3). The resulting promise problem to be verified is precisely an instance of $\text{GapQSD}_{\log}$, which is known to be BQL-complete [LGLW23].

1.3 Discussion and open problems

We introduce two models of space-bounded quantum interactive proof systems: QIPL and $\mathsf{QIP_UL}$. Unlike $\mathsf{BQL} = \mathsf{BQ_UL}$, we show that $\mathsf{QIP_UL} \subsetneq \mathsf{QIPL}$ unless $\mathsf{P} = \mathsf{NP}$. Our results highlight the distinctive role of intermediate measurements in space-bounded quantum interactive proofs, setting them apart from space-bounded quantum computation. This prompts an intriguing question:

(a) What is the computational power of space-bounded quantum interactive proofs beyond QIPL, specifically when allowing a general quantum logspace verifier?

A motivating example is a reversible generalization of QIPL, particularly space-bounded *isometric* quantum interactive proof systems ($\mathsf{QIPL}^\diamond$, see Remark 3.7), where all verifier actions are space-bounded *isometric* quantum circuits.[15] Notably, $\mathsf{QMA} \subseteq \mathsf{QIPL}^\diamond$.[16] Given a local Hamiltonian $H = \sum_{i=1}^m H_i$, we can construct a $\mathsf{QIPL}^\diamond$ proof system as follows:

(i) The verifier chooses a local term H_i uniformly at random from the set $\{H_1, \cdots, H_m\}$.

(ii) The prover sends a ground state $|\Omega\rangle$ qubit by qubit, while the verifier sends a state $|0\rangle$ in each round and retains only the qubits associated with H_i in its private registers.

(iii) The verifier performs the POVM corresponding to the decomposition $I = H_i + (I - H_i)$.[17]

[15] An isometric quantum circuit is a generalization of a unitary quantum circuit that allows ancillary gates, each introducing an ancillary qubit $|0\rangle$. See Definition 2.8 for a formal definition.

[16] A similar approach is used in a streaming version of QMAL (with online access to the message) in [GR23].

[17] See the proof of [KSV02, Proposition 14.2] for an explicit construction of such POVMs.

Further analysis indicates that the verifier accepts with probability $1 - m^{-1}\langle\Omega|H|\Omega\rangle$, and direct sequential repetition yields a $\mathsf{QIPL}^{\diamond}$ proof system. Additionally, it is evident that all candidate models of Question (a) are contained in QIP, and thus in PSPACE.

Furthermore, space-bounded *unitary* quantum interactive proofs ($\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$) can simulate the classical counterparts with $O(\log n)$ public coins [For89] (see Theorem 1.2), raising the question:

(b) Can we achieve a tighter characterization of $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$? For example, does $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ contain space-bounded classical interactive proofs with $\omega(\log n)$ public coins?

In addition to $\mathsf{QIPL}^{\diamond}$ and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, QIPL exactly characterizes NP (see Theorem 1.1). However, the high concentration requirement for *yes* instances (the completeness condition) in the definition of QIPL appears not entirely natural. This raises an interesting question:

(c) What is the computational power of the variant of QIPL without this high concentration requirement in the completeness condition?

Finally, for *constant*-turn space-bounded quantum interactive proofs, the three models discussed here become equivalent due to the principle of deferred measurements, contrasting with the aforementioned polynomial-turn settings. However, this equivalence does not directly extend to more general verifiers (see Footnote 8), leading to the following question:

(d) What is the computational power of constant-turn space-bounded quantum interactive proofs with a general quantum logspace verifier?

1.4 Related works

Several variants of (time-bounded) quantum interactive proofs with short messages were explored in [BSW11, Per12]. These variants are as powerful as QMA or BQP, depending on the specific settings. Recently, a space-bounded classical Merlin-Arthur-type proof system was proposed in [GRZ24] that exactly characterizes BQL. This setting is very similar to [Lip90] (see Footnote 3), but here, the honest prover's power is limited to quantum logspace.

The concept of interactive proof systems has been extended to other computational models. Quantum interactive proofs for synthesizing quantum states, known as $\mathsf{stateQIP}$, were introduced in [RY22]. Follow-up research established the equivalence $\mathsf{stateQIP} = \mathsf{statePSPACE}$ [MY23] and developed a parallelization technique for $\mathsf{stateQIP}$ [INN+22, Ros24]. A Merlin-Arthur-type variant was also explored in [DLGLM23, DLG24]. More recently, quantum interactive proofs for unitary synthesis and related problems have been studied in [BEM+23, LMW24]. Another interesting but less related variant is the exploration of interactive proof systems in distributed computing [KOS18, NPY20], and more recently, quantum distributed interactive proof systems have been investigated [FPLGN21, LGMN23, HKN24].

Finally, space-bounded (classical) statistical zero-knowledge, where the verifier has *read-only (i.e., two-way) access* to (polynomial-length) messages during interactions, was studied in [DGRV11, AHT23, AGM+23]. More recently, a variant where the verifier has *online (i.e., one-way) access* to messages has also been explored [CDGH24].

2 Preliminaries

We assume that the reader is familiar with quantum computation and the theory of quantum information. For an introduction, the textbooks by [NC10] and [dW19] provide a good starting point, while for a more comprehensive survey on quantum complexity theory, refer to [Wat09a].

We introduce several conventions throughout the paper: (1) we denote $[n] \coloneqq \{1, 2, \ldots, n\}$; (2) we use the logarithmic function $\log(x)$ with base 2; and (3) we utilize the notation $|\bar{0}\rangle$ to represent $|0\rangle^{\otimes a}$ with $a > 1$. In addition to these conventions, we provide two useful definitions. We say that $\mathcal{I} = (\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$ is a *promise problem*, if it satisfies that $\mathcal{I}_{\mathrm{yes}} \cap \mathcal{I}_{\mathrm{no}} = \emptyset$ and $\mathcal{I}_{\mathrm{yes}} \cup \mathcal{I}_{\mathrm{no}} \subseteq$

$\{0,1\}^*$. For simplicity, we use the abbreviation $x \in \mathcal{I}$ to denote $x \in \mathcal{I}_{\text{yes}} \cup \mathcal{I}_{\text{no}}$. A function $\mu(n)$ is said to be *negligible*, if for every integer $c \geq 1$, there is an integer $n_c > 0$ such that for all $n \geq n_c$, $\mu(n) < n^{-c}$.

2.1 Distance-like measures for quantum states

We will provide an overview of relevant quantum distances and divergences, along with useful inequalities among different quantum distance-like measures. We say that a square matrix ρ is a *quantum state* if ρ is positive semi-definite and $\mathrm{Tr}(\rho) = 1$.

Definition 2.1 (Trace distance and fidelity)**.** *For any quantum states ρ_0 and ρ_1, we define two distance-like measures:*

- **Trace distance**. $\mathrm{T}(\rho_0, \rho_1) \coloneqq \frac{1}{2}\mathrm{Tr}|\rho_0 - \rho_1| = \frac{1}{2}\mathrm{Tr}(((\rho_0 - \rho_1)^\dagger(\rho_0 - \rho_1))^{1/2})$.
- **(Uhlmann) Fidelity**. $\mathrm{F}(\rho_0, \rho_1) \coloneqq \mathrm{Tr}|\sqrt{\rho_0}\sqrt{\rho_1}|$.

We begin by listing two useful bounds on tensor-product quantum states with respect to the trace distance:

Lemma 2.2 (Trace distance on tensor-product states, adapted from Exercise 9.1.2 and Corollary 9.1.10 in [Wil13])**.** *For any quantum states $\rho_1 \otimes \cdots \otimes \rho_k$ and $\rho'_1 \otimes \cdots \otimes \rho'_k$, where ρ_i and ρ'_i use the same number of qubits for all $i \in [k]$, it holds that*

(1) $\forall i \in [k]$, $\mathrm{T}(\rho_i, \rho'_i) \leq \mathrm{T}(\rho_1 \otimes \cdots \otimes \rho_k, \rho'_1 \otimes \cdots \otimes \rho'_k)$.

(2) $\mathrm{T}(\rho_1 \otimes \cdots \otimes \rho_k, \rho'_1 \otimes \cdots \otimes \rho'_k) \leq \sum_{i\in[k]} \mathrm{T}(\rho_i, \rho'_i)$.

We then provide two fundamental properties of the trace distance.

Lemma 2.3 (Data-processing inequality for the trace distance, adapted from [NC10, Theorem 9.2])**.** *Let ρ_0 and ρ_1 be quantum states. For any quantum channel $\mathcal{E}$, it holds that*

$$\mathrm{T}(\mathcal{E}(\rho_0), \mathcal{E}(\rho_1)) \leq \mathrm{T}(\rho_0, \rho_1).$$

Lemma 2.4 (Unitary invariance for the trace distance, adapted from [NC10, Equation (9.21)])**.** *Let ρ_0 and ρ_1 be quantum states. For any unitary transformation U, it holds that*

$$\mathrm{T}\left(U\rho_0 U^\dagger, U\rho_1 U^\dagger\right) = \mathrm{T}(\rho_0, \rho_1).$$

Next, we present two basic properties for the fidelity.

Lemma 2.5 (Data-processing inequality for the fidelity, adapted from Theorem 9.6 in [NC10])**.** *Let ρ_0 and ρ_1 be quantum states. For any quantum channel $\mathcal{E}$, it holds that*

$$\mathrm{F}(\mathcal{E}(\rho_0), \mathcal{E}(\rho_1)) \geq \mathrm{F}(\rho_0, \rho_1).$$

Lemma 2.6 ([SR01, Lemma 2] & [NS03, Lemma 3.3])**.** *Let ρ_0 and ρ_1 be m-qubit quantum states. Then, for any m-qubit quantum state ξ, it holds that*

$$\mathrm{F}(\rho_0, \xi)^2 + \mathrm{F}(\xi, \rho_1)^2 \leq 1 + \mathrm{F}(\rho_0, \rho_1).$$

Lastly, we present a lemma concerning the freedom in purifications of quantum states:

Lemma 2.7 (Unitary equivalence of purifying the same state, adapted from [NC10, Exercise 2.81])**.** *Let $|\psi\rangle_{\mathsf{AB}}$ and $|\phi\rangle_{\mathsf{AB}}$ be pure states on the registers A and B such that*

$$\mathrm{Tr}_{\mathsf{B}}(|\psi\rangle\langle\psi|_{\mathsf{AB}}) = \rho_{\mathsf{A}} = \mathrm{Tr}_{\mathsf{B}}(|\phi\rangle\langle\phi|_{\mathsf{AB}}),$$

where ρ_A is a mixed state on the register A. Then, there exists a unitary transformation U_{B} acting on the register B such that $|\psi\rangle_{\mathsf{AB}} = (I_{\mathsf{A}} \otimes U_{\mathsf{B}})|\phi\rangle_{\mathsf{AB}}$.

2.2 Space-bounded quantum computation

We say that a function $s(n)$ is *space-constructible* if there exists a deterministic space $s(n)$ Turing machine that takes 1^n as an input and outputs $s(n)$ in the unary encoding. Moreover, we say that a function $f(n)$ is $s(n)$-*space computable* if there exists a deterministic space $s(n)$ Turing machine that takes 1^n as an input and outputs $f(n)$. Our definitions of space-bounded quantum computation are formulated in terms of *quantum circuits*. For a discussion on the equivalence between space-bounded quantum computation using *quantum circuits* and *quantum Turing machines*, we refer readers to [FL18, Appendix A] and [FR21, Section 2.2].

We begin by introducing three types of space-bounded quantum circuit families, as formalized in Definition 2.8. Our definitions align with [VW16, Section 2.3]. Throughout this work, we adopt the shorthand notation C_x to indicate that the circuit $C_{|x|}$ takes input x.

Definition 2.8 (Space-bounded quantum circuit families: unitary, almost-unitary, and isometric)**.** *Let us define three types of quantum circuits:*

- **Unitary quantum circuit**. *A unitary quantum circuit consists of a sequence of unitary quantum gates, each of which belongs to some fixed gate set that is universal for quantum computation, such as* $\{\mathrm{H}, \mathrm{CNOT}, \mathrm{T}\}$.
- **Almost-unitary quantum circuit**. *An almost-unitary quantum circuit generalizes a unitary quantum circuit acting on* $O(s(n))$ *qubits by allowing* $O(s(n))$ *single-qubit measurement gates* M *in the computational basis, defined as:*

$$\Phi^{\mathrm{M}}(\rho) \coloneqq |0\rangle\langle 0|\mathrm{Tr}(M_0\rho) + |1\rangle\langle 1|\mathrm{Tr}(M_1\rho), \text{ where } M_b \coloneqq |b\rangle\langle b| \text{ for } b \in \{0,1\}.$$

- **Isometric quantum circuit**. *An isometric quantum circuit extends a unitary quantum circuit acting on* $O(s(n))$ *qubits by allowing* $O(s(n))$ *ancillary gates. An ancillary gate is a non-unitary gate that takes no input and produces a single qubit in the state* $|0\rangle$ *as output.*

For convenience, we treat almost-unitary quantum circuits as a special case of isometric quantum circuits.[18] *For a promise problem* $\mathcal{I} = (\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$*, a family of unitary, almost-unitary, or isometric quantum circuits* $\{C_x : x \in \mathcal{I}\}$ *is called* $s(n)$-space-bounded *if there is a deterministic Turing machine that, given any input* $x \in \mathcal{I}$ *with input length* $n \coloneqq |x|$*, runs in space* $O(s(n))$ *(and hence time* $2^{O(s(n))}$*) and outputs a description of* C_x*, where* C_x *accepts if* $x \in \mathcal{I}_{\mathrm{yes}}$*, rejects if* $x \in \mathcal{I}_{\mathrm{no}}$*, acts on* $O(s(n))$ *qubits, and consists of* $2^{O(s(n))}$ *gates.*

Remark 2.9 (Subtleties on space-bounded quantum circuit families)*.* In the context of space-bounded quantum circuits, as defined in Definition 2.8, there are important subtleties:

(1) Space-bounded almost-unitary quantum circuits are *oblivious* to the intermediate measurement outcomes, implying that qubits in these circuits cannot be directly reset to zero.

(2) Space-bounded unitary and almost-unitary quantum circuits are equivalent for promise problems via the principle of deferred measurements. However, such equivalences are unknown in more general settings, analogous to the scenario in Footnote 20.

In this work, we focus on (log)space-bounded quantum circuits with $s(n) = O(\log n)$. The complexity classes corresponding to space-bounded unitary and general quantum circuits with $s(n) = \Theta(\log(n))$ are known as $\mathsf{BQ_UL}$ and BQL, respectively. As described in [VW16, Section 2.3], a *general quantum circuit* extends a unitary quantum circuit by including ancillary gates and *erasure gates*.[19] It has been established that $\mathsf{BQ_UL} = \mathsf{BQL}$ for promise problems [FR21] (see

[18] More specifically, by applying the principle of deferred measurements (e.g., [NC10, Section 4.4]) to an almost-unitary quantum circuit, we obtain an isometric quantum circuit in which each measurement gate is simulated by an ancillary gate, and all ancillary qubits are measured at the end.

[19] An erasure gate is a non-unitary gate that takes a single qubit as input and produces no output. Alternatively, a general quantum circuit can also be defined by extending a unitary quantum circuit with measurement gates

also [GRZ21, GR22]), whereas such equivalences remain unproven in more general forms.[20] For detailed definitions and known properties of these classes, we refer to [LGLW23, Section 2.3] as a brief introduction.

Lastly, we define *logspace (many-to-one) reductions*. We begin by slightly abusing notation and considering parameterized promise problems of the form $\mathcal{I} = \{\mathcal{I}_{n,t_1,\cdots,t_r}\}_{n\in\mathbb{N}}$ for functions $t_1,\cdots,t_r\colon \mathbb{N}\to\mathbb{R}$, where $\mathcal{I}_{n,t_1,\cdots,t_r}$ consists of instances of size n which satisfy conditions expressed in terms of $t_1(n),\cdots,t_r(n)$. We say that $\mathcal{I} = \{\mathcal{I}_{n,t_1,\cdots,t_r}\}_{n\in\mathbb{N}}$ is (many-to-one) reducible to $\mathcal{I}' = \{\mathcal{I}'_{m,t'_1,\cdots,t'_{r'}}\}_{m\in\mathbb{N}}$ if there exist $(r+1)$-variable real polynomials $p_0,\cdots,p_{r'}$ such that for all $n\in\mathbb{N}$, there exists a function $g_n\colon \mathcal{I}_{n,t_1,\cdots,t_r}\to\mathcal{I}'_{m,t'_1,\cdots,t'_{r'}}$ satisfying the following conditions: (1) $m = p_0(n,t_1(n),\cdots,t_r(n))$; (2) $t'_j(m) = p_j(n,t_1(n),\cdots,t_r(n))$ for all $j\in[r']$; (3) $g_n(x)\in\mathcal{I}'_{m,t'_1,\cdots,t'_{r'}}$ for all $x\in\mathcal{I}_{n,t_1,\cdots,t_r}$. If the family of functions $\{g_n\}_{n\in\mathbb{N}}$ is computable in deterministic logspace, we say that $\mathcal{I}$ is logspace-reducible to $\mathcal{I}'$, denoted by $\mathcal{I}\leq^m_{\mathsf{L}}\mathcal{I}'$.

2.3 Space-bounded quantum state testing

We begin by defining the space-bounded quantum state testing problem with respect to the trace distance, denoted as $\textsc{GapQSD}_{\log}$:

Definition 2.10 (Space-bounded Quantum State Distinguishability Problem, adapted from Definition 4.1 and 4.2 [LGLW23])**.** *Let $\alpha(n)$, $\beta(n)$, $r(n)$ be logspace computable functions such that $0\leq\beta(n)<\alpha(n)\leq 1$, $\alpha(n)-\beta(n)\geq 1/\mathrm{poly}(n)$ and $1\leq r(n)\leq O(\log n)$. Let Q_0 and Q_1 be polynomial-size unitary quantum circuits acting on $O(\log n)$ qubits, with $r(n)$ specified output qubits. Here, n represents the total number of gates in Q_0 and Q_1. For $b\in\{0,1\}$, let ρ_b denote the quantum states obtained by running Q_b on the all-zero state $|\bar{0}\rangle$ and tracing out the non-output qubits, then the promise is that one of the following holds:*

- Yes *instances: A pair of quantum circuits (Q_0,Q_1) such that $\mathrm{T}(\rho_0,\rho_1)\geq\alpha(n)$;*
- No *instances: A pair of quantum circuits (Q_0,Q_1) such that $\mathrm{T}(\rho_0,\rho_1)\leq\beta(n)$.*

Moreover, we use the notation $\overline{\textsc{GapQSD}}_{\log}$ to denote the *complement* of $\textsc{GapQSD}_{\log}$ with respect to the chosen parameters $\alpha(n)$ and $\beta(n)$. As established in [LGLW23], $\textsc{GapQSD}_{\log}$ is BQL-complete, and we are particularly interested in the BQL containment:

Theorem 2.11 ($\textsc{GapQSD}_{\log}$ is in BQL, adapted from [LGLW23, Theorem 4.10])**.** *Let $\alpha(n)$ and $\beta(n)$ be logspace computable functions such that $\alpha(n)-\beta(n)\geq 1/\mathrm{poly}(n)$. It holds that*

$$\textsc{GapQSD}_{\log}[\alpha(n),\beta(n)]\in\mathsf{BQL}.$$

Lastly, it is worth noting that by removing the space constraints on the quantum circuits Q_0 and Q_1 and allowing $r(n)\leq n$, where n denotes the input length of these state-preparation circuits, we obtain a variant of Definition 2.10 that aligns with the definition of $\textsc{GapQSD}[\alpha(n),\beta(n)]$. This promise problem was considered in [Wat02] with the condition $\alpha^2>\beta$, referred to as $\mathrm{QSD}[\alpha(n),\beta(n)]$.

2.4 Classical concepts, tools, and complexity classes

3-SAT. The 3-SAT problem is one of the simplest examples of NP-complete problems. We provide only a brief introduction to 3-SAT here. For further details, see [AB09, Section 2.3].

and reset-to-zero gates, as in [FR21, GR22]. Notably, a reset-to-zero gate can be simulated by first applying an erasure gate to remove the original qubit and then using an ancillary gate to introduce a new qubit.

[20] Specifically, this refers to the transformation of a unitary quantum logspace circuit C' from a general quantum logspace circuit C (with all-zero states as input) such that the final state of C and C' are identical. This is a stronger requirement than merely ensuring that the output qubits of these circuits are the same. This general form only can be simulated in NC^2 [Wat99, Wat03a].

A 3-SAT formula can be written as $\phi = C_1 \wedge \cdots \wedge C_k$, where each clause C_i for $i \in [k]$ is of the form $\left(l_1^{(i)} \vee l_2^{(i)} \vee l_3^{(i)}\right)$, with each literal $l_j^{(i)}$ being either one of the variables $x_1, \cdots, x_n$ or its negation. For instance, $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_4 \vee \neg x_2 \vee x_3) \wedge (x_4 \vee \neg x_1 \vee \neg x_3)$ illustrates the structure. An assignment of a 3-SAT formula assigns each variable x_j for $j \in [n]$ a value of either $\top$ (true) or $\bot$ (false). The 3-SAT problem aims to decide whether a given formula ϕ is satisfiable. We say that ϕ is satisfiable if there exists an assignment α such that $\Phi(\alpha) = \top$.

Lemma 2.12 ([AB09, Exercise 4.6])**.** 3-SAT *is* NP*-complete under logspace reductions.*

Fingerprinting of multisets. A fingerprint of a multiset $\{x_1, \cdots, x_k\}$, where all elements are non-negative integers and duplicates are allowed, is defined as $\Pi_{i=1}^{k}(x_i + r) \mod p$, with p being a prime and $r \in [p-1]$. The fingerprinting lemma [Lip90] aims to compare whether two multisets are equal by using short fingerprints:

Lemma 2.13 (Fingerprinting lemma, adapted from [Lip90, Theorem 3.1])**.** *Let* $A := \{x_1, \cdots, x_{\ell_1}\}$ *and* $B := \{y_1, \cdots, y_{\ell_2}\}$ *be two multisets in which all elements are* b*-bit non-negative integers, with* $\ell := \max\{\ell_1, \ell_2\}$*. If the prime* p *is chosen uniformly at random from the interval* $[(b\ell)^2, 2(b\ell)^2]$ *and the integer* r *is chosen uniformly at random from the interval* $[1, p-1]$*, the probability that the* distinct *multisets* A *and* B *produce the* same *fingerprint is at most* $O\left(\frac{\log b + \log \ell}{b\ell} + \frac{1}{b^2\ell}\right)$.

(Uniform) SAC^1**.** The complexity class (uniform) SAC^1 is a restricted subclass of (uniform) AC^1. Throughout this paper, SAC^1 circuits will refer to (logspace-)uniform SAC^1 circuits. We define SAC^1 circuits and their corresponding circuit evaluation problem as follows:

Definition 2.14 (Uniform SAC^1 Circuit Evaluation, adapted from [BCD+89])**.** *A Boolean circuit* $C\colon \{0,1\}^n \to \{0,1\}$ *is defined as an* SAC^1 *circuit if it has depth* $O(\log n)$*, includes unbounded fan-in OR* ($\vee$) *gates, bounded fan-in AND* ($\wedge$) *gates (e.g., with fan-in* 2*), and has negation* ($\neg$) *gate restricted to the input level. The problem is to decide whether a given (logspace-)uniform* SAC^1 *circuit* C*, whose description can be computed by a deterministic logspace Turing machine, evaluates to* 1*.*

Venkateswaran [Ven91] established that SAC^1 is equivalent to LOGCFL, the complexity class consists of languages that are logspace-reducible to context-free languages [Sud78]. To compare with other classes of logspace-uniform bounded-depth Boolean circuits, it is known that:

$$\mathsf{NL} \subseteq \mathsf{SAC}^1 = \mathsf{LOGCFL} \subseteq \mathsf{AC}^1 \subseteq \mathsf{NC}^2$$

Here, NL is a logspace version of NP with polynomial-length witness, AC^1 captures the power of $O(\log n)$-depth Boolean circuits using *unbounded* fan-in gates, and NC^2 characterizes the power of $O(\log^2 n)$-depth Boolean circuits with *bounded* fan-in gates.

Similar to NL, LOGCFL (equivalently, SAC^1) is closed under complementation [BCD+89]. However, whether SAC^1 is contained in BPL or BQL remains an open problem.

3 Space-bounded (unitary) quantum interactive proofs

In this section, we introduce space-bounded quantum interactive proofs (QIPL), where the verifier's actions are implemented using space-bounded *almost-unitary* quantum circuits (see Definition 2.8); along with the variant $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, in which the verifier's actions are restricted to *unitary* circuits. Both QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ are variants of single-prover quantum interactive proofs (QIP) [Wat03b, KW00] that have a space constraint. We establish three theorems concerning the classes QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, focusing on space-bounded (unitary) quantum interactive proofs with a *polynomial* number of messages.

The first theorem shows that QIPL provides a new exact characterization of NP, as stated in Theorem 3.1. This result can be seen as a quantum analog of classical works [Lip90, CL95].

Theorem 3.1 (The equivalence of QIPL and NP)**.** *The following holds:*

(1) *For any logspace-computable function $m(n)$ such that $1 \leq m(n) \leq \mathrm{poly}(n)$,*

$$\cup_{c(n)-s(n)\geq 1/\mathrm{poly}(n)} \mathsf{QIPL}_m[c,s] \subseteq \mathsf{NP}.$$

(2) $\mathsf{NP} \subseteq \mathsf{QIPL}_m$*, where $m(n)$ is some polynomial in n.*

The second theorem addresses two fundamental but crucial properties of QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$. Specifically, closure under perfect completeness (Lemma 3.12) and error reduction through sequential repetition (Lemma 3.13):

Theorem 3.2 (Basic properties for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\mathrm{poly}(n)$, and $1 \leq m(n) \leq \mathrm{poly}(n)$. Then the following properties hold:*

(1) **Closure under perfect completeness**.

$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{QIPL}_{m+2}[1, 1-(c-s)^2/2] \text{ and } \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c,s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m+2}[1, 1-(c-s)^2/2].$$

(2) **Error reduction**. *For any polynomial $k(n)$,*

$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{QIPL}_{m'}[1, 2^{-k}] \text{ and } \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c,s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m'}[1, 2^{-k}].$$

Here, m' is some polynomial in n.

The third theorem provides a lower bound for $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, which serves as a quantum analog of the space-bounded public-coin classical interactive proof for SAC^1 established in [For89]:

Theorem 3.3. $\mathsf{SAC}^1 \cup \mathsf{BQL} \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m$*, where $m(n)$ is some polynomial in n.*

In the remainder of this section, we provide the definitions of space-bounded (unitary) quantum interactive proofs, specifically the classes QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, in Section 3.1. We then formulate QIPL proof systems as semi-definite programs in Section 3.2, leading to the inclusion $\mathsf{QIPL} \subseteq \mathsf{NP}$ (Theorem 3.8). Next, the proof of the two basic properties in Theorem 3.2 is presented in Section 3.3. Lastly, the lower bounds for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, particularly the inclusions $\mathsf{NP} \subseteq \mathsf{QIPL}$ (Theorem 3.14) and $\mathsf{SAC}^1 \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ (Theorem 3.3), are established in Section 3.4.

3.1 Definitions of space-bounded quantum interactive proof systems

Our definitions of space-bounded quantum interactive proofs follow that of [KW00, Section 2.3] and [Wat02, Section 2.3]. In this framework, a (log)space-bounded quantum interactive proof system consists of two parties: an untrusted prover with unbounded computational power, and a verifier constrained to using only $O(\log n)$ *qubits*, enabling at most polynomial-time quantum computation. The primary distinction between standard single-prover quantum interactive proofs and their space-bounded variants lies in this additional space constraint on the verifier, which prompts a subtle question:

Problem 3.4. *Is it necessary to allow $O(\log n)$ intermediate measurements in the computational basis during* each *verifier's action in space-bounded quantum interactive proof systems?*

Interestingly, the answer to Problem 3.4 does not affect one-message proof systems, specifically (unitary) QMAL [FKL+16, FR21], where the unitary verification circuit acts on $O(\log n)$ qubits and inherently allows $O(\log n)$ intermediate measurements (see also Remark 2.9).

To address Problem 3.4, we introduce two types of space-bounded quantum interactive proof systems, denoted by QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$. In both proof systems, the verifier has *direct access* to the messages exchanged during interactions, which limits each message size to $O(\log n)$. The key distinction between these proof systems lies in their differing responses to Problem 3.4.

In QIPL, the verifier's actions correspond to space-bounded *almost-unitary* quantum circuits (see Definition 2.8), allowing $O(\log n)$ intermediate measurements in each verifier's action. As an additional restriction, for *yes* instances, the distribution of these intermediate measurement outcomes, conditioned on acceptance, is *highly concentrated*. However, in $\mathsf{QIP_UL}$, the verifier's actions are implemented using space-bounded *unitary* quantum circuits. The former model can be considered the weakest one that still encompasses space-bounded (private-coin) classical interactive proof systems, particularly the model described in [CL95].[21]

Formal definitions of QIPL and $\mathsf{QIP_UL}$. Given a promise problem $\mathcal{I} = (\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$, a quantum verifier is *logspace-computable* mapping V, where for each input string $x \in \mathcal{I} \subseteq \{0,1\}^*$, $V(x)$ is interpreted as an encoding of a $k(|x|)$-tuple $(V(x)_1, \cdots, V(x)_k)$ of quantum circuits. These circuits represent the verifier's actions at each round of the proof system, with specific constraints depending on the proof system, QIPL or $\mathsf{QIP_UL}$:

- In a QIPL proof system, each $V(x)_j$ is a space-bounded *almost-unitary* quantum circuit (an isometry, see Definition 2.8) that takes qubits in registers (M, W) as input and outputs qubits in registers $(\mathsf{M}, \mathsf{W}, \mathsf{E}_j)$, where E_j holds $q_{\mathsf{E}_j}(|x|)$ qubits. At the end of the verifier's j-th action $V(x)_j$, the (newly introduced) environment register E_j is measured in the computational basis, with the measurement outcome denoted as u_j. The total number of qubits satisfies $q_{\mathsf{M}}(|x|) + q_{\mathsf{W}}(|x|) + q_{\mathsf{E}_j}(|x|) \leq O(\log n)$, with both W and E_j private to the verifier.
- In a $\mathsf{QIP_UL}$ proof system, each $V(x)_j$ is a space-bounded *unitary* quantum circuit acting on two registers M and W, which hold $q_{\mathsf{M}}(|x|)$ and $q_{\mathsf{W}}(|x|)$ qubits, respectively. The total number of qubits satisfies $q_{\mathsf{M}}(|x|) + q_{\mathsf{W}}(|x|) \leq O(\log n)$, with W private to the verifier.

Furthermore, the logspace-computability of $V(x)$ requires *a strong notion of uniformity*: there must exist a logspace deterministic Turing machine $\mathcal{M}$ that, for each input x, outputs the classical description of $(V(x)_1, \cdots, V(x)_k)$.[22] Lastly, the verifier V is called $m(|x|)$-message if $k(|x|) = \lfloor m(|x|)/2 + 1 \rfloor$ for all integer $|x|$, depending on whether m is even or odd.

Similar to standard quantum interactive proofs, the prover and the verifier in the same space-bounded quantum interactive proof system must be *compatible*. This means that they must agree on the maximum length $q_{\mathsf{M}}(|x|)$ of each message exchanged in the proof system and the total number $m(|x|)$ of these messages. Hence, a quantum prover P is a function that maps each input $x \in \mathcal{I}$ to an $l(|x|)$-tuple $(P(x)_1, \cdots, P(x)_l)$ of quantum circuits, where $l(|x|) = \lfloor (m(|x|) + 1)/2 \rfloor$. Each circuit $P(x)_j$ acts on two registers Q and M with $q_{\mathsf{Q}}(|x|)$ and $q_{\mathsf{M}}(|x|)$ qubits, respectively, satisfying that Q is private to the prover. Since there are no restrictions on the prover P, each $P(x)_j$ can be viewed as an arbitrary unitary transformation in general.

Given an input $x \in \mathcal{I}$, and a prover P and a verifier V that exchange $m(|x|)$ messages, we define an $m(|x|)$-turn space-bounded quantum interactive proof system $(P \rightleftharpoons V)(x)$, namely a QIPL proof system, as a quantum circuit acting on the registers Q, M, W, and additional environment registers $\{\mathsf{E}_j\}$ as follows:

- If $m(|x|) = 2l(|x|)$ is even, circuits $V(x)_1, P(x)_1, \cdots, V(x)_l, P(x)_l, V(x)_{l+1}$ are applied in sequence to the registers M, W, and E_j, or to the registers Q and M accordingly. It is important to note that the register E_j is inaccessible after the j-th round.
- If $m(|x|) = 2l(|x|) + 1$ is odd, the situation is similar, except that the prover starts the protocol, so the circuits $P(x)_1, V(x)_1, \cdots, P(x)_{l+1}, V(x)_{l+1}$ are applied in sequence.

[21] For any proof system that ensures soundness against classical messages, we can construct a corresponding proof system that guarantees soundness against quantum messages by measuring the message in the computational basis at the beginning of each verifier's action.

[22] This uniformity requirement is slightly stronger and less general than merely requiring all quantum circuits $V(x)_1, \cdots, V(x)_k$ to be logspace-bounded (referred to as *a weaker notion of uniformity*), as the classical descriptions of these quantum circuits may not be generated by a single logspace deterministic Turing machine (although a polynomial-time deterministic Turing machine would suffice).

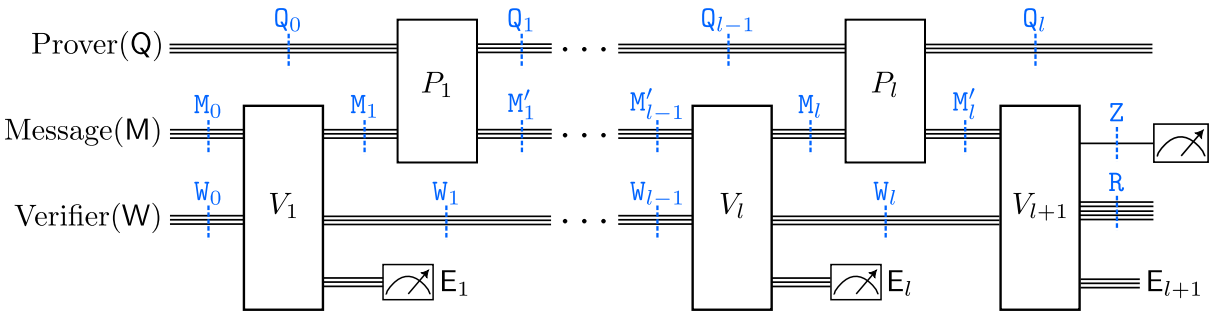

Figure 3.1: A $2l$-turn space-bounded quantum interactive proof system (with snapshots).

Analogously, for an $m(|x|)$-turn space-bounded *unitary* quantum interactive proof system, namely a $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof system, the definition remains the same except that environment registers $\{\mathsf{E}_j\}$ are no longer involved. Without the loss of generality, we assume that the prover always sends the last message. See also Figure 3.1 for an illumination of the case when $m(|x|)$ is even. For convenience, we sometimes omit the dependence on x and $|x|$ when describing P and V, e.g., using P_j and V_j to denote $P(x)_j$ and $V(x)_j$, respectively, and m to denote $m(|x|)$.

Assuming the mapping $V(x) = (V(x)_1, \cdots, V(x)_k)$ in a QIPL proof system is a collection of almost-unitary quantum circuits,[23] the state of the qubits in the circuit $P \rightleftharpoons V$ is an (unnormalized) *pure state* on the registers $(\mathsf{Q}, \mathsf{M}, \mathsf{V}, \mathsf{E}_j)$ after the verifier's j-th action.[24] A similar observation holds for $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof systems. Thus, for a given input x, the probability that $P \rightleftharpoons V$ accepts x is defined as the probability that measuring the designated output qubit – typically the first qubit of (M, W) – of $(P \rightleftharpoons V)(x)|\bar{0}\rangle_{\mathsf{Q}}|\bar{0}\rangle_{\mathsf{M}}|\bar{0}\rangle_{\mathsf{W}}$ in the computational basis yields the outcome 1.

Let $\omega(V)$ denote the maximum acceptance probability of the verifier V in the proof system $P \rightleftharpoons V$. For QIPL proof systems, we impose an additional restriction on the distribution of intermediate measurement outcomes $u := (u_1, \cdots, u_l)$, conditioned on acceptance, for *yes* instances. We define $\omega(V)|^u$ as the contribution of the measurement outcome u to $\omega(V)$, where all post-measurement states remain *unnormalized*. A direct calculation then implies that

$$\omega(V) = \sum\nolimits_{u \in \{0,1\}^{q_{\mathsf{E}_1} + \cdots + q_{\mathsf{E}_l}}} \omega(V)|^u. \tag{3.1}$$

We are now ready to formally define space-bounded quantum interactive proof systems:

Definition 3.5 (Space-bounded quantum interactive proofs, QIPL)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions of the input length $n := |x|$ such that $0 \le s(n) < c(n) \le 1$ and $1 \le m(n) \le \mathrm{poly}(n)$. A promise problem $\mathcal{I} = (\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$ is in $\mathsf{QIPL}_m[c, s]$, if there exists an $m(n)$-turn logspace-computable* almost-unitary *quantum verifier V such that:*

- **Completeness**. *For any $x \in \mathcal{I}_{\mathrm{yes}}$, there exists an $m(n)$-message prover P such that there exists an intermediate measurement outcome $u^* = (u_1^*, \cdots, u_l^*)$ with*

$$\omega(V)|^{u^*} \ge c(n).$$

- **Soundness**. *For any $x \in \mathcal{I}_{\mathrm{no}}$ and any $m(n)$-message prover P,*

$$\omega(V) \le s(n).$$

Furthermore, we define $\mathsf{QIPL}_m := \mathsf{QIPL}_m[2/3, 1/3]$ and $\mathsf{QIPL} := \cup_{m \le \mathrm{poly}(n)} \mathsf{QIPL}_m$.

[23] This assumption about the verifier's actions is crucial for adapting several techniques from standard quantum interactive proofs. For more on general verifiers in the standard scenario, see [VW16, Section 4.1.4].

[24] Specifically, a pure state occupies the registers $(\mathsf{Q}, \mathsf{M}, \mathsf{V}, \mathsf{E}_1)$ before the measurements at the end of the verifier's first action. To align with the proof of Lemma 3.11 (the upper bound for QIPL), which considers only a specific measurement outcome, this post-measurement pure state will be unnormalized.

Interestingly, the completeness condition in Definition 3.5 can be relaxed to $\sum_{u\in\mathcal{J}}\omega(V)|^u \geq c(n)$, where $\mathcal{J}$ is an index set of size *polynomial* in n. The resulting class, denoted by $\mathsf{QIPL}^\star$, remains equivalent to QIPL. Analogously, we can define space-bounded unitary quantum interactive proof systems, without imposing the additional restriction on *yes* instances:

Definition 3.6 (Space-bounded unitary quantum interactive proofs, $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions of the input length $n := |x|$ such that $0 \leq s(n) < c(n) \leq 1$ and $1 \leq m(n) \leq \mathrm{poly}(n)$. A promise problem $\mathcal{I} = (\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$ is in $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c,s]$, if there exists an $m(n)$-turn logspace-computable* unitary *quantum verifier V such that*:

- **Completeness**. *For any $x \in \mathcal{I}_{\mathrm{yes}}$, there exists an $m(n)$-message prover P such that*
$$\omega(V) \geq c(n).$$
- **Soundness**. *For any $x \in \mathcal{I}_{\mathrm{no}}$ and any $m(n)$-message prover P,*
$$\omega(V) \leq s(n).$$

Furthermore, we define $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m := \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[2/3, 1/3]$ and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L} := \cup_{m\leq\mathrm{poly}(n)}\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m$.

Remark 3.7 (A reversible generalization of QIPL). We introduce the class $\mathsf{QIPL}^\diamond$ as a *reversible* generalization of QIPL, primarily for convenience. A $\mathsf{QIPL}^\diamond$ proof system is defined similarly to a QIPL proof system, but with three crucial differences:

(1) All of the verifier's actions are *isometric* quantum circuits, without restrictions. In particular, any unitary elementary gate can act on the ancillary qubits $|\bar{0}\rangle$, which are introduced by $O(\log n)$ ancillary gates, and qubits in the message register M.

(2) The environment register E_k, introduced during the verifier's k-th action, remains private to the verifier and is *accessible only during that turn.* Importantly, the qubits in E_k are not measured at the end of the turn. Consequently, the qubits in E_k remain unchanged after that turn, although the entanglement shared among $\mathsf{E}_1, \cdots, \mathsf{E}_k$, and M may change.

(3) The completeness condition is simply $\omega(V) \geq c(n)$, without any addition restrictions.

3.2 An upper bound for QIPL via SDP formulations

We begin with the upper bound for the class QIPL:[25]

Theorem 3.8 (QIPL is in NP)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\mathrm{poly}(n)$, and $1 \leq m(n) \leq \mathrm{poly}(n)$, it holds that*
$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{NP}.$$

Before presenting the proof, we introduce the term *snapshot registers* to refer to the registers Q, M, and W after each turn in a QIPL proof system. We also refer to the quantum state within these snapshot registers as *snapshot states*. For example, the snapshot registers corresponding to W at distinct time points are $\mathtt{W}_0, \cdots, \mathtt{W}_l$, as illustrated in Figure 3.1. More precisely, for an l-round (i.e., $2l$-turn) QIPL proof system, we define the following:

(1) $\mathtt{Q}_0$, $\mathtt{M}_0$, and $\mathtt{W}_0$, which contain the all-zero state, are the snapshot registers of registers Q, M, and W, respectively, before the protocol begins;

(2) $\mathtt{M}_j$ and $\mathtt{W}_j$ $(1 \leq j \leq l)$ are the snapshot registers of the registers M and W, respectively, after the verifier sends the message in the j-th round;

[25] It is noteworthy that this upper bound also applies to two variants of QIPL: (1) when the verifier's mapping satisfies a weaker notion of uniformity (see Footnote 22 for details), and (2) to $\mathsf{QIPL}^\star$, as defined in Section 3.1, where the completeness condition is slightly relaxed.

(3) $\mathtt{M}'_j$ and $\mathtt{W}_j$ $(1 \le j \le l)$ are the snapshot registers of the registers M and W, respectively, after the verifier receives the message from the prover in the j-th round;

(4) $\mathtt{Q}_j$ $(1 \le j \le l)$ is the snapshot registers of the register Q immediately after applying the prover's j-th action P_j, i.e., after the prover sends the message in the j-th round;

(5) $(\mathtt{Z}, \mathtt{R})$ represents the snapshot registers of registers M and W, respectively, just before the verifier performs the final measurement, where $\mathtt{Z}$ corresponds to the designated output qubit of the verifier and $\mathtt{R}$ contains the remained qubits.

In the remainder of this subsection, we first present semi-definite programming (SDP) formulations for QIPL proof systems in Section 3.2.1, and then establish an upper bound for QIPL (Theorem 3.8) in Section 3.2.2.

3.2.1 Semi-definite program formulations for QIPL proof systems

To establish upper bounds for space-bounded (unitary) quantum interactive proofs, a commonplace approach involves solving the optimization problem of approximating the maximum acceptance probability of a QIPL and $\mathsf{QIP_UL}$ proof system over all prover strategies. For clarity, we first present an SDP for characterizing QIPL proof systems (Lemma 3.9), directly extended from [VW16, Section 4.3] and [Wat16, Section 4]. Next, we introduce another SDP characterization (Lemma 3.11) that fully incorporates all restrictions of QIPL proof systems.

First SDP formulation for QIPL proof systems. For an m-turn QIPL proof system with even m,[26] we formulate this optimization problem as a semi-definite program (SDP) from the verifier's perspective, following the approach described in [VW16, Section 4.3]:

Lemma 3.9 (First SDP formulation for QIPL proof systems)**.** *For any $l(n)$-round space-bounded quantum interactive proof system $P \rightleftharpoons V$ with completeness $c(n)$, soundness $s(n)$, which corresponds to a promise problem $\mathcal{I} = (\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$ in $\mathsf{QIPL}_{2l}[c, s]$, there is an SDP program to compute the maximum acceptance probability $\omega(V)$ of the proof system $P \rightleftharpoons V$:*

$$\begin{aligned}
\text{maximize} \quad & \omega(V) = \mathrm{Tr}\left(\widetilde{V}_{l+1}^{\dagger} |1\rangle\langle 1|_{\text{out}} \widetilde{V}_{l+1} \rho_{\mathtt{M}'_l \mathtt{W}_l \mathtt{E}_1 \cdots \mathtt{E}_l}\right) \\
\text{subject to} \quad & \mathrm{Tr}_{\mathtt{M}'_1}\left(\rho_{\mathtt{M}'_1 \mathtt{W}_1 \mathtt{E}_1}\right) = \mathrm{Tr}_{\mathtt{M}_1}\left(V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathtt{M}_0 \mathtt{W}_0} V_1^{\dagger}\right), \\
& \mathrm{Tr}_{\mathtt{M}'_j}\left(\rho_{\mathtt{M}'_j \mathtt{W}_j \mathtt{E}_1 \cdots \mathtt{E}_j}\right) = \mathrm{Tr}_{\mathtt{M}_j}\left(\widetilde{V}_j \rho_{\mathtt{M}'_{j-1} \mathtt{W}_{j-1} \mathtt{E}_1 \cdots \mathtt{E}_{j-1}} \widetilde{V}_j^{\dagger}\right), \quad j \in [l] \setminus [1], \\
& \mathrm{Tr}\left(\rho_{\mathtt{M}'_j \mathtt{W}_j \mathtt{E}_1 \cdots \mathtt{E}_j}\right) = 1, \quad j \in [l], \\
& \rho_{\mathtt{M}'_j \mathtt{W}_j \mathtt{E}_1 \cdots \mathtt{E}_j} \succeq 0, \quad j \in [l]
\end{aligned} \tag{3.2}$$

Here, the verifier's actions $V_1, \cdots, V_{l+1}$ are considered space-bounded isometric *quantum circuits, with the notation $\widetilde{V}_j := V_j \otimes I_{\mathtt{E}_1 \cdots \mathtt{E}_{j-1}}$ for each $j \in [l+1] \setminus [1]$. The variables in this SDP are $\rho_{\mathtt{M}'_1 \mathtt{W}_1 \mathtt{E}_1}, \cdots, \rho_{\mathtt{M}'_l \mathtt{W}_l \mathtt{E}_1 \cdots \mathtt{E}_l}$, collectively holding $O(l^2(n) \cdot \log n)$ qubits.*

Remark 3.10 (The applicability of the first SDP formulation for QIPL). The SDP program in Equation (3.2) essentially characterizes $\mathsf{QIPL}^{\diamond}$ proof systems (Remark 3.7), a reversible generalization of QIPL where the verifier's actions are isometric quantum circuits and the completeness condition is simply $\omega(V) \ge c(n)$. Additionally, by disregarding all environment registers $\{E_j\}$ in Equation (3.2), we immediately obtain an SDP formulation for $\mathsf{QIP_UL}$ proof systems, where the variables are $\rho_{\mathtt{M}'_1 \mathtt{W}_1}, \cdots, \rho_{\mathtt{M}'_l \mathtt{W}_l}$, collectively holding $O(l(n) \cdot \log n)$ qubits.

Our SDP program in Equation (3.2) consists of two types of constraints, both of which can be described by simple equations: (1) the verifier remains honest, and (2) the prover's actions do not

[26] Adapting the proof to the case of odd m is straightforward, so we omit the details.

interfere with the verifier's private qubits. Importantly, every feasible solution to Equation (3.2) corresponds to a valid strategy for the prover. We now proceed with the detailed proof:

Proof of Lemma 3.9. For any $m(n)$-turn proof system $P \rightleftharpoons V$, with completeness c, soundness s, and m being even, which corresponds to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}_m[c,s]$, we consider the verifier's maximum acceptance probability $\omega(V)$ as the objective function to be maximized. In our SDP formulation for the QIPL proof system $P \rightleftharpoons V$, we focus on the verifier's actions (e.g., V_j), specifically isometric quantum circuits that do not measure the new environment register (e.g., E_j) at the end. As defined in Section 3.1, the verifier V is described by an $(l+1)$-tuple $(V_1, \cdots, V_{l+1})$ of space-bounded unitary quantum circuits $\{V_j\}_{j\in[l+1]}$, where $l = m/2$.

To represent the variables in our SDP program, which are the states in the snapshot registers corresponding to the message register M, the verifier's private register W, and the environment registers $\mathsf{E}_1 \cdots, \mathsf{E}_j$ after the verifier's j-th action in $P \rightleftharpoons V$, we use the notations defined in Figure 3.1. Specifically, let $\rho_{\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}$ denote the state in the snapshot registers $(\mathtt{M}_j, \mathtt{W}_j, \mathtt{E}_1, \cdots, \mathtt{E}_j)$. Similarly, we can define snapshot states $\rho_{\mathtt{M}'_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}$ for $j \in [l]$ and $\rho_{\mathtt{ZR}\mathtt{E}_1\cdots\mathtt{E}_{l+1}}$ accordingly.

Assuming that $P \rightleftharpoons V$ begins with the verifier, it follows that the objective function

$$\begin{aligned}
\omega(V) &= \| |1\rangle\langle 1|_{\text{out}} V_{l+1} P_l V_l \cdots P_1 V_1 |\bar{0}\rangle_{\mathtt{Q}_0,\mathtt{M}_0,\mathtt{W}_0} \|_2^2 \\
&= \mathrm{Tr}\Big(\mathrm{Tr}_{\mathsf{Q}} \Big((V_{l+1} P_l V_l \cdots P_1 V_1)^\dagger |1\rangle\langle 1|_{\text{out}} (V_{l+1} P_l V_l \cdots P_1 V_1) |\bar{0}\rangle\langle\bar{0}|_{\mathtt{Q}_0,\mathtt{M}_0,\mathtt{W}_0} \Big) \Big) \\
&= \mathrm{Tr}\Big(\big(V_{l+1}^\dagger \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_l}\big) |1\rangle\langle 1|_{\text{out}} \big(V_{l+1}^\dagger \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_l}\big) \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l} \Big)
\end{aligned} \tag{3.3}$$

Noting that the verifier V remains honest and the verifier's j-th action does not act on the environment registers $\mathsf{E}_1, \cdots, \mathsf{E}_{j-1}$, we obtain the first type of constraints:

$$\begin{aligned}
\rho_{\mathtt{M}_1\mathtt{W}_1\mathtt{E}_1} &= V_1 \rho_{\mathtt{M}_0\mathtt{W}_0} V_1^\dagger = V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathtt{M}_0\mathtt{W}_0} V_1^\dagger; \\
\forall j \in \{2, \cdots, l\},\ \rho_{\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j} &= \big(V_j \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_{j-1}}\big) \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}\mathtt{E}_1\cdots\mathtt{E}_{j-1}} \big(V_j^\dagger \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_{j-1}}\big); \\
\rho_{\mathtt{ZR}\mathtt{E}_1\cdots\mathtt{E}_{l+1}} &= \big(V_{l+1} \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_l}\big) \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l} \big(V_{l+1}^\dagger \otimes I_{\mathsf{E}_1\cdots\mathsf{E}_l}\big).
\end{aligned} \tag{3.4}$$

Since the prover's actions are described by unitary quantum circuits and the verifier's actions by isometric quantum circuits, all intermediate states in $(\mathsf{Q}, \mathsf{M}, \mathsf{W}, \mathsf{E}_1, \cdots, \mathsf{E}_j)$ after the verifier's j-th action in $P \rightleftharpoons V$ are pure states. These states, denoted by $|\psi\rangle_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}$ and $|\phi\rangle_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}$ for $j \in [l]$, satisfy the relation $|\phi\rangle_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j} = P_j |\psi\rangle_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}$. By the unitary freedom in purifications (Lemma 2.7), we have:

$$\forall j \in [l],\ \mathrm{Tr}_{\mathtt{Q}_{j-1}\mathtt{M}_j}\big(|\psi\rangle\langle\psi|_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}\big) = \rho_{\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j} = \mathrm{Tr}_{\mathtt{Q}_j\mathtt{M}'_j}\Big(|\phi\rangle\langle\phi|_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}\Big).$$

Here, the underlying unitary transformation corresponds to the prover's action P_j in the j-th round. As a consequence, the prover's actions do not interfere with the verifier's private register W or the environment registers $\mathsf{E}_1, \cdots, \mathsf{E}_j$ (after the verifier's j-th action) during the execution of $P \rightleftharpoons V$. This property leads to the second type of constraints:

$$\forall j \in [l],\ \mathrm{Tr}_{\mathtt{M}_j}\big(\rho_{\mathtt{M}_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}\big) = \mathrm{Tr}_{\mathtt{M}'_j}\Big(\rho_{\mathtt{M}'_j\mathtt{W}_j\mathtt{E}_1\cdots\mathtt{E}_j}\Big). \tag{3.5}$$

Putting Equation (3.3), Equation (3.4), and Equation (3.5) all together, we conclude our SDP formulation for the given QIPL proof system $P \rightleftharpoons V$, as detailed in Equation (3.2). □

Second SDP formulation for QIPL proof systems. The main challenge in fully utilizing all restrictions of QIPL proof systems in an SDP program is effectively using the measurement outcomes from the newly introduced environment register E_j after the verifier's j-th action. Specifically, when applying the principle of deferred measurements to the verifier's j-th action (almost-unitary quantum circuit), an environment register E_j is introduced. Thus, the variables

in Equation (3.2) correspond to $\rho_{\mathsf{M}'_1\mathsf{W}_1\mathsf{E}_1}$, $\rho_{\mathsf{M}'_1\mathsf{W}_1\mathsf{E}_1\mathsf{E}_2}$, $\cdots$, $\rho_{\mathsf{M}'_l\mathsf{W}_l\mathsf{E}_1\cdots\mathsf{E}_l}$. In general, when the verifier's actions are isometric quantum circuits, the environment registers $\mathsf{E}_1, \cdots, \mathsf{E}_l$ may be *entangled*.[27]

However, an almost-unitary quantum circuit (the verifier's j-th action) corresponds to a specific type of isometric quantum circuit, followed by measuring the environment register E_j in the computational basis at the end of the circuit. Therefore, the environment registers $\mathsf{E}_1, \cdots, \mathsf{E}_l$ remain *independent* in this restricted setting, leading to the following SDP program:

Lemma 3.11 (Second SDP formulation for QIPL proof systems)**.** *For any $l(n)$-round space-bounded quantum interactive proof system $P \rightleftharpoons V$ with completeness $c(n)$ and soundness $s(n)$, corresponding to a promise problem $\mathcal{I} = (\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$ in $\mathsf{QIPL}_{2l}[c, s]$, there is a family of SDP programs for computing the contribution $\omega(V)|^u$ of the measurement outcome $u = (u_1, \cdots, u_l)$ to the maximum acceptance probability $\omega(V)$ of the proof system $P \rightleftharpoons V$. Here, each u_j for $j \in [l]$ is a measurement outcome from the environment register E_j following the verifier's j-th action. Specifically, an SDP program for computing $\omega(V)|^u$ can be formulated as follows*:

$$
\begin{aligned}
\text{maximize} \quad & \omega(V)|^u = \mathrm{Tr}\Big(V_{l+1}^\dagger |1\rangle\langle 1|_{\text{out}} V_{l+1} \rho_{\mathsf{M}'_l\mathsf{W}_l}\Big) \\
\text{subject to} \quad & \mathrm{Tr}_{\mathsf{M}'_1}\big(\rho_{\mathsf{M}'_1\mathsf{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathsf{E}_1}\big) = \mathrm{Tr}_{\mathsf{M}_1}\Big(\big(I_{\mathsf{M}_1\mathsf{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathsf{E}_1}\big) V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathsf{M}_0\mathsf{W}_0} V_1^\dagger\Big), \\
& \mathrm{Tr}_{\mathsf{M}'_j}\big(\rho_{\mathsf{M}'_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\big) = \mathrm{Tr}_{\mathsf{M}_j}\Big(\big(I_{\mathsf{M}_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\big) V_j \rho_{\mathsf{M}'_{j-1}\mathsf{W}_{j-1}} V_j^\dagger\Big), \quad j \in [l]\backslash[1], \\
& \mathrm{Tr}\big(\rho_{\mathsf{M}'_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\big) \le 1, \qquad\qquad j \in [l], \\
& \rho_{\mathsf{M}'_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j} \succeq 0, \qquad\qquad j \in [l]
\end{aligned}
\tag{3.6}
$$

Here, the verifier's actions $V_1, \cdots, V_{l+1}$ correspond to space-bounded almost-unitary *quantum circuits, which we interpret as a special class of* isometric *quantum circuits. The variables in this SDP are unnormalized states $\rho_{\mathsf{M}'_1\mathsf{W}_1}, \cdots, \rho_{\mathsf{M}'_l\mathsf{W}_l}$, collectively holding $O(l(n) \cdot \log n)$ qubits.*

Proof. Our proof strategy follows a similar approach to that of Lemma 3.9, so we will only highlight the key differences. As defined in Section 3.1, the verifier V is described by an $(l+1)$-tuple $(V_1, \cdots, V_{l+1})$ of space-bounded almost-unitary quantum circuits $\{V_j\}_{j\in[l+1]}$, which are a special class of isometric quantum circuits, where $l = m/2$.

To represent the variables in this SDP program, which are the *unnormalized* states in the snapshot registers corresponding to the message register M and the verifier's private register W, we use the notations from Figure 3.1. Assume that it suffices to consider the measurement outcome u_j obtained from measuring E_j in the computational basis at the end of the verifier's j-th action. Consequently, the state $|u_j\rangle\langle u_j|$ in the environment register E_j is treated as part of the SDP constraints, not as the variables. In particular, we slightly abuse the notation by letting $\rho_{\mathsf{M}_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}$ denote the unnormalized snapshot state in the registers $(\mathsf{M}, \mathsf{W}, \mathsf{E}_j)$ after the j-th verifier's action. Similarly, we define unnormalized snapshot states $\rho_{\mathsf{M}'_j\mathsf{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}$ for $1 \le j \le l$ and $\rho_{\mathsf{ZRE}_{l+1}}$ for the corresponding registers.

Objective function and dependence on measurement outcomes. Assuming that $P \rightleftharpoons V$ begins with the verifier, it follows that the objective function is defined as

$$
\omega(V) = \| |1\rangle\langle 1|_{\text{out}} V_{l+1} P_l V_l \cdots P_1 V_1 |\bar{0}\rangle_{\mathsf{Q}_0\mathsf{M}_0\mathsf{W}_0} \|_2^2 = \mathrm{Tr}\Big(V_{l+1}^\dagger |1\rangle\langle 1|_{\text{out}} V_{l+1} \rho_{\mathsf{M}'_l\mathsf{W}_l}\Big). \tag{3.7}
$$

We now explain why it suffices to focus on a specific measurement outcome u_j obtained from measuring E_j for $j \in [l]$, along with the unnormalized resulting state in the registers $(\mathsf{Q}, \mathsf{M}, \mathsf{W})$ after these intermediate measurements. For *yes* instances, the completeness condition

[27] For instance, the prover may send a highly-entangled n-qubit state ϱ in $n/\log n$ batches, each containing $\log n$ qubits. In this case, the verifier keeps only $O(\log n)$ qubits, which are not necessarily adjacent, and swaps the remaining qubits with fresh qubits in the environment registers.

in Definition 3.5 guarantees the existence of a measurement outcome $u^* = (u_1^*, \cdots, u_l^*)$ such that $\omega(V)|^{u^*} \geq c(n)$. However, for *no* instances, the inequality $\omega(V) = \sum_u \omega(V)|^u \leq s(n)$ implies that $\omega(V)|^u \leq s(n)$ for any measurement outcome $u \in \{0,1\}^{q_{\mathsf{E}_1}+\cdots+q_{\mathsf{E}_l}}$.

Constraints. Note that the verifier V remains honest, with the initial state given by $\rho_{\mathtt{M}_0\mathtt{W}_0} = |\bar{0}\rangle\langle\bar{0}|_{\mathtt{M}_0\mathtt{W}_0}$. The first type of constraints arises from applying the verifier's action V_j:

$$\begin{aligned} \rho_{\mathtt{M}_1\mathtt{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathsf{E}_1} &= \left(I_{\mathtt{M}_1\mathtt{W}_1} \otimes |u_1\rangle\langle u_1|_{\mathsf{E}_1}\right) V_1 |\bar{0}\rangle\langle\bar{0}|_{\mathtt{M}_0\mathtt{W}_0} V_1^\dagger; \\ \forall j \in \{2, \cdots, l\},\ \rho_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j} &= \left(I_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\right) V_j \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}} V_j^\dagger; \\ \rho_{\mathtt{Z}\mathtt{R}\mathsf{E}_{l+1}} &= V_{l+1} \rho_{\mathtt{M}'_l\mathtt{W}_l} V_{l+1}^\dagger. \end{aligned} \tag{3.8}$$

Since the prover's actions are described by unitary quantum circuits, and the verifier's j-th action is an isometric quantum circuit, followed by measuring the (newly introduced) environment register E_j at the end of this turn, all intermediate states in $(\mathsf{Q}, \mathsf{M}, \mathsf{W}, \mathsf{E}_j)$ after the verifier's j-th action in $P \rightleftharpoons V$ are pure states. These unnormalized states, denoted by $|\psi\rangle_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle_{\mathsf{E}_j}$ and $|\phi\rangle_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j} \otimes |u_j\rangle_{\mathsf{E}_j}$ for $j \in [l]$, satisfy the relation $|\phi\rangle_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j} \otimes |u_j\rangle_{\mathsf{E}_j} = P_j |\psi\rangle_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle_{\mathsf{E}_j}$. By the unitary freedom in purifications (Lemma 2.7), it follows that:

$$\forall j \in [l],\ \mathrm{Tr}_{\mathtt{Q}_{j-1}\mathtt{M}_j}\left(|\psi\rangle\langle\psi|_{\mathtt{Q}_{j-1}\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\right) = \rho_{\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j} = \mathrm{Tr}_{\mathtt{Q}_j\mathtt{M}'_j}\left(|\phi\rangle\langle\phi|_{\mathtt{Q}_j\mathtt{M}'_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\right).$$

Here, the underlying unitary transformation corresponds to the prover's action P_j in the j-th round. Consequently, the prover's actions do not interfere with the verifier's private register W or the environment register E_j (introduced by the verifier's j-th action) during the execution of $P \rightleftharpoons V$. This property gives rise to the second type of constraints:

$$\forall j \in [l],\ \mathrm{Tr}_{\mathtt{M}_j}\left(\rho_{\mathtt{M}_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\right) = \mathrm{Tr}_{\mathtt{M}'_j}\left(\rho_{\mathtt{M}'_j\mathtt{W}_j} \otimes |u_j\rangle\langle u_j|_{\mathsf{E}_j}\right). \tag{3.9}$$

Replacing the constraints in Equation (3.2) by Equation (3.8) and Equation (3.9), we conclude the second SDP formulation for the given QIPL protocol, as specified in Equation (3.6). □

3.2.2 QIPL is in NP

To establish that $\mathsf{QIPL} \subseteq \mathsf{NP}$ (Theorem 3.8), we choose the classical witness for the NP containment as consisting of the variables $\rho_{\mathtt{M}'_1\mathtt{W}_1}, \cdots, \rho_{\mathtt{M}'_l\mathtt{W}_l}$ from the SDP program specified in Lemma 3.11, along with the measurement outcomes $u_1, \cdots, u_l$, which determine the SDP program. Then, the verification procedure is simply performing the basic matrix operations on polynomial-dimension matrices. We now proceed with the proof.

Proof of Theorem 3.8. Without loss of generality, we assume that the number of turns $m(n)$ is even, and particularly $m(n) = 2l(n)$. For any l-round proof system $P \rightleftharpoons V$ with completeness c and soundness s, corresponding to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}_{2l}[c, s]$, we can leverage Lemma 3.11 to obtain a family of SDP program, which depends on a measurement outcome $u = (u_1, \cdots, u_l)$ obtained by the verifier, as detailed in Equation (3.6). Specifically, given the measurement outcome u, this SDP program maximizes the contribution of u to the verifier's maximum acceptance probability $\omega(V)|^u$ over all choices of unnormalized states $\rho_{\mathtt{M}'_1\mathtt{W}_1}, \cdots, \rho_{\mathtt{M}'_l\mathtt{W}_l}$. Furthermore, for *yes* instances, there exists a measurement outcome u^* such that $\omega(V)|^{u^*} \geq c(n)$; whereas for *no* instances, $\omega(V)|^u \leq s(n)$ holds for any measurement outcome u.

To establish an NP containment, we thus choose the classical witness, denoted by w, as the classical description of the unnormalized states $\rho_{\mathtt{M}'_1\mathtt{W}_1}, \cdots, \rho_{\mathtt{M}'_l\mathtt{W}_l}$, alongside the binary strings representing the measurement outcomes $u_1, \cdots, u_l$. The size of w remains polynomial in n for the following reasons: (i) the dimension of each unnormalized state $\rho_{\mathtt{M}'_j\mathtt{W}_j}$, for $1 \leq j \leq l$, is at most $2^{O(\log n)}$, which is polynomial in n; (ii) the length of each binary string u_j is bounded by $O(\log n)$; and (iii) the number of rounds $l(n)$ is at most $\mathrm{poly}(n)$.

We now describe the NP verification procedure to complete the proof. Given the classical witness w, the procedure $\widehat{V}$ executes the following steps:

(1) Check whether w represents a feasible solution of the SDP program for computing $\omega(V)|^u$, where u is the measurement outcome given in w, as specified in Equation (3.6).

(2) Compute the value of $\omega(V)|^u$ by performing a polynomial number of matrix multiplications and partial traces of polynomial-dimensional matrices.

It is evident that these steps can be accomplished in deterministic polynomial time. The verification procedure $\widehat{V}$ accepts if the witness w is a feasible solution to Equation (3.6) concerning the given u and the value of $\omega(V)|^u$ is at least $c(n)$; otherwise, $\widehat{V}$ rejects. □

3.3 Basic properties: Error reduction and perfect completeness

The verifier's space constraint for the class QIPL presents several challenges when adapting techniques from standard quantum interactive proofs. For instance, techniques such as error reduction through parallel repetition [KW00, Section 5] and the parallelization approach described in [KW00, Section 4] are applicable to QIPL only under certain conditions. Nevertheless, two basic properties can still be established without additional assumptions:

(1) Achieving perfect completeness in QIPL or $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof systems by adapting the technique in [VW16, Section 4.2.1], as detailed in Lemma 3.12.

(2) Error reduction for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ via sequential repetition, as stated in Lemma 3.13;

We will provide detailed proofs of these properties in the remainder of this subsection.

3.3.1 Achieving perfect completeness for QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$

Our construction and analysis in Lemma 3.12 are inspired by [VW16, Section 4.2.1].

Lemma 3.12 (QIPL and $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ are closed under perfect completeness)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\operatorname{poly}(n)$, and $1 \leq m(n) \leq \operatorname{poly}(n)$. Then, it follows that*

$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{QIPL}_{m+2}\big[1, 1-(c-s)^2/2\big] \text{ and } \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c,s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m+2}\big[1, 1-(c-s)^2/2\big].$$

Proof. Since $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof systems are a special subclass of QIPL proof systems, it suffices to establish the inclusion for the latter. For any m-turn QIPL proof system $P \rightleftharpoons V$ with completeness c and soundness s, which corresponds to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}_m[c,s]$, $P \rightleftharpoons V$ acts on registers Q, M, and $\mathsf{W} = (\mathtt{Z}, \mathtt{R})$, where Z represents the output qubit just before the final measurement. To achieve perfect completeness, we propose a new proof system $P' \rightleftharpoons V'$ based on $P \rightleftharpoons V$, as detailed in Protocol 3.1. In this proof system, we also introduce a new single-qubit register Z', which is initialized to be $|0\rangle$ and is accessible only to V'.

It is noteworthy that the environment register introduced by the verifier's j-th action for $j \in \lfloor (m+1)/2 \rfloor + 1$ in $P \rightleftharpoons V$ has no effect on the new proof system $P' \rightleftharpoons V'$. This is because E_j is measured at the end of the turn in which it is introduced, it collapses to a quantum state that contains only a binary string and becomes inaccessible afterward. Additionally, since the verifier's new turn in Protocol 3.1 does not involve any intermediate measurements, the additional restriction in the completeness condition of QIPL also has no effect on $P' \rightleftharpoons V'$.

Next, we establish the correctness of Protocol 3.1:

- For *yes* instances, the state of the total system after step 2 can be adjusted with the help of an honest prover to

$$|\Phi\rangle = \sqrt{1-\alpha}|00\rangle_{\mathsf{Z}'\mathsf{Z}}|\phi_0\rangle + \sqrt{\alpha}|11\rangle_{\mathsf{Z}'\mathsf{Z}}|\phi_1\rangle,$$

Protocol 3.1: Achieving perfect completeness for QIPL (or $\mathsf{QIP_UL}$).

Parameters: α is a dyadic rational number such that $\frac{3c+s}{4} \leq \alpha \leq c$.

1. The verifier V' executes the original proof system $P \rightleftharpoons V$ (with the prover P'), except for the verifier's final measurement on the register $\mathtt{Z}$;

2. The verifier V' creates a pseudo-copy of the output qubit of V by applying a $\mathrm{CNOT}_{\mathtt{Z}\to\mathtt{Z}'}$ gate, and then sends the register $\mathsf{W} = (\mathtt{Z}, \mathtt{R})$ to the prover;

3. The verifier V' receives a single-qubit state from the prover, places it in the register $\mathtt{Z}$, and measures registers $\mathtt{Z}$ and $\mathtt{Z}'$ in the binary-valued measurement $\{|\gamma\rangle\langle\gamma|, I - |\gamma\rangle\langle\gamma|\}$, where $|\gamma\rangle \coloneqq \sqrt{1-\alpha}|00\rangle + \sqrt{\alpha}|11\rangle$. The verifier V' accepts if the measurement outcome is consistent with $|\gamma\rangle$; otherwise, it rejects.

where $|\phi_0\rangle$ and $|\phi_1\rangle$ are normalized states that may not be orthogonal. This adjustment can be done because $c \geq \alpha$. The prover then applies a unitary U on all the qubits except Z' (which are owned by the prover) to "disentangle" $|\Phi\rangle$:

$$(I_{\mathsf{Z}'} \otimes U)|\Phi\rangle = \left(\sqrt{1-\alpha}|00\rangle_{\mathsf{Z}'\mathsf{Z}} + \sqrt{\alpha}|11\rangle_{\mathsf{Z}'\mathsf{Z}}\right) \otimes |\phi\rangle, \text{ where } |\phi\rangle \text{ is a normalized state.}$$

Consequently in Step 3, the verifier V' holds the state $\left(\sqrt{1-\alpha}|00\rangle + \sqrt{\alpha}|11\rangle\right) = |\gamma\rangle$ in registers $\mathtt{Z}$ and Z', ensuring that V' accepts with certainty.

- For *no* instances, the original proof system $P \rightleftharpoons V$ accepts with probability at most $s = \alpha - \varepsilon$ where $\varepsilon \geq \frac{3c+s}{4} - s = \frac{3}{4}(c-s)$. The reduced density matrix in the register Z' after Step 2 is

$$\rho_{\mathsf{Z}'} = \begin{pmatrix} 1-\alpha+\varepsilon & 0 \\ 0 & \alpha-\varepsilon \end{pmatrix}.$$

Let $\sigma_{\mathtt{Z}\mathsf{Z}'}$ denote a two-qubit quantum state in registers $(\mathtt{Z}, \mathsf{Z}')$ after the verifier V' receives the single-qubit state from the prover P' in Step 3. Regardless of the prover's actions, V' accepts with probability

$$\begin{aligned}
\mathrm{Tr}(|\gamma\rangle\langle\gamma|\sigma_{\mathtt{Z}\mathsf{Z}'}) &= \mathrm{F}(|\gamma\rangle\langle\gamma|, \sigma_{\mathtt{Z}\mathsf{Z}'})^2 \\
&\leq \mathrm{F}(\mathrm{Tr}_{\mathtt{Z}}(|\gamma\rangle\langle\gamma|), \rho_{\mathsf{Z}'})^2 \\
&= \mathrm{F}\left(\begin{pmatrix} 1-\alpha & 0 \\ 0 & \alpha \end{pmatrix}, \begin{pmatrix} 1-\alpha+\varepsilon & 0 \\ 0 & \alpha-\varepsilon \end{pmatrix}\right)^2 \\
&\leq 1 - \varepsilon^2 \\
&\leq 1 - \frac{1}{2}(c-s)^2.
\end{aligned}$$

Here, the second line owes to the data-processing inequality for the fidelity (Lemma 2.5), the fourth line follows from [VW16, Equation (4.25)], and the last line is because of

$$1 - \varepsilon^2 \leq 1 - \left(\frac{3(c-s)}{4}\right)^2 \leq 1 - \frac{1}{2}(c-s)^2. \qquad \square$$

3.3.2 Error reduction for QIPL and $\mathsf{QIP_UL}$

The main challenge in performing sequential repetition of a given QIPL or $\mathsf{QIP_UL}$ proof system $P \rightleftharpoons V$ lies in resetting the qubits in the verifier's private register W to the all-zero state after each execution of $P \rightleftharpoons V$. In non-interactive proof systems with a unitary logspace verifier, the resetting operation can be achieved by running the inverse of the verification circuit, as shown in [MW05, FKL+16]. However, in interactive proof systems with a restricted logspace verifier – whether unitary or almost-unitary – the resetting operation requires assistance from

the prover,[28] who may not be honest. We now proceed to the formal statement:

Lemma 3.13 (Error reduction for QIPL and $\mathsf{QIP_UL}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \le s(n) < c(n) \le 1$, $c(n) - s(n) \ge 1/\operatorname{poly}(n)$, and $1 \le m(n) \le \operatorname{poly}(n)$. For any polynomial $k(n)$, it holds that*

$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{QIPL}_{m'}\left[1, 2^{-k}\right] \text{ and } \mathsf{QIP_UL}_m[c,s] \subseteq \mathsf{QIP_UL}_{m'}\left[1, 2^{-k}\right].$$

Here, the number of turns $m' := O\big(km/\log \frac{1}{1-(c-s)^2/2}\big)$.

Proof. Since $\mathsf{QIP_UL}$ proof systems are a special subclass of QIPL proof systems, it suffices to establish the inclusion for the latter. For any m-turn proof system $P' \rightleftharpoons V'$ with completeness c and soundness s, corresponding to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}_m[c,s]$, applying Lemma 3.12 yields a new $(m+2)$-turn proof system $P \rightleftharpoons V$ with completeness 1 and soundness $1-(c-s)^2/2$. This proof system $P \rightleftharpoons V$, viewed as an isometric quantum circuit, acts on registers Q, M, and $\mathsf{W} = (\mathtt{Z}, \mathtt{R})$, where $\mathtt{Z}$ denotes the output qubit just before the final measurement. Without loss of generality, we can assume that registers M and V are of equal size.

Error reduction for the given proof system $P \rightleftharpoons V$ is achieved using r-fold AND-type sequential repetition of $P \rightleftharpoons V$, as detailed in Protocol 3.2. In this resulting proof system $\widehat{P} \rightleftharpoons \widehat{V}$, we introduce two new $\lceil \log r \rceil$-qubit registers, $\widehat{\mathsf{S}}$ and $\widehat{\mathsf{T}}$, which are initialized to be the all-zero state and are private to $\widehat{V}$. The procedure from Step 3.a to Step 3.c aims to reset the verifier's original private register W with the help of the prover. Moreover, the multiple-controlled adder in Step 3.b can be implemented by $O(q_{\mathsf{W}}^2)$ uses of elementary quantum gates and the adder U_{add}, following from [BBC+95, Lemma 7.5 and Corollary 7.6].

A subtle but important point concerns the effect of the environment register E_j introduced by the verifier's j-th action for $j \in [\lfloor (m+1)/2 \rfloor + 1]$ in $P \rightleftharpoons V$. Since E_j is measured at the end of the turn in which it is introduced, it collapses to a state that contains only a binary string and becomes inaccessible afterward. Therefore, this newly introduced register E_j has no impact on our sequential repetition protocol.

It remains to establish the correctness of $\widehat{P} \rightleftharpoons \widehat{V}$. Let X_i be a random variable indicating whether the i-th execution of $P \rightleftharpoons V$ is accepted, with $\Pr[X_i = 1]$ denoting the verifier V's maximum acceptance probability of the i-th execution. By a direct calculation, it holds that

$$\Pr[\mathrm{Cnt}_{\mathrm{acc}} = r] = \Pr[X_1 = 1 \wedge \cdots \wedge X_r = 1] = \omega(V)^r. \tag{3.10}$$

As a consequence, we conclude the following:

- For *yes* instances, an honest prover always sends $|O\rangle = |0\rangle^{\otimes q_{\mathsf{W}}}$, and runs each of the executions independently, ensuring that the condition $\mathrm{Cnt}_{\mathrm{clean}} = r - 1$ is satisfied. By combining Equation (3.10) with the completeness condition of $P \rightleftharpoons V$, it follows that Protocol 3.2 accepts with certainty. Furthermore, since the resulting proof system $\widehat{P} \rightleftharpoons \widehat{V}$ accepts only if *all* executions of $P \rightleftharpoons V$ accept, there exists $\widehat{u}^*$, which is formed by concatenating r copies of u^*, such that $\omega(\widehat{V})^{|\widehat{u}^*} = 1$. In other words, this specific form of sequential repetition aligns with the completeness condition in the definition of QIPL.

- For *no* instances, it is enough to analyze the acceptance probability when the prover always sends $|0\rangle^{\otimes q_{\mathsf{W}}}$ at Step 3.a, as the verifier will otherwise reject. Although the random variables X_i may not be independent, the probability of $X_i = 1$ is at most $1 - (c-s)^2/2$ for any prover. Therefore, we can be upper bound the acceptance probability using independent binary random variables X_i', where the probability of $X_i' = 1$ is $1-(c-s)^2/2$. Consequently,

[28] In the case of QIPL° proof systems, where the verifier's actions are isometric quantum circuits, implementing the reset operation becomes straightforward. However, for QIPL proof systems, the $O(\log n)$ intermediate measurements conducted during each verifier action are insufficient for re-using the workspace qubits. This limitation arises because the measurement outcome yields merely a binary string encoded in a state.

Protocol 3.2: Error reduction for QIPL (or $\mathsf{QIP_UL}$) via sequential repetition.

Parameters: $r := O\big(k/(c-s)^2\big)$.

For $i \leftarrow 1$ **to** r **:**

1. The verifier $\widehat{V}$ executes the original proof system $P \rightleftharpoons V$ (with the prover $\widehat{P}$), except for the verifier's final measurement on the register Z;

2. The verifier $\widehat{V}$ performs a controlled adder, where the single-qubit control register is $\widehat{\mathsf{Z}}$, and the $\lceil \log r \rceil$-qubit target register is $\widehat{\mathsf{S}}$;

If $i < r$ **:**

3.a The prover $\widehat{P}$ sends a q_{W}-qubit state $|O\rangle$ to the verifier $\widehat{V}$, where $|O\rangle = |0\rangle^{\otimes q_{\mathsf{W}}}$ for an honest prover;

3.b The verifier $\widehat{V}$ performs a multiple-controlled adder, where the q_{W}-qubit control register is $\widehat{\mathsf{M}}$ (containing $|O\rangle$), the $\lceil \log r \rceil$-qubit target register is $\widehat{\mathsf{T}}$, and the adder is activated if $|O\rangle = |0\rangle^{\otimes q_{\mathsf{W}}}$;

3.c The verifier $\widehat{V}$ performs a SWAP gate between registers $\widehat{\mathsf{M}}$ and $\widehat{\mathsf{W}}$;

4. The verifier $\widehat{V}$ measures the registers $\widehat{\mathsf{S}}$ and $\widehat{\mathsf{T}}$ in the computational basis, with the outcomes denoted as $\mathrm{Cnt}_{\mathrm{acc}}$ and $\mathrm{Cnt}_{\mathrm{clean}}$, respectively;

5. The verifier $\widehat{V}$ accepts if both $\mathrm{Cnt}_{\mathrm{clean}} = r - 1$ and $\mathrm{Cnt}_{\mathrm{acc}} = r$ are satisfied.

applying Equation (3.10) (by substituting X_i with X_i'), we can obtain that Protocol 3.2 accepts with probability at most 2^{-k} by choosing $r = O\Big(k/\log \frac{1}{1-(c-s)^2/2}\Big)$. This choice results in $m' = r(m+2) = O\Big(km/\log \frac{1}{1-(c-s)^2/2}\Big)$. □

3.4 Lower bounds for QIPL and $\mathsf{QIP_UL}$

Our lower bounds for QIPL and $\mathsf{QIP_UL}$ are motivated by the prior works on space-bounded classical interactive proofs, particularly those involving either private coins [CL95] or public coins [For89, FL93, Con92, GKR15, CR23]:[29]

- **Private-coin proof systems vs. QIPL:** Soundness against classical messages may not extend to quantum messages for private-coin proof systems. This is because the prover could generate shared entanglement with the verifier, potentially leaking information about the private coins. Consequently, space-bounded private-coin classical interactive proofs are simulatable only by QIPL proof systems (Theorem 3.14).
- **Public-coin proof systems vs. $\mathsf{QIP_UL}$:** Public coins can be simulated by halves of EPR pairs sent from the verifier in $\mathsf{QIP_UL}$ proof systems, thus avoiding the soundness issue. However, the verifier is limited to sending only $O(\log n)$ halves of EPR pairs, leading to a limitation on completeness. Therefore, only the work of [For89] can be simulated by $\mathsf{QIP_UL}$ proof systems (Lemma 3.15 and thus Theorem 3.3).

In the remainder of this subsection, we first establish that $\mathsf{NP} \subseteq \mathsf{QIPL}$ in Section 3.4.1, and then demonstrate that $\mathsf{SAC}^1 \cup \mathsf{BQL} \subseteq \mathsf{QIP_UL}$ in Section 3.4.2.

3.4.1 NP is in QIPL

Our approach in Theorem 3.14 draws inspiration from [CL95, Lemma 2]. Soundness against classical messages is guaranteed by *the fingerprinting lemma* [Lip90], which is used for comparing multisets through short fingerprints (see Section 2.4). To ensure soundness against quantum

[29] Space-bounded classical interactive proofs with $\mathrm{poly}(n)$ public coins are in P, see [Con92, Theorem 6].

messages in private-coin proof systems, the verifier must measure the received quantum message in the computational basis at the beginning of each action.

Theorem 3.14. $\mathsf{NP} \subseteq \mathsf{QIPL}_m$, *where* $m(n)$ *is some polynomial in* n.

Proof. We begin by noting that 3-SAT is NP-hard under logspace reductions (Lemma 2.12), and QIPL is also closed under logspace reductions.[30] It thus suffices to establish that 3-SAT is in QIPL.

To verify whether a 3-SAT instance $\phi = C_1 \wedge \cdots \wedge C_k$ is satisfied by an assignment α, we encode $\phi(\alpha)$ as a collection of $3k$ tuples (l, i, v), denoted as $\mathrm{Enc}(\phi(\alpha))$. In this encoding, each (l, i, v) represents the literal $l \in \{x_j\}_{j\in[n]} \cup \{\neg x_j\}_{j\in[n]}$ in the clause C_i is assigned the value $v \in \{\top, \bot\}$. Hence, $\mathrm{Enc}(\phi(\alpha))$ forms a multiset of $\ell = 3k$ elements, with each element representable by $b = 2\lceil \log n \rceil + \lceil \log k \rceil + 1 = O(\log(kn))$ bits. For convenience, let $a_{(l,i,v)}$ denote the non-negative integer that encodes the triple (l, i, v), and let $\mathrm{var}(l)$ be the variable in the literal l. Next, we proceed with the detailed QIPL protocol, as outlined in Protocol 3.3.

To establish the correctness of Protocol 3.3, we first observe that since k is a polynomial in n, the integer $2(bl)^2$ can be represented using $O(\log n)$ bits. Following the argument in the proof [CL95, Lemma 2],[31] a classical logspace verifier can find the prime p and the integer r in Step 1 of Protocol 3.3 with probability at least $3/4$, using $O(\log n)$ random bits. Since a classical operation depending on r random bits can be simulated by a corresponding unitary controlled by the state $|+\rangle^{\otimes r}$, Step 1 can be implemented using $O(\log n)$ ancillary qubits, which remain untouched throughout the rest of the proof system. Additionally, because the state $|\psi_{\mathrm{var}l,C_i}\rangle$, which encodes the triple (l, i, v), requires at most $O(\log n)$ qubits, space-bounded almost-unitary quantum circuits suffice to implement the verifier's actions in Protocol 3.3.

Although there are $3k + k = 4k$ rounds in Protocol 3.3, the verifier's actions are of only constantly many kinds. Therefore, the verifier's mapping is logspace computable. We now turn to the analysis required to finish the proof:

- For *yes* instances, where the 3-SAT formula ϕ is satisfiable, there is a prover strategy such that the verifier in Protocol 3.3 accepts with certainty. Additionally, the intermediate measurement outcome u^* corresponds to this prover strategy, specifically the classical messages sent by the prover, implying that $\omega(V)|^{u^*} = 1$.

- For *no* instances, we first bound the probability that the verifier in Protocol 3.3 accepts an unsatisfiable ϕ, assuming the prover sends only classical messages. This event may happen if: (1) the verifier fails to successfully choose the random prime p and the random integer r; or (2) the two multisets sent by the prover – specifically in Step 2 and Step 3 of Protocol 3.3 – are unequal but still yield the same fingerprint. According to the proof of [CL95, Lemma 2], the former occurs with probability at most $1/4$. And the fingerprinting lemma (Lemma 2.13) ensures that the latter occurs with probability at most

$$O\left(\frac{\log b + \log \ell}{b\ell} + \frac{1}{b^2\ell}\right) = O\left(\frac{\log\log(kn) + \log(3k)}{\log(kn)\cdot 3k} + \frac{1}{\log(kn)^2 \cdot 3k}\right) = O\left(\frac{1}{k}\right).$$

  Therefore, the acceptance probability is at most $1/4 + O(1/k) \leq 1/3$.

  Next, let $\omega(V)|_u$ be the verifier V's maximum acceptance probability conditioned on the intermediate measurement outcome u. A direct calculation shows that $\omega(V)$ is a convex

[30] QIPL is closed under logspace reductions if, for any logspace reduction R such that $\mathcal{I} \leq_{\mathrm{L}}^m \mathcal{I}'$, it holds that $\mathcal{I} \in \mathsf{QIPL}$ if $\mathcal{I}' \in \mathsf{QIPL}$. Let $\mathcal{M}_{\mathcal{I}'}$ and $\mathcal{M}_R$ be the deterministic logspace Turing machines that compute (the description of) the verifier's mapping associated with $\mathcal{I}'$ and the reduction R, respectively. The closure under logspace reductions for QIPL is then achieved by considering the concatenation $\mathcal{M}_{\mathcal{I}} \coloneqq \mathcal{M}_R \circ \mathcal{M}_{\mathcal{I}'}$. A similar argument applies to a weaker notion of the uniformity of the verifier's mapping discussed in Footnote 22.

[31] See the first paragraph on Page 515 in [CL95] for the details.

Protocol 3.3: A QIPL proof system for 3-SAT.

Parameters: $\phi(\alpha)$ is a 3-SAT formula ϕ with an assignment α; n and k are the number of variables and clauses in ϕ, respectively.

1. The verifier V chooses a prime p and an integer r uniformly at random from the intervals $[(b\ell)^2, 2(b\ell)^2]$ and $[1, p-1]$, respectively. The verifier then initializes the (partial) fingerprints $F_{\text{var}} = 1$ and $F_{\text{cl}} = 1$.

2. Consistency Check (for each variable) **:**

- The prover P sends the triples (l, i, v) in $\text{Enc}(\phi(\alpha))$, represented as quantum states $|\psi_{\text{var}(l),C_i}\rangle$, to the verifier V, ordered by the variable $\text{var}(l)$ in the literal l and then by the clause index i;
- For each state $|\psi_{\text{var}(l),C_i}\rangle$ received **:**
  - **2.a** V measures the state $|\psi_{\text{var}(l),C_i}\rangle$ in the computational basis, with the measurement outcome denoted by the triple (l, i, v);
  - **2.b** V rejects if the following conditions hold: (i) (l, i, v) is not the first triple in $\text{Enc}(\phi(\alpha))$, (ii) $\text{var}(l) = \text{var}(l')$, and (iii) the value v and v' are inconsistent;
  - **2.c** V updates the fingerprint $F_{\text{var}} = F_{\text{var}} \cdot \left(a_{(l,i,v)} + r\right) \mod p$;
  - **2.d** V sends the previous triple (l', i', v') back to P, if applicable. V then retains the current triple (l, i, v) in its private memory, unless (l, i, v) is the last triple in $\text{Enc}(\phi(\alpha))$.

3. Satisfiability Check (for each clause) **:**

- The prover P sends the triples (l_1, i, v_1), (l_2, i, v_2), and (l_3, i, v_3) in $\text{Enc}(\phi(\alpha))$, represented as states $|\psi_{C_i}\rangle$, to the verifier V, ordered by the clause index i;
- For each clause C_i, with the state $|\psi_{C_i}\rangle$ received **:**
  - **3.a** V measures the state $|\psi_{C_i}\rangle$ in the computational basis, where this state is expected to be $|\psi_{\text{var}(l_1),C_i}\rangle \otimes |\psi_{\text{var}(l_2),C_i}\rangle \otimes |\psi_{\text{var}(l_3),C_i}\rangle$. The measurement outcomes are denoted by the triples (l_1, i, v_1), (l_2, i, v_2), and (l_3, i, v_3);
  - **3.b** If $v_1 \vee v_2 \vee v_3 = \perp$, V rejects;
  - **3.c** V updates the fingerprints $F_{\text{cl}} = F_{\text{cl}} \cdot \prod_{j\in[3]} \left(a_{(l_j,i,v_j)} + r\right) \mod p$;
  - **3.d** V returns the triples (l_1, i, v_1), (l_2, i, v_2), and (l_3, i, v_3) to P.

4. The verifier V accepts if the fingerprints $F_{\text{var}} = F_{\text{cl}}$; otherwise, it rejects.

combination of $\omega(V)|_u$ over all obtainable measurement outcomes u. Since each verifier action begins with a measurement that forces the prover's message to be classical, we have:

$$\omega(V) = \sum_{u\in\mathcal{J}_{\text{obt}}} p_u \cdot \omega(V)|_u \leq \sum_{u\in\mathcal{J}_{\text{obt}}} p_u \cdot \frac{1}{3} = \frac{1}{3}.$$

Here, p_u denotes the probability of obtaining the measurement outcome u, $\mathcal{J}_{\text{obt}}$ represents the index set of all obtainable intermediate measurement outcomes, and the inequality follows from the established soundness against classical messages.

In conclusion, we complete the proof by establishing that 3-SAT $\in \mathsf{QIPL}_{8k}[1, 1/3]$. □

3.4.2 $\mathsf{SAC}^1 \cup \mathsf{BQL}$ is in $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$

Our approach follows [For89, Section 3.4]. To establish Theorem 3.3, along with error reduction for $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ (Lemma 3.13) and $\mathsf{BQL} \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}$, we need to prove the following:

Lemma 3.15. $\mathsf{SAC}^1 \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{O(\log n)}[1, 1 - 1/p(n)]$, *where* $p(n)$ *is some polynomial in* n.

Proof. For any promise problem $\mathcal{I} = (\mathcal{I}_{\text{yes}}, \mathcal{I}_{\text{no}})$ in SAC^1 where $\mathcal{I}_{\text{yes}} \cup \mathcal{I}_{\text{no}} = \{0,1\}^*$, it suffices to consider the corresponding (uniform) SAC^1 circuit evaluation problem, as defined in Defini-

tion 2.14. Let C be the (uniform) SAC^1 circuit associated with $\mathcal{I}$, taking $x \in \mathcal{I}$ as input, such that $C(x) = 1$ if and only if $x \in \mathcal{I}_{\text{yes}}$.

We now present the $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof system for SAC^1, as detailed in Protocol 3.4.

Protocol 3.4: A $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ proof system for (uniform) SAC^1.

---

1. The prover P and verifier V start at the output level of the circuit C **:**

- **If** the top gate G is an OR gate **:**
  - **2.1** P selects one of G's child gates;
  - **2.2** P sends the selection to V by sending a quantum state $|\psi\rangle$;
- **If** the top gate G is an AND gate **:**
  - **2.a** V selects one of the two child gate of G uniformly at random;
  - **2.b** V send the selection to P using half of an EPR pair;

2. At each intermediate level of the circuit C, the prover P and verifier V repeat this process on the selected child gate, as outlined in Step 1.

3. At the input level of the circuit C **:**

3.1 The verifier V measures the register containing the currently selected child gate in the computational basis. Here, the measurement outcomes corresponds to either x_j or $\neg x_j$ for some $j \in [n]$;

3.2 The verifier V checks the following **:**

- For **an input** x_i: V accepts if $x_i = 1$; otherwise, V rejects;
- For **a negation of an input** x_i: V accepts if $x_i = 0$; otherwise, V rejects.

To establish the correctness of Protocol 3.4, we first observe that the depth of the circuit C is $O(\log n)$, implying that the size is polynomial in n. Consequently, each prover's selection can be represented by an $O(\log n)$-bit string. This observation also implies that Protocol 3.4 has $O(\log n)$ rounds, but the verifier's actions are of only three kinds. Therefore, the verifier's mapping is logspace computable. We now turn to the analysis required to complete the proof:

- For *yes* instances, for any choices made by the verifier V at the AND gates, there exist corresponding choices at the OR gates that lead to the circuit C to accept. Therefore, the prover P has a winning strategy for all verifier choices, ensuring that the verifier V accepts with certainty.

- For *no* instances, since the computational paths in the circuit C do not interfere with each other, it suffices to establish soundness against classical messages. Note that certain verifier choices will cause V to reject. Given the $O(\log n)$ depth of the circuit C, V makes one choice out of two at each AND gate, resulting in a rejection probability of at least $2^{-O(\log n)} = 1/p(n)$ for some polynomial $p(n)$. Therefore, the verifier accepts with probability at most $1 - 1/p(n)$. □

4 Constant-message space-bounded quantum interactive proofs

In this section, we investigate space-bounded quantum interactive proof systems with a *constant* number of messages. Building on the definitions in Section 3.1, we define the classes $\mathsf{QIPL}_{\mathrm{O}(1)}$ and QMAML with *constant* promise gap as follows:

$$\mathsf{QIPL}_{\mathrm{O}(1)} \coloneqq \cup_{m \leq O(1)} \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[2/3, 1/3] \text{ and } \mathsf{QMAML} \coloneqq \mathsf{QMAML}[1, 1/3]. \tag{4.1}$$

Here, the class QMAML possesses *public-coin* three-message space-bounded *unitary* quantum interactive proofs, which is the space-bounded variant of the class QMAM introduced in [MW05].

Importantly, the definitions in Equation (4.1) align with those in Section 3.1 without loss of generality. Specifically, the two notions of space-bounded quantum interactive proof systems, QIPL and $\mathsf{QIP_UL}$, coincide when the number of messages is constant:

Remark 4.1 (The equivalence of $\mathsf{QIPL}_{O(1)}$ and $\mathsf{QIP_UL}_{O(1)}$). In a $\mathsf{QIPL}_{O(1)}$ proof system (or its reversible generalization, $\mathsf{QIPL}^{\diamond}_{O(1)}$, see Remark 3.7), the number of turns (i.e., messages) is a constant. Consequently, the verifier's actions during the execution of the proof system introduce only a constant number of additional environment registers, each containing $O(\log n)$ qubits. Therefore, a $\mathsf{QIPL}_{O(1)}$ proof system (even its reversible generalization) can be straightforwardly simulated by a $\mathsf{QIP_UL}_{O(1)}$ proof system with the same parameters $m(n)$, $c(n)$, and $s(n)$.

We establish the following upper bounds for $\mathsf{QIP_UL}$ and $\mathsf{QIPL}_{\mathrm{O}(1)}$. The first theorem, as detailed in Theorem 4.2, is obtained by combining Corollary 4.8 and Lemma 4.9, where the former is a direct corollary of the parallelization.[32] It is noteworthy that the second inclusion in Theorem 4.2 applies to the case where $m(n) \leq O(1)$ and $c(n) - s(n) \geq 1/\operatorname{poly}(n)$.

Theorem 4.2 ($\mathsf{QIP_UL} \subseteq \mathsf{P}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n)-s(n) \geq 1/\operatorname{poly}(n)$, and $1 \leq m(n) \leq \operatorname{poly}(n)$. Then, it holds that*

$$\mathsf{QIP_UL}_m[c,s] \subseteq \mathsf{QIPL}_3\left[1, 1-\frac{1}{q(n)}\right] \subseteq \mathsf{P}, \text{ where } q(n) := \frac{2(m(n)+1)^2}{(c(n)-s(n))^2}.$$

The second theorem, as stated in Theorem 4.3, is derived by combining Corollary 4.7 and Lemma 4.10. The class NC captures the power of (logspace-uniform) classical poly-logarithmic depth computation using bounded fan-in gates. Specifically, Corollary 4.7 is a direct consequence of the parallelization, while the NC containment (Lemma 4.10) follows directly from the celebrated $\mathsf{QIP} = \mathsf{PSPACE}$ result [JJUW11].

Theorem 4.3 ($\mathsf{QIPL}_{\mathrm{O}(1)} \subseteq \mathsf{NC}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq \Omega(1)$, and $1 \leq m(n) \leq O(1)$. Then, it holds that*

$$\mathsf{QIPL}_{\mathrm{O}(1)}[c,s] = \mathsf{QMAML} \subseteq \mathsf{NC}.$$

Unlike standard quantum interactive proofs, constant-message space-bounded quantum interactive proofs with constant promise gap ($\mathsf{QIPL}_{\mathrm{O}(1)}$) are unlikely to be as computationally powerful as their polynomial-message counterparts (QIPL). Furthermore, space-bounded quantum interactive proofs (QIPL) appears to be more powerful than their unitary counterparts ($\mathsf{QIP_UL}$). These distinctions align with the widely believed conjectured separations $\mathsf{NC} \subsetneq \mathsf{P} \subsetneq \mathsf{NP}$.

In the remainder of this section, we first demonstrate error reduction for $\mathsf{QIPL}_{\mathrm{O}(1)}$ via parallel repetition in Section 4.1. Then, Section 4.2 provides the parallelization technique for $\mathsf{QIP_UL}$ and $\mathsf{QIPL}_{\mathrm{O}(1)}$, drawing inspiration from [KKMV09], with a focus on the turn-halving lemma (Lemma 4.5) and its corollaries. Next, we proceed to establish an upper bound for $\mathsf{QIPL}_{\mathrm{O}(1)}$ with weak error bounds in Section 4.3. Finally, Section 4.4 presents the equivalence of $\mathsf{QIPL}_{\mathrm{O}(1)}$ and QMAML (Corollary 4.11) and the NC containment (Lemma 4.10), using a simplified version of the turn-halving lemma.

4.1 Error reduction for $\mathsf{QIPL}_{\mathrm{O}(1)}$ via parallel repetition

Beyond the sequential repetition approach presented in the proof of Lemma 3.13, another common method for error reduction is the parallel repetition of the original proof system $P \rightleftharpoons V$.

[32] Importantly, Theorem 4.2 holds only when we define the class $\mathsf{QIP_UL}$ with the strong notion of uniformity for the verifier's mapping, as detailed in Section 3.1. If the verifier's mapping satisfies only a weaker notion of uniformity, as specified in Footnote 22, then the verifier's action in the resulting proof system may not be space-bounded. This is because its description might not be produced by a logspace deterministic Turing machine.

In this approach, k pairs of provers and verifiers execute $P \rightleftharpoons V$ in parallel, where all k provers are independent only when they are honest. However, when adapting this approach for QIPL (or $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$), the message size in the parallelized protocol $P' \rightleftharpoons V'$ becomes $O(k \log n)$, meaning $P' \rightleftharpoons V'$ remains a QIPL (or $\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$) proof system only when k is constant.

Next, we provide the formal statement and its proof inspired by [VW16, Section 4.3]:

Lemma 4.4 (Error reduction for $\mathsf{QIPL}_{O(1)}$ via parallel repetition)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\operatorname{poly}(n)$, and $1 \leq m(n) \leq O(1)$. For any constant $k(n)$, it holds that*

$$\mathsf{QIPL}_m[c, s] \subseteq \mathsf{QIPL}_m\left[c^k, s^k\right].$$

Proof. For convenience, we prove the inclusion for $\mathsf{QIPL}^{\diamond}_{O(1)}$ proof systems, which serves as a reversible generalization of $\mathsf{QIPL}_{O(1)}$ (see Remark 3.7) and is equivalent to it (see Remark 4.1).

For any l-round proof system $P \rightleftharpoons V$ with completeness c and soundness s, corresponding to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}^{\diamond}_m[c, s]$, the maximum acceptance probability of the verifier, denoted by $\omega(V)$, serves as the objective function in the SDP formulation provided in Lemma 3.9. See also Remark 3.10 for the applicability of the SDP to $\mathsf{QIPL}^{\diamond}$ proof systems.

Now, consider a k-fold parallel repetition of $P \rightleftharpoons V$, involving k verifiers $V^{(i)} \coloneqq \left(V_1^{(i)}, \cdots, V_{l+1}^{(i)}\right)$ for $i \in [k]$. Let $V^{(1)} \otimes \cdots \otimes V^{(k)}$ be the combined verifier, obtained by executing $V^{(1)}, \cdots, V^{(k)}$ in parallel, with the output bit being the AND of the output bits of $V^{(1)}, \cdots, V^{(k)}$. It follows that

$$\omega\left(V^{(1)} \otimes \cdots \otimes V^{(k)}\right) \geq \omega\left(V^{(1)}\right) \cdots \omega\left(V^{(k)}\right),$$

since dishonest provers $P^{(1)}, \cdots, P^{(k)}$ may apply entangled actions. However, due to the strong duality of the SDP program for computing $\omega\left(V^{(i)}\right)$ for each $i \in [k]$, the equality also holds:

$$\omega\left(V^{(1)} \otimes \cdots \otimes V^{(k)}\right) = \omega\left(V^{(1)}\right) \cdots \omega\left(V^{(k)}\right). \tag{4.2}$$

In particular, the SDP formulation specified in Lemma 3.9 serves as the primal form of an SDP program for computing $\omega\left(V^{(i)}\right)$, with the corresponding dual form obtainable similarly to [VW16, Figure 4.7]. Following an argument analogous to Equation (4.49) through (4.53) in [VW16, Section 4.3], we arrive at Equation (4.2), with details omitted here. □

4.2 Parallelization via the turn-halving lemma

Our approach to do parallelization for $\mathsf{QIPL}_{O(1)}$ is inspired by [KKMV09, Section 4]. We begin with the key lemma that halves the number of turns (i.e., messages) in the proof system:

Lemma 4.5 (Turn-halving lemma)**.** *Let $s(n)$ and $m(n)$ be logspace-computable functions such that $0 \leq s(n) \leq 1$ and $1 \leq m(n) \leq \operatorname{poly}(n)$. Then, it holds that*

$$\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{4m+1}[1, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{2m+1}\left[1, (1 + \sqrt{s})/2\right].$$

Remark 4.6 (Limitations on the turn-halving lemma). The parallelization technique described in [KKMV09, Section 4] requires the verifier's actions to be *reversible*. This requirement implies that these actions must be implemented using either unitary or isometric quantum circuits. Therefore, it is straightforward to extend Lemma 4.5 to $\mathsf{QIPL}^{\diamond}$, a reversible generalization of QIPL. However, this extended version can be applied recursively at most a *constant* number of times; otherwise, the new verifier in the resulting proof system would no longer be space-bounded. Specifically, if the extended version is applied $\omega(1)$ times, it will introduce $2^{\omega(1)}$ environment registers, each containing $O(\log n)$ qubits, during a single verifier action.

Before presenting the proof, we provide two corollaries of the turn-halving lemma (Lemma 4.5) with their proofs to illuminate its applications, as stated in Corollary 4.7 and Corollary 4.8.

Corollary 4.7 ($\mathsf{QIPL}_{\mathrm{O}(1)} \subseteq \mathsf{QIPL}_3$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions with $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq \Omega(1)$, and $3 \leq m(n) \leq O(1)$. Then, it holds that*

$$\mathsf{QIPL}_m[c, s] \subseteq \mathsf{QIPL}_3[1, 1/16].$$

Proof. It suffices to prove the following inclusions:

$$\mathsf{QIPL}_m[c, s] \subseteq \mathsf{QIPL}_{m+2}\left[1, 1 - \frac{(c-s)^2}{2}\right] \subseteq \mathsf{QIPL}_3\left[1, 1 - \frac{(c-s)^2}{2(m+1)^2}\right] \subseteq \mathsf{QIPL}_3\left[1, \frac{1}{16}\right]. \quad (4.3)$$

The first inclusion in Equation (4.3) follows directly from Lemma 3.12. To show the last inclusion in Equation (4.3), we consider an r-fold (AND-type) parallel repetition of this three-message QIPL proof system. By applying Lemma 4.4 with $r = O\left(k/\log \frac{1}{1-(c-s)^2/(2(m+1)^2)}\right)$, we derive the following inclusions:

$$\mathsf{QIPL}_3\left[1, 1 - \frac{(c-s)^2}{2(m+1)^2}\right] \subseteq \mathsf{QIPL}_3\left[1, \left(1 - \frac{(c-s)^2}{2(m+1)^2}\right)^r\right] \subseteq \mathsf{QIPL}_3\left[1, \frac{1}{16}\right].$$

It remains to show the second inclusion in Equation (4.3). By repeatedly applying the turn-halving lemma (Lemma 4.5) l times, where l satisfies $2^l + 1 \leq m + 2 \leq 2^{l+1} + 1$, we obtain:

$$\mathsf{QIPL}_{m+2}\left[1, 1 - \frac{(c-s)^2}{2}\right] \subseteq \mathsf{QIPL}_{2^{l+1}+1}\left[1, 1 - \frac{(c-s)^2}{2}\right] \subseteq \mathsf{QIPL}_3\left[1, 1 - \frac{(c-s)^2}{2(m+1)^2}\right]. \quad (4.4)$$

Here, in the last inclusion, the reasoning behind the parameters – particularly $m(n)$, $c(n)$, and $s(n)$ – follows directly from the proof of [KKMV09, Lemma 4.2], so we omit the details.

Let $P \rightleftharpoons V$ be the original proof system. For the resulting proof system $P' \rightleftharpoons V'$, we now need to establish that the verifier's actions are space-bounded unitary circuits, and that the verifier's mapping is logspace computable. This follows because (1) an operation depending on r random coins can be simulated by applying a corresponding unitary controlled by $|+\rangle^{\otimes r}$, meaning that simulating l random coins in all of the verifier's actions requires l ancillary qubits; and (2) since a logspace deterministic Turing machine (DTM) can produce the description of all verifier's actions in $P \rightleftharpoons V$, there is another logspace DTM which can produce the description of each verifier's action in $P' \rightleftharpoons V'$. □

Corollary 4.8 ($\mathsf{QIP}_{\mathrm{U}}\mathsf{L}$ is parallelized to three messages with weaker error bounds)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions with $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\operatorname{poly}(n)$, and $3 \leq m(n) \leq \operatorname{poly}(n)$. Then, it holds that*

$$\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_3\left[1, 1 - \frac{1}{q(n)}\right] \text{ where } q(n) := \frac{2(m(n)+1)^2}{(c(n)-s(n))^2}.$$

Proof. It suffices to prove the following inclusions:

$$\mathsf{QIP}_{\mathrm{U}}\mathsf{L}_m[c, s] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_{m+2}\left[1, 1 - \frac{(c-s)^2}{2}\right] \subseteq \mathsf{QIP}_{\mathrm{U}}\mathsf{L}_3\left[1, 1 - \frac{(c-s)^2}{2(m+1)^2}\right]. \quad (4.5)$$

The first inclusion in Equation (4.5) follows directly from Lemma 3.12. The second inclusion in Equation (4.5) is achieved using a similar approach as that used to prove Equation (4.4) in Corollary 4.7. Finally, we need to show that the verifier's actions in the resulting proof system are space-bounded unitary circuits, and that the verifier's mapping is logspace computable. By noticing that $l = O(\log n)$, we can achieve this using reasoning analogous to the proof of Corollary 4.7, and we omit the details for brevity. □

4.2.1 Proof of the turn-halving lemma

Next, we proceed with the detailed proof of the turn-halving lemma (Lemma 4.5):

Proof of Lemma 4.5. Our proof strategy is inspired by the proof of [KKMV09, Lemma 4.1]. For any $(4m+1)$-message $\mathsf{QIP_UL}$ proof system $P \rightleftharpoons V$ with completeness c and soundness s, corresponding to a promise problem $\mathcal{I} \in \mathsf{QIP_UL}$, we can construct a new $(2m+1)$-message proof system $\widehat{P} \rightleftharpoons \widehat{V}$. We use the notations specified in Figure 4.1 to denote the snapshot states on registers Q, M, and W (resp., $\widehat{\mathsf{Q}}$, $\widehat{\mathsf{M}}$, and $\widehat{\mathsf{W}}$) during the execution of $P \rightleftharpoons V$ (resp., $\widehat{P} \rightleftharpoons \widehat{V}$), with slight adjustments for convenience compared to Figure 3.1 in Section 3.2.

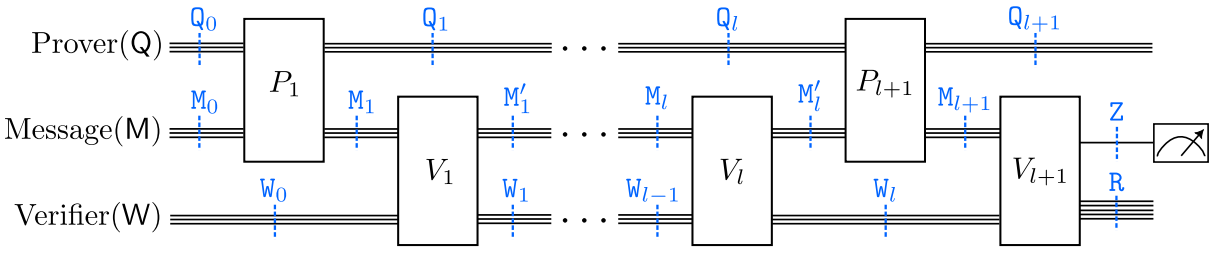

Figure 4.1: A $(2l+1)$-turn space-bounded unitary quantum interactive proof system.

We now describe the new proof system $\widehat{P} \rightleftharpoons \widehat{V}$ in an intuitive manner: the verifier $\widehat{V}$ first receives the snapshot state $\rho_{\mathtt{M}_{m+1}\mathtt{W}_m}$, which corresponds to the state after the prover P sent the $(2m+1)$-st message in $P \rightleftharpoons V$. The verifier $\widehat{V}$ then executes $P \rightleftharpoons V$ either forward or backward from the given snapshot, with equal probability. In the forward execution, $\widehat{V}$ accepts if V accepts; while in the backward execution, $\widehat{V}$ accepts if $\widehat{\mathtt{W}}_0$ contains the all-zero state.[33] The detailed proof system $\widehat{P} \rightleftharpoons \widehat{V}$ is presented in Protocol 4.1.

Protocol 4.1: A QIPL proof system for halving the number of messages in $P \rightleftharpoons V$.

1. The verifier $\widehat{V}$ receives the snapshot state $\rho_{m+1} = \rho_{\mathtt{M}_{m+1}\mathtt{W}_m}$ from the prover $\widehat{P}$, and then transfers the qubits corresponding to $\mathtt{W}_m$ in ρ_{m+1} to its private register $\widehat{\mathsf{W}}$;

2. The verifier $\widehat{V}$ chooses $b \in \{0,1\}$ uniformly at random, and executes the original proof system $P \rightleftharpoons V$ either forward (if $b=0$) or backward (if $b=1$) **:**

- Forward execution *of* $P \rightleftharpoons V$ $(b=0)$ **:**
  - **2.1** $\widehat{V}$ applies V_{m+1} to $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$, and then sends b and $\rho_{\widehat{\mathsf{M}}}$ to $\widehat{P}$;
  - **2.2 For** $j \leftarrow m+2$ **to** $2m$ **:**
    - $\widehat{V}$ receives $\rho_j = \rho_{\mathtt{M}_j}$ from $\widehat{P}$, applies V_j on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$, and sends $\rho_{\widehat{\mathsf{M}}}$ to $\widehat{V}$;
  - **2.3** $\widehat{V}$ receives $\rho_{2m+1} = \rho_{\mathtt{M}_{2m+1}}$ from $\widehat{P}$, applies V_{2m+1} on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$. $\widehat{V}$ accepts if $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$ contains an accepting state of $P \rightleftharpoons V$; otherwise, it rejects;
- Backward execution of $P \rightleftharpoons V$ $(b=1)$ **:**
  - **2.a** $\widehat{V}$ sends b and $\rho_{\widehat{\mathsf{M}}} = \rho_{\mathtt{M}_{m+1}}$ to $\widehat{P}$;
  - **2.b For** $j \leftarrow m$ **to** 2 **:**
    - $\widehat{V}$ receives $\rho_j = \rho_{\mathtt{M}_j}$ from $\widehat{P}$, applies $V_j^\dagger$ on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$, and sends $\rho_{\widehat{\mathsf{M}}}$ to $\widehat{V}$;
  - **2.c** $\widehat{V}$ receives $\rho_1 = \rho_{\mathtt{M}_1}$ from $\widehat{P}$, applies $V_1^\dagger$ on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$. $\widehat{V}$ accepts if $\widehat{\mathsf{W}}$ contains the all-zero state; otherwise, it rejects;

It remains to establish the correctness of the proof system $\widehat{P} \rightleftharpoons \widehat{V}$. Since the verifier V's mapping (of $P \rightleftharpoons V$) is logspace computable, and in each turn, the verifier $\widehat{V}$'s action corresponds to one of two possible actions of V (depending on b),[34], the verifier $\widehat{V}$'s mapping is likewise logspace computable. Moreover, we need the following analyses:

[33] Since the proof system $\mathcal{A}$ begins with the prover P, the verifier $\widehat{V}$ does not need to measure $\widehat{\mathtt{M}}_0$.

[34] The random bit b can be simulated by a fresh qubit and an intermediate measurement, following an argument

- For *yes* instances, an honest prover $\widehat{P}$ can prepare the pure state $|\psi_{\mathtt{Q}_{m+1}\mathtt{M}_{m+1}\mathtt{W}_m}\rangle$, which corresponds to the state in $(\mathsf{Q},\mathsf{M},\mathsf{W})$ after the $(2m+1)$-st turn in $P \rightleftharpoons V$. Depending on the value of b, the prover $\widehat{P}$ then applies the corresponding prover's actions P_j (if $b=0$) or $P_j^\dagger$ (if $b=1$) in $P \rightleftharpoons V$ during the execution of Step 2 in Protocol 4.1. For any $x \in \mathcal{I}_{\text{yes}}$, as $(P \rightleftharpoons V)(x)$ accepts with certainty, it follows that $(\widehat{P} \rightleftharpoons \widehat{V})(x)$ also accepts with certainty.
- For *no* instances, let $|\psi\rangle$ be the state in $(\widehat{\mathsf{Q}},\widehat{\mathsf{M}},\widehat{\mathsf{W}})$ just after the first turn in $\widetilde{P} \rightleftharpoons \widehat{V}$. Let $\widetilde{P}_j^{(b)}$ be the prover $\widetilde{P}$'s action, which is an arbitrary unitary transformation on $(\widehat{\mathsf{Q}},\widehat{\mathsf{M}})$, at the $(2j-1)$-st turn for $2 \le j \le m+1$. We can then define unitary transformations $U^{(0)}$ and $U^{(1)}$ corresponding to the forward and backward execution of $P \rightleftharpoons V$, respectively:

$$U^{(0)} \coloneqq V_{2m+1}\widetilde{P}_{m+1}^{(0)}V_{2m}\cdots\widetilde{P}_2^{(0)} \text{ and } U^{(1)} \coloneqq V_1^\dagger\widetilde{P}_{m+1}\cdots V_m^\dagger\widetilde{P}_2. \tag{4.6}$$

Based on Equation (4.6), we define the snapshot state $|\Psi^{(b)}\rangle$ at Step 2.3 for $b=0$ and at Step 2.c for $b=1$, respectively, before the corresponding final measurement:

$$|\Psi^{(0)}\rangle \coloneqq \frac{1}{\sqrt{p_{\text{acc}}^{(0)}}}\Pi_{\text{acc}}^{(0)}U^{(0)}|\psi\rangle, \text{ where } p_{\text{acc}}^{(0)} \coloneqq \left\|\Pi_{\text{acc}}^{(0)}U^{(0)}|\psi\rangle\right\|_2^2 \text{ and } \Pi_{\text{acc}}^{(0)} \coloneqq |1\rangle\langle 1|_{\text{out}} = \Pi_{\text{acc}},$$
$$|\Psi^{(1)}\rangle \coloneqq \frac{1}{\sqrt{p_{\text{acc}}^{(1)}}}\Pi_{\text{acc}}^{(1)}U^{(1)}|\psi\rangle, \text{ where } p_{\text{acc}}^{(1)} \coloneqq \left\|\Pi_{\text{acc}}^{(1)}U^{(1)}|\psi\rangle\right\|_2^2 \text{ and } \Pi_{\text{acc}}^{(1)} \coloneqq |\bar{0}\rangle\langle\bar{0}|_{\widehat{\mathsf{W}}}.$$

As a result, the acceptance probability $p_{\text{acc}}^{(b)}$ for $b \in \{0,1\}$ can be expressed as:

$$\forall b \in \{0,1\},\ p_{\text{acc}}^{(b)} = \frac{1}{\left\|\Pi_{\text{acc}}^{(b)}U^{(b)}|\psi\rangle\right\|_2^2}\left|\langle\psi|U^{(b)\dagger}\Pi_{\text{acc}}^{(b)}U^{(b)}|\psi\rangle\right|^2 = \left|\langle\psi|U^{(b)\dagger}|\Psi^{(b)}\rangle\right|^2. \tag{4.7}$$

Note that the acceptance probability of the proof system $(\widetilde{P} \rightleftharpoons \widehat{V})(x)$ for $x \in \mathcal{I}_{\text{no}}$ can be written as $p_{\text{acc}} = \frac{1}{2}\left(p_{\text{acc}}^{(0)} + p_{\text{acc}}^{(1)}\right)$. Substituting Equation (4.7) into this equality, we obtain:

$$\begin{aligned}
p_{\text{acc}} &= \frac{1}{2}\left(\mathrm{F}\left(U^{(0)\dagger}|\Psi^{(0)}\rangle\langle\Psi^{(0)}|U^{(0)}, |\psi\rangle\langle\psi|\right)^2 + \mathrm{F}\left(U^{(1)\dagger}|\Psi^{(1)}\rangle\langle\Psi^{(1)}|U^{(1)}, |\psi\rangle\langle\psi|\right)^2\right) \\
&\le \frac{1}{2}\left(1 + \mathrm{F}\left(U^{(0)\dagger}|\Psi^{(0)}\rangle\langle\Psi^{(0)}|U^{(0)}, U^{(1)\dagger}|\Psi^{(1)}\rangle\langle\Psi^{(1)}|U^{(1)}\right)\right) \\
&\le \frac{1}{2}\left(1 + \left\|\Pi_{\text{acc}}U^{(0)}U^{(1)\dagger}|\Psi^{(1)}\rangle\right\|_2\right) \\
&\le \frac{1}{2}(1+\sqrt{s}).
\end{aligned}$$

Here, the second line follows from Lemma 2.6, and the last line is due to the fact that $p_{\text{acc}} = \left\|\Pi_{\text{acc}}U^{(0)}U^{(1)\dagger}|\Psi^{(1)}\rangle\right\|_2^2 \le s$, as guaranteed by the soundness condition of $P \rightleftharpoons V$. □

4.3 Weakness of $\mathsf{QIPL}_{\mathrm{O}(1)}$ with weak error bounds

We demonstrate an upper bound for $\mathsf{QIPL}_{\mathrm{O}(1)}$ with weak error bounds:

Lemma 4.9 ($\mathsf{QIPL}_{\mathrm{O}(1)}$ is in P)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions such that $0 \le s(n) < c(n) \le 1$, $c(n) - s(n) \ge 1/\operatorname{poly}(n)$, and $1 \le m(n) \le O(1)$, it holds that*

$$\mathsf{QIPL}_m[c,s] \subseteq \mathsf{P}.$$

Our approach parallels the proof of $\mathsf{QIP} \subseteq \mathsf{EXP}$, as outlined in [Wat16, Page 31] and originally established in [KW00, Section 6]. Specifically, the SDP program for characterizing constant-turn QIPL proof systems, as detailed in Lemma 3.9, has a dimension of $\operatorname{poly}(n)$. Therefore, we conclude a deterministic polynomial-time algorithm using a standard SDP solver.

similar to the proof of Theorem 3.14 concerning Step 1 in Protocol 3.3

Proof of Lemma 4.9. For any m-turn proof system $P \rightleftharpoons V$, with completeness c, soundness s, and m being even, which corresponds to a promise problem $\mathcal{I}$ in $\mathsf{QIPL}_m[c,s]$, we can utilize Lemma 3.9 to obtain an SDP program, as specified in Equation (3.2). This SDP program maximizes the verifier's maximum acceptance probability $\omega(V)$ over all choices of quantum states (variables) $\rho_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1}, \cdots, \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}$, where the number of rounds $l := m/2$. Since the number of turns $m(n) \leq O(1)$, the variables in this SDP collectively hold $O(\log n)$ qubits. Thus, we can compute a description of this SDP program in deterministic polynomial time.

Next, consider the Frobenius norm defined as $\|X\|_{\mathrm{F}} := \sqrt{\mathrm{Tr}(X^\dagger X)}$, and let $\{\sigma_i(X)\}$ be the singular values of a square matrix X. We then have the following:

$$\begin{aligned}\|\rho_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1} \otimes \cdots \otimes \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}\|_{\mathrm{F}} &= \sqrt{\mathrm{Tr}\left(\rho^2_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1} \otimes \cdots \otimes \rho^2_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}\right)} \\ &= \sqrt{\sum_{i=1}^{D} \sigma_i^2\left(\rho_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1} \otimes \cdots \otimes \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}\right)} \\ &\leq \sqrt{D}.\end{aligned}$$

Here, D represents the dimension of $\rho_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1} \otimes \cdots \otimes \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}$, which is bounded by $2^{O(\log n)}$, indicating that it is in $\mathrm{poly}(n)$. The last line follows from the fact that all singular values of the density matrix $\rho_{\mathtt{M}'_1\mathtt{W}_1\mathtt{E}_1} \otimes \cdots \otimes \rho_{\mathtt{M}'_l\mathtt{W}_l\mathtt{E}_1\cdots\mathtt{E}_l}$ are at most 1.

As a consequence, by employing the standard SDP solver based on the ellipsoid method (e.g., [GM12, Theorem 2.6.1]; see also [GLS93, Chapter 3]), we obtain an algorithm for approximately solving the SDP program in Equation (3.2), ensuring that the condition $\omega(V) \geq c(n)$ is satisfied. This algorithm runs in deterministic time $\mathrm{poly}(D) \cdot \mathrm{polylog}\left(\sqrt{D}/\varepsilon\right)$, or expressed as $\mathrm{poly}(D, \log(1/\varepsilon))$, outputting either an ε-approximate feasible solution $\hat{X}$ or a certificate indicating that no such solution exists. Particularly, the error parameter $\varepsilon(n)$ ensures that $\|\hat{X} - X\|_{\mathrm{F}} \leq \varepsilon(n)$ for some feasible solution X, with $\omega(V)|_{\hat{X}} \geq c(n) - \varepsilon(n)$, where $\omega(V)|_{\hat{X}}$ represents the objective function evaluated at $\hat{X}$. We conclude the proof by observing that $\varepsilon(n) \leq 1/\mathrm{poly}(n)$ holds as long as $c(n) - s(n) \geq 1/\mathrm{poly}(n)$. □

4.4 Weakness of $\mathsf{QIPL}_{\mathrm{O}(1)}$: QMAML and NC containment

We present an upper bound of $\mathsf{QIPL}_{\mathrm{O}(1)}$:

Lemma 4.10. $\mathsf{QIPL}_3[1, 1/16] \subseteq \mathsf{QMAML}^{\odot}[1, 5/8] \subseteq \mathsf{NC}$.

The proof of Lemma 4.10 relies crucially on the *single-coin* variant of public-coin three-message quantum interactive proofs $\mathsf{QMAML}^{\odot}$, where the second message is a single random coin. Specifically, the first inclusion corresponds to a space-bounded variant of $\mathsf{QIP}(3) \subseteq \mathsf{QMAM}$ [MW05, Section 5], and the turn-halving lemma (Lemma 4.5) naturally extends this result. The second inclusion is exactly a down-scaling version of $\mathsf{QMAM}^{\odot} \subseteq \mathsf{NC}(\mathrm{poly})$ [JJUW11].[35]

Furthermore, the first inclusion in Lemma 4.10 implies the following:[36]

Corollary 4.11. $\mathsf{QIPL}_3 = \mathsf{QMAML}$.

We now move to the proof of Lemma 4.10.

[35] This inclusion does not extend to the variant with $1/\mathrm{poly}(n)$ promise gap. Specifically, applying the parallel SDP solver in [JJUW11], the resulting algorithm runs in parallel time (i.e., circuit depth) $\mathrm{poly}\log(n) \cdot \mathrm{poly}(1/\epsilon)$, using $\mathrm{poly}(n)$ processors (i.e., circuit width), as noted in the first paragraph of [JY11]. Since the parameter ε for $\mathsf{QMAML}^{\odot}[1, 1 - 1/p(n)]$ is polynomially small, this algorithm does not run in NC as the parallel running time becomes $\mathrm{poly}(n)$. This issue also arises when applying width-independent parallel SDP solvers [JY11, AZLO16].

[36] The inclusion $\mathsf{QMAML}^{\odot}[1, 5/8] \subseteq \mathsf{QMAML}[1, 125/512] \subseteq \mathsf{QMAML}[1, 1/3]$ is obtained by applying three-fold parallel repetition (Lemma 4.4), where the second message in the resulting proof system is three random coins.

Proof of Lemma 4.10. We begin by proving the first inclusion using a variant of the turn-halving lemma (Lemma 4.5). Let $P \rightleftharpoons V$ denote the original QIPL_3 proof system, which acts on registers Q, M, and W, following the notations in Figure 4.1 with $l = 1$. We propose a $\mathsf{QMAML}^{\odot}$ proof system $\widehat{P} \rightleftharpoons \widehat{V}$, which acts on registers $\widehat{\mathsf{Q}}$, $\widehat{\mathsf{M}}$, and $\widehat{\mathsf{W}}$, as described in Protocol 4.2. It is noteworthy that this proof system is a simplified version of Protocol 4.1.

Protocol 4.2: A $\mathsf{QMAML}^{\odot}$ proof system for verifying a QIPL_3 proof system $P \rightleftharpoons V$.

1. The verifier $\widehat{V}$ receives the qubits contained in $\mathtt{W}_1$ from the prover $\widehat{P}$, and then transfers them to $\widehat{\mathsf{W}}$.

2. The verifier $\widehat{V}$ chooses $b \in \{0, 1\}$ uniformly at random and sends b to the prover $\widehat{P}$.

3. The verifier $\widehat{V}$ receives the qubits written in $\widehat{\mathsf{M}}$ from the prover $\widehat{P}$ **:**

- If $b = 0$, the verifier $\widehat{V}$ applies V_2 on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$. $\widehat{V}$ accepts if $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$ contains an accepting state of $P \rightleftharpoons V$, and rejects otherwise.
- If $b = 1$, the verifier $\widehat{V}$ applies $V_1^{\dagger}$ on $(\widehat{\mathsf{M}}, \widehat{\mathsf{W}})$. $\widehat{V}$ accepts if $\widehat{\mathsf{W}}$ contains the all-zero state, and rejects otherwise.

It remains to establish the correctness of Protocol 4.2. This is straightforward for *yes* instances. For *no* instances, the desired bound essentially follows from the inequality in Lemma 2.6, using reasoning similar to that in the proof of Lemma 4.5. We omit the details.

Next, we address the second inclusion. We start by observing that Protocol 4.2 aligns with the definition of single-coin quantum Arthur-Merlin games as described in [JJUW11, Section 2.4], with two key differences: the message length m is $\log(n)$ rather than $\mathrm{poly}(n)$, and the verifier is space-bounded instead of polynomial-time bounded. This proof system achieves completeness 1 and soundness 5/8 due to the first inclusion. Consequently, we can obtain the corresponding primal-dual SDP programs of dimension $\mathrm{poly}(n)$, as opposed to $\exp(\mathrm{poly}(n))$, following [JJUW11, Section 2.5]. Therefore, we conclude an NC containment by applying the parallel SDP solver from [JJUW11] to the resulting SDP programs of dimension $\mathrm{poly}(n)$. □

5 Space-bounded unitary quantum statistical zero-knowledge

We now introduce (honest-verifier) space-bounded unitary quantum statistical zero-knowledge, denoted as $\mathsf{QSZK_UL}$ and $\mathsf{QSZK_UL_{HV}}$, as specific types of space-bounded unitary quantum interactive proofs ($\mathsf{QIP_UL}$) that possess an additional statistical zero-knowledge property.

Before presenting our results, we start by defining the promise problem IndivProdQSD, which is analogous to QSD [Wat02] and $\textsc{GapQSD}_{\log}$ [LGLW23]:

Definition 5.1 (Individual Product State Distinguishability Problem, $\textsc{IndivProdQSD}[k, \alpha, \delta]$)**.** *Let $k(n)$, $\alpha(n)$, $\delta(n)$, and $r(n)$ be logspace computable functions such that $1 \le k(n) \le \mathrm{poly}(n)$, $0 \le \alpha(n), \delta(n) \le 1$, $\alpha(n) - \delta(n) \cdot k(n) \ge 1/\mathrm{poly}(n)$, and $1 \le r(n) \le O(\log n)$. Let $Q_1, \cdots, Q_k$ and $Q'_1, \cdots, Q'_k$ be polynomial-size unitary quantum circuits acting on $O(\log n)$ qubits, each with $r(n)$ specified output qubits. For $j \in [k]$, let σ_j and σ'_j denote the states obtained by running Q_j and Q'_j on the all-zero state $|\bar{0}\rangle$, respectively, and tracing out the non-output qubits, then the promise is that one of the following holds:*

- Yes *instances: Two k-tuples of quantum circuits $(Q_1, \cdots, Q_k)$ and $(Q'_1, \cdots, Q'_k)$ such that*
$$\mathrm{T}\big(\sigma_1 \otimes \cdots \otimes \sigma_k, \sigma'_1 \otimes \cdots \otimes \sigma'_k\big) \ge \alpha(n);$$
- No *instances: Two k-tuples of quantum circuits $(Q_1, \cdots, Q_k)$ and $(Q'_1, \cdots, Q'_k)$ such that*
$$\forall j \in [k], \quad \mathrm{T}\big(\sigma_j, \sigma'_j\big) \le \delta(n).$$

Additionally, we denote the *complement* of $\textsc{IndivProdQSD}[k(n), \alpha(n), \delta(n)]$, with respect to the chosen parameters $\alpha(n)$, $\delta(n)$, and $k(n)$, as $\overline{\textsc{IndivProdQSD}}$.

With these definitions in hand, we now provide our first theorem in this section:

Theorem 5.2 (The equivalence of $\mathsf{QSZK_UL}$ and BQL)**.** *The following holds:*

(1) *For any logspace-computable function $m(n)$ such that $1 \leq m(n) \leq \mathrm{poly}(n)$,*

$$\cup_{c(n)-s(n)\geq 1/\mathrm{poly}(n)} \mathsf{QSZK_UL_{HV}}[m, c, s] \subseteq \mathsf{BQL}.$$

(2) $\mathsf{BQL} \subseteq \mathsf{QSZK_UL} \subseteq \mathsf{QSZK_UL_{HV}}$.

The class $\mathsf{QSZK_UL}$ consists of space-bounded unitary quantum interactive proof systems that possess statistical zero-knowledge against *any* verifier, whereas QSZKL proof systems possess statistical zero-knowledge against only an *honest* verifier. Consequently, the inclusion in Theorem 5.2(2) is straightforward, following directly from these definitions. To establish the direction $\mathsf{QSZK_UL_{HV}} \subseteq \mathsf{BQL}$, we proceed by proving the following:

Theorem 5.3 ($\textsc{IndivProdQSD}$ is $\mathsf{QSZK_UL_{HV}}$-complete)**.** *The following holds:*

(1) *Let $c(n)$ and $s(n)$ be logspace computable functions such that $0 \leq s(n) < c(n) \leq 1$. For any logspace-computable function $m(n)$ such that $3 \leq m(n) \leq \mathrm{poly}(n)$,*

$$\overline{\textsc{IndivProdQSD}}[m/2, \alpha, 2\delta] \text{ is } \mathsf{QSZK_UL_{HV}}[m, c, s]\text{-hard.}$$

Here, $\alpha := (\sqrt{c} - \sqrt{s})^2/(2m-4)$ and δ is some negligible function.

(2) *Let $k(n)$, $\alpha(n)$ and $\delta(n)$ be logspace computable functions such that $1 \leq k(n) \leq \mathrm{poly}(n)$, $0 \leq \alpha(n), \delta(n) \leq 1$, and $\alpha(n) - \delta(n) \cdot k(n) \geq 1/\mathrm{poly}(n)$. Then, it holds that*

$$\textsc{IndivProdQSD}[k, \alpha, \delta] \in \mathsf{BQL} \subseteq \mathsf{QSZK_UL_{HV}}.$$

In the remainder of this section, we first provide the definition of honest-verifier space-bounded quantum statistical zero-knowledge proofs (the class $\mathsf{QSZK_UL_{HV}}$) in Section 5.1. Next, we establish that $\textsc{IndivProdQSD}$ is $\mathsf{QSZK_UL_{HV}}$-hard (Theorem 5.3(1)) in Section 5.2. Subsequently, we present the BQL upper bound for $\mathsf{QSZK_UL_{HV}}$ (Theorem 5.3(2)) in Section 5.3.

5.1 Definition of space-bounded unitary quantum statistical zero-knowledge

Our definition of (honest-verifier) space-bounded quantum statistical zero-knowledge follows that of [Wat02, Section 3.1]. In this framework, an honest-verifier space-bounded unitary quantum statistical zero-knowledge proof system is a space-bounded unitary quantum interactive proof system, as defined in Section 3.1, that satisfies an additional *zero-knowledge* property. Intuitively, the zero-knowledge property in $\mathsf{QIP_UL}$ proof systems requires that, after each message is sent, the quantum states representing the verifier's view – including snapshot states in the message register M and the verifier's private register W – should be approximately indistinguishable by a space-bounded unitary quantum circuit on accepted inputs.

We then formalize this notion. Consider a set $\{\rho_{x,i}\}$ of mixed states, we say that this state set is *logspace-preparable* if there exists a family of m-tuples $S_x := (S_{x,1}, \cdots, S_{x,m})$, where each $S_{x,i}$ for $i \in [m]$ is a space-bounded unitary quantum circuit (see Definition 2.8) with a specified collection of output qubits, such that for each input x and index i, the state $\rho_{x,i}$ is the mixed state obtained by running $S_{x,i}$ on the input state $|\bar{0}\rangle$, and then tracing out all non-output qubits. We refer to such $\{S_x\}_{x\in\mathcal{I}}$ as the *space-bounded simulator* for the promise problem $\mathcal{I}$.

Next, for any space-bounded quantum interactive proof system $P \rightleftharpoons V$, we define the verifier's view after the i-th turn, denoted by $\mathtt{view}_{P\rightleftharpoons V}(x, i)$, as the reduced state in registers (M, W) immediately after i messages have been exchanged, with the prover's private qubits traced out.

We are now ready for the formal definition:

Definition 5.4 (Honest-verifier space-bounded unitary quantum statistical zero-knowledge, $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace-computable functions of the input length $n := |x|$ such that $0 \leq s(n) < c(n) \leq 1$ and $1 \leq m(n) \leq \mathrm{poly}(n)$. A promise problem $\mathcal{I} = (\mathcal{I}_{\mathrm{yes}}, \mathcal{I}_{\mathrm{no}})$ is in $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}[m, c, s]$, if there exists an $m(n)$-message space-bounded unitary quantum interactive proof system $(P \rightleftharpoons V)(x)$ such that:*

- **Completeness**. *For any $x \in \mathcal{I}_{\mathrm{yes}}$, there exists an $m(n)$-message prover P such that*
$$\Pr\left[(P \rightleftharpoons V)(x) \text{ accepts}\right] \geq c(n).$$
- **Soundness**. *For any $x \in \mathcal{I}_{\mathrm{no}}$ and any $m(n)$-message prover P,*
$$\Pr\left[(P \rightleftharpoons V)(x) \text{ accepts}\right] \leq s(n).$$
- **Zero-knowledge**. *There exists a space-bounded simulator $\{S_x\}_{x \in \mathcal{I}}$ and a negligible function $\delta(n)$ such that for any $x \in \mathcal{I}_{\mathrm{yes}}$ and each message $i \in [m]$, the circuit $S_x(i)$ produces the corresponding state $\sigma_{x,i}$ satisfying*
$$\mathrm{T}(\sigma_{x,i}, \mathtt{view}_{P \rightleftharpoons V}(x, i)) \leq \delta(n).$$

We define $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}[m] := \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}\left[m, \frac{2}{3}, \frac{1}{3}\right]$ *and* $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}} := \cup_{m \leq \mathrm{poly}(n)} \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}[m]$.

Since the inequality condition in the zero-knowledge property holds independently for each message in Definition 5.4, error reduction via sequential repetition (Lemma 3.13) directly applies to an honest-verifier space-bounded quantum statistical zero-knowledge proof system, with the zero-knowledge property automatically preserved.

Remark 5.5 (Robustness of the zero-knowledge property in $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$). Let $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}$ denote a weaker version of $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$, where the threshold function $\delta(n) := (\sqrt{c} - \sqrt{s})^2/(2m^2)$,[37] rather than being negligible. While it is clear that $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}} \subseteq \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}$, the standard approach to establish the reverse direction does not apply to $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$.[38] Instead, the inclusion $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}} \subseteq \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$ only follows from $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}} = \mathsf{BQL}$ (Theorem 5.2).

5.2 $\overline{\textsc{IndivProdQSD}}$ is $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-hard

Instead of directly proving that $\overline{\textsc{IndivProdQSD}}$ is $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-hard, we establish a slightly stronger result: the promise problem $\overline{\textsc{IndivProdQSD}}$ is hard for the class $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}$ that contains $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$ (Remark 5.5), as detailed in Theorem 5.6. This result mirrors the relationship between $\overline{\mathrm{QSD}}$ and the class QSZK.

Theorem 5.6 ($\overline{\textsc{IndivProdQSD}}$ is $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}$-hard)**.** *Let $c(n)$, $s(n)$, and $m(n)$ be logspace computable functions such that $0 \leq s(n) < c(n) \leq 1$, $c(n) - s(n) \geq 1/\mathrm{poly}(n)$, and $3 \leq m(n) \leq \mathrm{poly}(n)$.*[39] *Then, it holds that*

$$\overline{\textsc{IndivProdQSD}}[\lceil m(n)/2 \rceil, \alpha(n), 2\delta(n)] \text{ is } \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}[m(n), c(n), s(n)]\text{-hard.}$$

Here, $\delta := (\sqrt{c} - \sqrt{s})^2/(2m^2)$ and $\alpha := (\sqrt{c} - \sqrt{s})^2/(2m - 4)$.

Before presenting the proof, we will first illustrate the properties of the simulator and explain the underlying intuition behind the proof. Our proof strategy follows some ideas from [Wat02, Section 5]. Consider a space-bounded quantum interactive proof system $P \rightleftharpoons V$ for a promise problem $\mathcal{I} \in \mathsf{QSZK}_{\mathrm{U}}\mathsf{L}^{\star}_{\mathrm{HV}}[m(n), c(n), s(n)]$ that is statistical zero-knowledge against an honest verifier. Without loss of generality, assume that the number of turns in $P \rightleftharpoons V$ is even. We use the notations introduced in Figure 3.1 and Section 3.2.

[37] This bound results from the reduction to the $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-hard problem $\textsc{IndivProdQSD}$, see Theorem 5.6.

[38] In particular, the polarization lemma for the trace distance [Wat02, Section 4.1] is not applicable in the space-bounded scenario due to message size constraints.

[39] Without loss of generality, we can assume that $m \geq 3$ by adding one or two dummy messages when $m < 3$, as discussed in Footnote 41.

Let us now focus on the space-bounded simulator $\{S_x\}_{x\in\mathcal{I}}$. Let $\xi'_0,\cdots,\xi'_l$ and $\xi_1,\cdots,\xi_{l+1}$ denote the simulator's approximation to the reduced snapshot states in registers (M,W) after the $(2j-1)$-st and the $(2j)$-th turn, respectively, during the execution of $P\rightleftharpoons V$, as specified in Figure 5.1. For *yes* instances, these states closely approximate the actual view of the verifier (the corresponding snapshot states) during the execution of $P\rightleftharpoons V$. However, there is no *direct* closeness guarantee for *no* instances. Consequently, we can assume that the state ξ_{l+1} satisfies $\mathrm{Tr}(|1\rangle\langle 1|_{\mathsf{Z}}\xi_{l+1}))=c(n)$ for *all* instances.

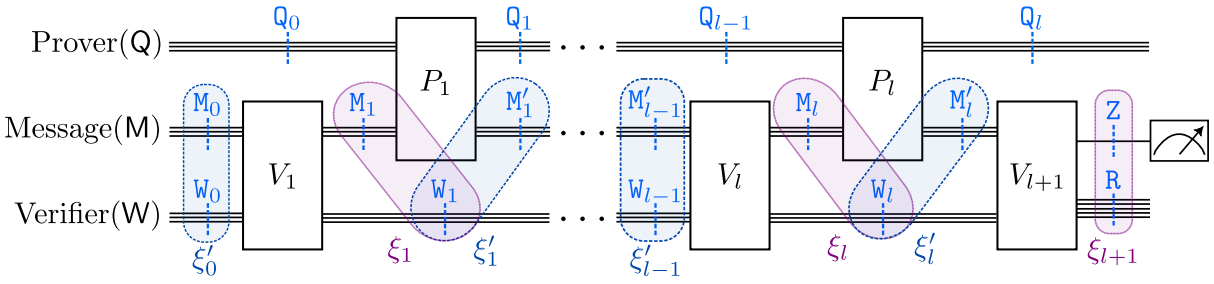

Figure 5.1: Quantum states $\xi'_0,\cdots,\xi'_l$ and $\xi,\cdots,\xi_{l+1}$ prepared by the simulator.

In addition, given that the verifier is always assumed to act honestly, we can take[40]

$$\xi'_0=(|0\rangle\langle 0|)^{\otimes(q_{\mathsf{M}}+q_{\mathsf{W}})}\text{ and }\xi_j=V_j\xi'_{j-1}V_j^\dagger\text{ for }j\in[l+1]. \tag{5.1}$$

Proof intuition. Notably, the space-bounded simulator $\{S_x\}_{x\in\mathcal{I}}$ essentially produces an approximation solution, in the form of snapshot states, to the SDP program Equation (3.2) for computing the maximum acceptance probability $\omega(V)$ of the space-bounded unitary quantum interactive proof systems $P\rightleftharpoons V$ for $\mathcal{I}\in\mathsf{QIP_UL}$. As we stated in the proof of Theorem 3.8, there are only two types of constraints: (1) Verifier's actions are honest; and (2) Prover's actions do not affect the verifier's private qubits.

As mentioned in Equation (5.1), these states produced by the simulator exactly satisfy the first type of constraints for all instances, but satisfy the second type of constraints *only* for *yes* instances. This observation leads to our proof and the hard problem IndivProdQSD. Specifically, we consider two tensor product states, each consisting of a polynomial number of $O(\log n)$-qubit states, where all components are defined in Figure 5.1:

$$\mathrm{Tr}_{\mathsf{M}}(\xi_1)\otimes\cdots\otimes\mathrm{Tr}_{\mathsf{M}}(\xi_l)\text{ and }\mathrm{Tr}_{\mathsf{M}}\big(\xi'_1\big)\otimes\cdots\otimes\mathrm{Tr}_{\mathsf{M}}\big(\xi'_l\big). \tag{5.2}$$

For *yes* instances, the zero-knowledge property ensures a component-wise closeness bound $\mathrm{Tr}_{\mathsf{M}}(\xi_j)\approx\mathrm{Tr}_{\mathsf{M}}(\xi'_j)$ for $j\in[l]$. For *no* instances, we need to show that the two states in Equation (5.2) are far from each other, given that $\omega(V)\le s(n)$. This follows directly from [Wat02, Lemma 15]. We state the counterpart result below and omit the detailed proof:

Proposition 5.6.1 (Adapted from [Wat02, Lemma 15])**.** *Let $P\rightleftharpoons V$ be an $m(n)$-turn space-bounded quantum interactive proof system, with even $m:=2l$, such that $\omega(V)\le s(n)$. Let $\xi'_0,\cdots,\xi'_l$ and $\xi_k,\cdots,\xi_{l+1}$ be the states produced by the simulators as defined in Figure 5.1. Assume that $\mathrm{Tr}(|\bar{0}\rangle\langle\bar{0}|_{\mathtt{M}_0\mathtt{W}_0}\xi'_0)=1$ and $\mathrm{Tr}(|1\rangle\langle 1|_{\mathsf{Z}}\xi_{l+1})=c$. Then, it holds that*

$$\mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_1)\otimes\cdots\otimes\mathrm{Tr}_{\mathsf{M}}(\xi_l),\mathrm{Tr}_{\mathsf{M}}\big(\xi'_1\big)\otimes\cdots\otimes\mathrm{Tr}_{\mathsf{M}}\big(\xi'_l\big)\big)\ge\frac{(\sqrt{c}-\sqrt{s})^2}{4(l-1)}.$$

Then, we proceed with the formal proof of Theorem 5.6:

Proof of Theorem 5.6. Let $P\rightleftharpoons V$ be an $m(n)$-turn honest-verifier unitary quantum statistical zero-knowledge proof system for a promise problem $\mathcal{I}\in\mathsf{QSZKL}^{\star}_{\mathrm{HV}}[m,c,s]$, with completeness

[40]Consequently, the simulator only needs to prepare ξ'_{j-1}, since ξ_j is obtained by applying V_j to this state.

$c(n)$ and soundness $s(n)$. Without loss of generality, we assume that m is even for all $x \in \mathcal{I}$.[41] Hence, we can denote the verifier's actions by $V_1, \cdots, V_{l+1}$ for $l = m/2$, and the verifier initiates the protocol. Let $\{\sigma_{x,i}\}_{x\in\mathcal{I}, i\in[m+2]}$ represent the mixed states produced by the simulator $\{S_x\}_{x\in\mathcal{I}}$, with the threshold function $\delta(n) := 1/m(n)^2$. For any $x \in \mathcal{I}$, we can define states $\xi'_0, \cdots, \xi'_l$ and $\xi_1, \cdots, \xi_{l+1}$ as illustrated in Figure 5.1:

- Initial state before executing $P \rightleftharpoons V$: $\xi'_0 := |\bar{0}\rangle\langle\bar{0}|_{\mathsf{M}_0\mathsf{W}_0}$.
- $(2j)$-th message for $j \in [l]$ in $P \rightleftharpoons V$: $\xi'_j := \sigma_{x,2j}$, where $\sigma_{x,2j}$ satisfies:
$$\forall x \in \mathcal{I}_{\text{yes}}, \quad \mathrm{T}(\sigma_{x,2j-1}, \mathtt{view}_{P\rightleftharpoons V}(x, 2j)) = \mathrm{T}\big(\sigma_{x,2j-1}, \rho_{\mathsf{M}_j\mathsf{W}_j}\big) \le \delta(n). \tag{5.3}$$
- $(2j+1)$-st message for $j \in [l]$ in $P \rightleftharpoons V$: $\xi_j := V_j \xi'_{j-1} V_j^\dagger$.
- State before the final measurement in $P \rightleftharpoons V$: $\xi_{l+1} := V_{l+1}\xi'_l V_{l+1}^\dagger$ satisfies
$$\mathrm{Tr}(|1\rangle\langle 1|_{\mathsf{Z}} \xi_{l+1}) = c(n).$$

Let $Q_1, \cdots, Q_k$ and $Q'_1, \cdots, Q'_k$ be polynomial-size unitary quantum circuits acting on $O(\log n)$ qubits which satisfy that $Q_j = S_{x,2j-1}$ and $Q'_j = S_{x,2j}$ for $j \in [l]$, and the output qubits are qubits in the verifier's private register W. It is evident that Q_j and Q'_j prepare the states $\mathrm{Tr}_{\mathsf{M}}(\xi_j)$ and $\mathrm{Tr}_{\mathsf{M}}\big(\xi'_j\big)$, respectively. We claim that the l-tuples $(Q_1, \cdots, Q_l)$ and $(Q'_1, \cdots Q'_l)$ form an instance of $\overline{\textsc{IndivProdQSD}}[l(n), \alpha(n), \delta'(n)]$, satisfying the following conditions:

$$\forall x \in \mathcal{I}_{\text{yes}}, \quad \mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_j), \mathrm{Tr}_{\mathsf{M}}\big(\xi'_j\big)\big) \le 2\delta = \frac{(\sqrt{c}-\sqrt{s})^2}{4l^2} := \delta' \text{ for } j \in [l]; \tag{5.4}$$

$$\forall x \in \mathcal{I}_{\text{no}}, \quad \mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_1) \otimes \cdots \otimes \mathrm{Tr}_{\mathsf{M}}(\xi_l), \mathrm{Tr}_{\mathsf{M}}\big(\xi'_1\big) \otimes \cdots \otimes \mathrm{Tr}_{\mathsf{M}}\big(\xi'_l\big)\big) \ge \frac{(\sqrt{c}-\sqrt{s})^2}{4(l-1)} := \alpha. \tag{5.5}$$

By substituting Equation (5.4) into Lemma 2.2, it follows that:

$$\begin{aligned} \mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_1) \otimes \cdots \otimes \mathrm{Tr}_{\mathsf{M}}(\xi_l), \mathrm{Tr}_{\mathsf{M}}\big(\xi'_1\big) \otimes \cdots \otimes \mathrm{Tr}_{\mathsf{M}}\big(\xi'_l\big)\big) &\le \sum_{j\in[l]} \mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_j), \mathrm{Tr}_{\mathsf{M}}\big(\xi'_j\big)\big) \\ &\le \frac{(\sqrt{c}-\sqrt{s})^2}{4l}. \end{aligned} \tag{5.6}$$

Consequently, by comparing Equations (5.4) to (5.6), we can conclude the parameter requirement of $\overline{\textsc{IndivProdQSD}}[l(n), \alpha(n), \delta'(n)]$, specifically that $\alpha(n) - \delta'(n) \cdot l(n) \ge 1/\operatorname{poly}(n)$.

It remains to establish Equation (5.4) and Equation (5.5). The latter follows directly from Proposition 5.6.1. To prove the former, note that the prover's actions do not affect the verifier's private register for *yes* instances, we thus derive the following for $j \in \{2, \cdots, l\}$:

$$\begin{aligned} \mathrm{T}\big(\mathrm{Tr}_{\mathsf{M}}(\xi_j), \mathrm{Tr}_{\mathsf{M}}\big(\xi'_j\big)\big) &\le \mathrm{T}\big(\xi_j, \xi'_j\big) \\ &\le \mathrm{T}\big(\xi_j, \rho_{\mathtt{M}_j\mathtt{W}_j}\big) + \mathrm{T}\Big(\rho_{\mathtt{M}_j\mathtt{W}_j}, \rho_{\mathtt{M}'_j\mathtt{W}_j}\Big) + \mathrm{T}\Big(\rho_{\mathtt{M}'_j\mathtt{W}_j}, \xi'_j\Big) \\ &= \mathrm{T}\Big(\xi'_{j-1}, \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}}\Big) + \mathrm{T}\Big(\rho_{\mathtt{M}_j\mathtt{W}_j}, \rho_{\mathtt{M}'_j\mathtt{W}_j}\Big) + \mathrm{T}\Big(\rho_{\mathtt{M}'_j\mathtt{W}_j}, \xi'_j\Big) \\ &\le \delta(n) + 0 + \delta(n) \\ &= 2\delta(n). \end{aligned}$$

Here, the first line follows from the data-process inequality (Lemma 2.3), the second line is due to the triangle inequality, the third line owes to the unitary invariance (Lemma 2.4) and the fact that $\rho_{\mathtt{M}_j\mathtt{W}_j} = V_j \rho_{\mathtt{M}'_{j-1}\mathtt{W}_{j-1}} V_j^\dagger$, and the fourth line is because of Equation (5.3). We complete the proof by noting that similar reasoning applies to the case of $j = 1$, using $\mathrm{T}(\xi_1, \rho_{\mathtt{M}_1\mathtt{W}_1}) = 0$ instead of at most $\delta(n)$. □

[41] If m is odd, we can add an initial turn to $P \rightleftharpoons V$ in which the verifier sends the all-zero state to the prover.

5.3 $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$ is in BQL

We will establish the hard direction in the equivalence of $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$ and BQL. The key lemma underlying the proof involves a logspace (many-to-one) reduction IndivProdQSD to an "existential" version of $\textsc{GapQSD}_{\log}$, where $\textsc{GapQSD}_{\log}$ is a BQL-complete problem (see Section 2.3). This reduction leads to a BQL containment for IndivProdQSD:

Lemma 5.7 (IndivProdQSD is in BQL)**.** *Let $k(n)$, $\alpha(n)$ and $\delta(n)$ be logspace computable functions such that $1 \leq k(n) \leq \mathrm{poly}(n)$, $0 \leq \alpha(n), \delta(n) \leq 1$, and $\alpha(n) - \delta(n) \cdot k(n) \geq 1/\mathrm{poly}(n)$. Then, it holds that*

$$\textsc{IndivProdQSD}[k(n), \alpha(n), \delta(n)] \in \mathsf{BQL}.$$

As $\overline{\textsc{IndivProdQSD}}$ is $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}}$-hard (Theorem 5.6), and given that $\mathsf{BQ}_{\mathrm{U}}\mathsf{L}$ is closed under complement [Wat99, Corollary 4.8] and the equivalence $\mathsf{BQL} = \mathsf{BQ}_{\mathrm{U}}\mathsf{L}$ [FR21], we can directly conclude the following corollary:

Corollary 5.8. $\mathsf{QSZK}_{\mathrm{U}}\mathsf{L}_{\mathrm{HV}} \subseteq \mathsf{BQL}$.

We now proceed with the formal proof of the key lemma:

Proof of Lemma 5.7. We first establish a logspace (many-to-one) reduction from IndivProdQSD to an "existential" version of $\textsc{GapQSD}_{\log}$. Let $(Q_1, \cdots, Q_k)$ and $(Q'_1, \cdots, Q'_k)$ be an instance of $\textsc{IndivProdQSD}[k, \alpha, \delta]$. For each $j \in [k]$, let σ_j and σ'_j denote the states obtained by running Q_j and Q'_j on the all-zero state $|\bar{0}\rangle$, respectively, and tracing out the non-output qubits. We now need to decide which of the following cases in Equation (5.7) and Equation (5.8) holds:

$$\mathrm{T}\big(\sigma_1 \otimes \cdots \otimes \sigma_k, \sigma'_1 \otimes \cdots \otimes \sigma'_k\big) \geq \alpha(n). \tag{5.7}$$

$$\forall j \in [k], \quad \mathrm{T}\big(\sigma_j, \sigma'_j\big) \leq \delta(n). \tag{5.8}$$

By combining Lemma 2.2 with Equation (5.7), we obtain:

$$\sum_{j \in [k]} \mathrm{T}\big(\sigma_j, \sigma'_j\big) \geq \mathrm{T}\big(\sigma_1 \otimes \cdots \otimes \sigma_k, \sigma'_1 \otimes \cdots \otimes \sigma'_k\big) \geq \alpha(n). \tag{5.9}$$

Applying an averaging argument to Equation (5.9), we can conclude that

$$\exists j \in [k], \quad \mathrm{T}\big(\sigma_j, \sigma'_j\big) \geq \alpha/k. \tag{5.10}$$

Clearly, a violation of Equation (5.10) implies a violation of Equation (5.7), without contradicting Equation (5.8). For each $j \in [k]$, the pair of circuits Q_j and Q'_j forms an instance of $\textsc{GapQSD}_{\log}$. The resulting promise problem is thus an "existential" version of $\textsc{GapQSD}_{\log}$, where *yes* instances satisfy Equation (5.10) and *no* instances satisfy Equation (5.8).

Next, we proceed by demonstrating the BQL containment. Given the equivalence of BQL and QMAL [FKL+16, FR21], it remains to establish a QMAL containment for this "existential" version of $\textsc{GapQSD}_{\log}$. The verification protocol is outlined in Protocol 5.1.

Protocol 5.1: A QMAL proof system for IndivProdQSD.

1. The verifier receives an index $j \in [k]$ from the prover.
2. The verifier executes the quantum logspace algorithm $\mathcal{A}$ for $\textsc{GapQSD}_{\log}[\alpha/k, \delta]$ underlying in Theorem 2.11, using the pair of circuits Q_j and Q'_j as the $\textsc{GapQSD}_{\log}$ instance. The verifier accepts (or rejects) if $\mathcal{A}$ accepts (or rejects).

To complete the proof, we establish the correctness of Protocol 5.1. Since the algorithm $\mathcal{A}$ is a BQL containment for $\textsc{GapQSD}_{\log}[\alpha/k, \delta]$ (Theorem 2.11), we conclude the following:

- For *yes* instances, Equation (5.10) ensures that there exists an $j \in [k]$ (the witness) such that $\mathrm{T}\big(\sigma_j, \sigma'_j\big) \geq \alpha/k$. Consequently, $\mathcal{A}$ accepts with probability at least $2/3$.

- For *no* instances, Equation (5.8) yields that for all $j \in [k]$, $\mathrm{T}(\sigma_j, \sigma'_j) \leq \delta$. This statement implies that $\mathcal{A}$ accepts with probability at most $1/3$. □

Acknowledgments

YL is grateful to Uma Girish for helpful discussion that inspired Question (d). This work was partially supported by MEXT Q-LEAP grant No. JPMXS0120319794. FLG was also supported by JSPS KAKENHI grants Nos. JP20H05966, 20H00579, 24H00071, and by MEXT JST CREST grant No. JPMJCR24I4. YL was further supported by JST SPRING grant No. JPMJSP2125 and acknowledges the "THERS Make New Standards Program for the Next Generation Researchers." HN was additionally supported by JSPS KAKENHI grants Nos. JP19H04066, JP20H05966, JP21H04879, and JP22H00522. QW was supported in part by the Engineering and Physical Sciences Research Council under Grant No. EP/X026167/1.

References

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 1st edition, 2009. 13, 14

[AGM+23] Eric Allender, Jacob Gray, Saachi Mutreja, Harsha Tirumala, and Pengxiang Wang. Robustness for space-bounded statistical zero knowledge. In *Proceedings of the International Workshop on Randomization and Computation*, volume 275 of *LIPIcs*, pages 56:1–56:21, 2023. ECCC:TR22-138. 10

[AHT23] Eric Allender, Shuichi Hirahara, and Harsha Tirumala. Kolmogorov complexity characterizes statistical zero knowledge. In *Proceedings of the 14th Innovations in Theoretical Computer Science Conference*, volume 251, page 3, 2023. ECCC:TR22-127. 10

[AN02] Dorit Aharonov and Tomer Naveh. Quantum NP - a survey. *arXiv preprint quant-ph/0210077*, 2002. arXiv:quant-ph/0210077. 4

[AZLO16] Zeyuan Allen-Zhu, Yin Tat Lee, and Lorenzo Orecchia. Using optimization to obtain a width-independent, parallel, simpler, and faster positive SDP solver. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 1824–1831, 2016. arXiv:1507.02259. 35

[Bab85] László Babai. Trading group theory for randomness. In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pages 421–429, 1985. 1, 4

[BBC+95] Adriano Barenco, Charles H Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A. Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical review A*, 52(5):3457, 1995. arXiv:quant-ph/9503016. 25

[BCD+89] Allan Borodin, Stephen A. Cook, Patrick W. Dymond, Walter L. Ruzzo, and Martin Tompa. Two applications of inductive counting for complementation problems. *SIAM Journal on computing*, 18(3):559–578, 1989. Preliminary version in *SCT 1988*. 14

[BEM+23] John Bostanci, Yuval Efron, Tony Metger, Alexander Poremba, Luowen Qian, and Henry Yuen. Unitary complexity and the Uhlmann transformation problem, 2023. arXiv:2306.13073. 10

[BSW11] Salman Beigi, Peter Shor, and John Watrous. Quantum interactive proofs with short messages. *Theory of Computing*, 7(1):101–117, 2011. arXiv:1004.0411. 10

[CDGH24] Graham Cormode, Marcel de Sena Dall'Agnol, Tom Gur, and Chris Hickey. Streaming zero-knowledge proofs. In *Proceedings of the 39th Computational Complexity Conference*, volume 300 of *LIPIcs*, pages 2:1–2:66. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. arXiv:2301.02161. 10

[CL95] Anne Condon and Richard Ladner. Interactive proof systems with polynomially bounded strategies. *Journal of Computer and System Sciences*, 50(3):506–518, 1995. Preliminary version in *SCT 1992*. 2, 3, 7, 14, 16, 26, 27

[Con91] Anne Condon. Space-bounded probabilistic game automata. *Journal of the ACM*, 38(2):472–494, 1991. Preliminary version in *SCT 1988*. 2

[Con92] Anne Condon. The complexity of space boundes interactive proof systems. In *Complexity Theory: Current Research*, pages 147–189, 1992. 2, 26

[CR23] Joshua Cook and Ron D. Rothblum. Efficient interactive proofs for non-deterministic bounded space. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2023)*, volume 275 of *LIPIcs*, pages 47:1–47:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. ECCC:TR23-097. 2, 26

[DGRV11] Zeev Dvir, Dan Gutfreund, Guy N. Rothblum, and Salil P. Vadhan. On approximating the entropy of polynomial mappings. In *Proceedings of Second Symposium on Innovations in Computer Science*, pages 460–475, 2011. 10

[DLG24] Hugo Delavenne and François Le Gall. Quantum state synthesis: Relation with decision complexity classes and impossibility of synthesis error reduction. *Quantum Information and Computation*, 24(9-10):754–765, 2024. 10

[DLGLM23] Hugo Delavenne, François Le Gall, Yupan Liu, and Masayuki Miyamoto. Quantum Merlin-Arthur proof systems for synthesizing quantum states. *arXiv preprint arXiv:2303.01877*, 2023. arXiv:2303.01877. 10

[DS92] Cynthia Dwork and Larry Stockmeyer. Finite state verifiers I: The power of interaction. *Journal of the ACM*, 39(4):800–828, 1992. Preliminary version in *FOCS 1989*. 2

[FKL+16] Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. Space-efficient error reduction for unitary quantum computations. In *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming*, volume 55 of *LIPIcs*, page 14, 2016. arXiv:1604.08192. 2, 9, 15, 24, 41

[FL93] Lance Fortnow and Carsten Lund. Interactive proof systems and alternating time—space complexity. *Theoretical Computer Science*, 113(1):55–73, 1993. Preliminary version in *STACS 1991*. 2, 26

[FL18] Bill Fefferman and Cedric Yen-Yu Lin. A complete characterization of unitary quantum space. In *Proceedings of the 9th Innovations in Theoretical Computer Science Conference*, volume 94 of *LIPIcs*, page 4, 2018. arXiv:1604.01384. 1, 12

[For89] Lance J. Fortnow. *Complexity-Theoretic Aspects of Interactive Proof Systems*. PhD thesis, Massachusetts Institute of Technology, 1989. 2, 8, 10, 15, 26, 28

[FPLGN21] Pierre Fraigniaud, Ami Paz, François Le Gall, and Harumichi Nishimura. Distributed quantum proofs for replicated data. In *Proceedings of the 12th Innovations in Theoretical Computer Science Conference*, volume 185 of *LIPIcs*, pages 28:1–28:20, 2021. arXiv:2002.10018. 10

[FR21] Bill Fefferman and Zachary Remscrim. Eliminating intermediate measurements in space-bounded quantum computation. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1343–1356, 2021. arXiv:2006.03530. 1, 2, 3, 9, 12, 13, 15, 41

[GKR15] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: interactive proofs for muggles. *Journal of the ACM (JACM)*, 62(4):1–64, 2015. Preliminary version in *STOC 2008*. ECCC:TR17-108. 2, 26

[GLS93] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Algorithms and Combinatorics. Springer Berlin Heidelberg, 2nd edition, 1993. 35

[GM12] Bernd Gärtner and Jiří Matoušek. *Approximation Algorithms and Semidefinite Programming*. Springer Berlin Heidelberg, 1st edition, 2012. 5, 35

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989. Preliminary version in *STOC 1985*. 1

[Gol08] Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, 1st edition, 2008. 2

[GR22] Uma Girish and Ran Raz. Eliminating intermediate measurements using pseudorandom generators. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference*, volume 215 of *LIPIcs*, pages 76:1–76:18, 2022. arXiv:2106.11877. 1, 3, 13

[GR23] Sevag Gharibian and Dorian Rudolph. Quantum space, ground space traversal, and how to embed multi-prover interactive proofs into unentanglement. In *Proceedings of the 14th Innovations in Theoretical Computer Science*, volume 251 of *LIPIcs*, pages 53:1–53:23, 2023. arXiv:2206.05243. 2, 9

[GRZ21] Uma Girish, Ran Raz, and Wei Zhan. Quantum logspace algorithm for powering matrices with bounded norm. In *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming*, volume 198 of *LIPIcs*, pages 73:1–73:20, 2021. arXiv:2006.04880. 1, 3, 13

[GRZ24] Uma Girish, Ran Raz, and Wei Zhan. Quantum logspace computations are verifiable. In *Proceedings of the 2024 Symposium on Simplicity in Algorithms*, pages 144–150, 2024. arXiv:2307.11083. 10

[GS86] Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 59–68, 1986. 1, 4

[HKN24] Atsuya Hasegawa, Srijita Kundu, and Harumichi Nishimura. On the power of quantum distributed proofs. In *Proceedings of the 43rd ACM Symposium on Principles of Distributed Computing*, pages 220–230, 2024. arXiv:2403.14108. 10

[IKW12] Tsuyoshi Ito, Hirotada Kobayashi, and John Watrous. Quantum interactive proofs with weak error bounds. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 266–275, 2012. arXiv:1012.4427. 1

[INN+22] Sandy Irani, Anand Natarajan, Chinmay Nirkhe, Sujit Rao, and Henry Yuen. Quantum search-to-decision reductions and the state synthesis problem. In *Proceedings of the 37th Computational Complexity Conference*, pages 1–19, 2022. arXiv:2111.02999. 10

[JJUW11] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. *Journal of the ACM*, 58(6):1–27, 2011. Preliminary version in *STOC 2010*. arXiv:0907.4737. 1, 4, 7, 30, 35, 36

[JY11] Rahul Jain and Penghui Yao. A parallel approximation algorithm for positive semidefinite programming. In *Proceedings of the 52nd Annual Symposium on Foundations of Computer Science*, pages 463–471, 2011. arXiv:1104.2502. 35

[KKMV09] Julia Kempe, Hirotada Kobayashi, Keiji Matsumoto, and Thomas Vidick. Using entanglement in quantum multi-prover interactive proofs. *Computational Complexity*, 18:273–307, 2009. Preliminary version in *CCC 2008*. arXiv:0711.3715. 7, 30, 31, 32, 33

[KLGN19] Hirotada Kobayashi, François Le Gall, and Harumichi Nishimura. Generalized quantum Arthur–Merlin games. *SIAM Journal on Computing*, 48(3):865–902, 2019. Preliminary version in *CCC 2015*. arXiv:1312.4673. 1

[KOS18] Gillat Kol, Rotem Oshman, and Raghuvansh R Saxena. Interactive distributed proofs. In *Proceedings of the 37th ACM Symposium on Principles of Distributed Computing*, pages 255–264, 2018. 10

[KSV02] Alexei Y. Kitaev, Alexander H. Shen, and Mikhail N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. American Mathematical Society, 1st edition, 2002. 9

[KW00] Alexei Kitaev and John Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 608–617, 2000. 1, 2, 4, 6, 7, 14, 15, 23, 34

[LFKN92] Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *Journal of the ACM*, 39(4):859–868, 1992. Preliminary version in *FOCS 1990*. 1, 4

[LGLW23] François Le Gall, Yupan Liu, and Qisheng Wang. Space-bounded quantum state testing via space-efficient quantum singular value transformation. *arXiv preprint arXiv:2308.05079*, 2023. arXiv:2308.05079. 5, 9, 13, 36

[LGMN23] François Le Gall, Masayuki Miyamoto, and Harumichi Nishimura. Distributed quantum interactive proofs. In *Proceedings of the 40th International Symposium on Theoretical Aspects of Computer Science*, volume 254 of *LIPIcs*, pages 42:1–42:21, 2023. arXiv:2210.01390. 10

[Lip90] Richard J Lipton. Efficient checking of computations. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 207–215, 1990. 2, 8, 10, 14, 26

[LMW24] Alex Lombardi, Fermi Ma, and John Wright. A one-query lower bound for unitary synthesis and breaking quantum cryptography. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 979–990, 2024. arXiv:2310.08870. 10

[MW05] Chris Marriott and John Watrous. Quantum Arthur–Merlin games. *Computational Complexity*, 14(2):122–152, 2005. Preliminary version in *CCC 2004*. arXiv:cs/0506068. 1, 7, 24, 29, 35

[MY23] Tony Metger and Henry Yuen. stateQIP = statePSPACE. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science*, pages 1349–1356. IEEE, 2023. arXiv:2301.07730. 5, 10

[NC10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge university press, 10th anniversary edition, 2010. 3, 10, 11, 12

[NPY20] Moni Naor, Merav Parter, and Eylon Yogev. The power of distributed verifiers in interactive proofs. In *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1096–115. SIAM, 2020. arXiv:1812.10917. 10

[NS03] Ashwin Nayak and Peter Shor. Bit-commitment-based quantum coin flipping. *Physical Review A*, 67(1):012304, 2003. arXiv:quant-ph/0206123. 11

[NY09] Harumichi Nishimura and Tomoyuki Yamakami. An application of quantum finite automata to interactive proof systems. *Journal of Computer and System Sciences*, 75(4):255–269, 2009. Preliminary version in *CIAA 2004*. arXiv:quant-ph/0410040. 2

[NY15] Harumichi Nishimura and Tomoyuki Yamakami. Interactive proofs with quantum finite automata. *Theoretical Computer Science*, 568:1–18, 2015. arXiv:1401.2929. 2

[Per12] Attila Pereszlényi. On quantum interactive proofs with short messages. *Chicago Journal of Theoretical Computer Science*, 2012(9):1–10, 2012. arXiv:1109.0964. 10

[Ros24] Gregory Rosenthal. Efficient quantum state synthesis with one query. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2508–2534, 2024. arXiv:2306.01723. 10

[RY22] Gregory Rosenthal and Henry Yuen. Interactive proofs for synthesizing quantum states and unitaries. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference*, volume 215, pages 112:1–112:4, 2022. arXiv:2108.07192. 10

[Sha92] Adi Shamir. IP = PSPACE. *Journal of the ACM*, 39(4):869–877, 1992. Preliminary version in *FOCS 1990*. 1, 4

[SR01] Robert W Spekkens and Terry Rudolph. Degrees of concealment and bindingness in quantum bit commitment protocols. *Physical Review A*, 65(1):012310, 2001. arXiv:quant-ph/0106019. 11

[Sud78] Ivan Hal Sudborough. On the tape complexity of deterministic context-free languages. *Journal of the ACM*, 25(3):405–414, 1978. Preliminary version in *STOC 1976*. 14

[TS13] Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 881–890, 2013. 1

[Vad99] Salil P Vadhan. *A study of statistical zero-knowledge proofs*. PhD thesis, Massachusetts Institute of Technology, 1999. 5

[Ven91] Hari Venkateswaran. Properties that characterize LOGCFL. *Journal of Computer and System Sciences*, 43(2):380–404, 1991. Preliminary version in *STOC 1987*. 4, 14

[VW16] Thomas Vidick and John Watrous. Quantum proofs. *Foundations and Trends® in Theoretical Computer Science*, 11(1-2):1–215, 2016. arXiv:1610.01664. 5, 6, 12, 17, 19, 23, 24, 31

[Wat99] John Watrous. Space-bounded quantum complexity. *Journal of Computer and System Sciences*, 59(2):281–326, 1999. Preliminary version in *CCC 1998*. 1, 13, 41

[Wat02] John Watrous. Limits on the power of quantum statistical zero-knowledge. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 459–468, 2002. arXiv:quant-ph/0202111. 5, 9, 13, 15, 36, 37, 38, 39

[Wat03a] John Watrous. On the complexity of simulating space-bounded quantum computations. *Computational Complexity*, 12:48–84, 2003. Preliminary version in *FOCS 1999*. arXiv:cs/9911008. 1, 4, 13

[Wat03b] John Watrous. PSPACE has constant-round quantum interactive proof systems. *Theoretical Computer Science*, 292(3):575–588, 2003. Preliminary version in *FOCS 1999*. arXiv:cs/9901015. 1, 2, 4, 14

[Wat09a] John Watrous. Quantum computational complexity. *Encyclopedia of Complexity and Systems Science*, pages 7174–7201, 2009. arXiv:0804.3401. 10

[Wat09b] John Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009. Preliminary version in *STOC 2006*. arXiv:quant-ph/0511020. 5

[Wat16] John Watrous. Semidefinite programs for interactive proofs (Tutorial at the 19th Conference on Quantum Information Processing, QIP 2016). https://qipconference.org/2016/qip-sdp-handout.pdf, 2016. Accessed: 2024-09-18. 5, 19, 34

[Wil13] Mark M Wilde. *Quantum Information Theory*. Cambridge University Press, 1st edition, 2013. 11

[dW19] Ronald de Wolf. Quantum computing: Lecture notes, 2019. arXiv:1907.09415. 10

[Yak13] Abuzer Yakaryılmaz. Public qubits versus private coins. In *Proceedings of Workshop on Quantum and Classical Complexity*, pages 45–60, 2013. ECCC:TR12-130. 2

[Zha24] Mark Zhandry. The space-time cost of purifying quantum computations. In *Proceedings of the 15th Innovations in Theoretical Computer Science Conference*, volume 287 of *LIPIcs*, pages 102:1–102:22, 2024. arXiv:2401.07974. 1

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il