Distributed Merlin-Arthur Synthesis of Quantum States and Its Applications

François Le Gall
Graduate School of Mathematics, Nagoya University, Japan

Masayuki Miyamoto
Graduate School of Mathematics, Nagoya University, Japan

Harumichi Nishimura
Graduate School of Informatics, Nagoya University, Japan

Abstract

The generation and verification of quantum states are fundamental tasks for quantum information processing that have recently been investigated by Irani, Natarajan, Nirkhe, Rao and Yuen [CCC 2022], Rosenthal and Yuen [ITCS 2022], Metger and Yuen [QIP 2023] under the term *state synthesis*. This paper studies this concept from the viewpoint of quantum distributed computing, and especially distributed quantum Merlin-Arthur (dQMA) protocols. We first introduce a novel task, on a line, called state generation with distributed inputs (SGDI). In this task, the goal is to generate the quantum state $U|\psi\rangle$ at the rightmost node of the line, where $|\psi\rangle$ is a quantum state given at the leftmost node and U is a unitary matrix whose description is distributed over the nodes of the line. We give a dQMA protocol for SGDI and utilize this protocol to construct a dQMA protocol for the Set Equality problem studied by Naor, Parter and Yogev [SODA 2020], and complement our protocol by showing classical lower bounds for this problem. Our second contribution is a dQMA protocol, based on a recent work by Zhu and Hayashi [Physical Review A, 2019], to create EPR-pairs between adjacent nodes of a network without quantum communication. As an application of this dQMA protocol, we prove a general result showing how to convert any dQMA protocol on an arbitrary network into another dQMA protocol where the verification stage does not require any quantum communication.

2012 ACM Subject Classification Theory of computation → Distributed algorithms; Theory of computation → Quantum computation theory

Keywords and phrases distributed quantum Merlin-Arthur, distributed verification, quantum computation

Digital Object Identifier 10.4230/LIPIcs.MFCS.2023.63

Related Version *Full Version*: https://arxiv.org/abs/2210.01389

Funding FLG was supported by the JSPS KAKENHI grants JP16H01705, JP19H04066, JP20H00579, JP20H04139, JP20H05966, JP21H04879 and by the MEXT Q-LEAP grants JPMXS0118067394 and JPMXS0120319794. MM would like to take this opportunity to thank the "Nagoya University Interdisciplinary Frontier Fellowship" supported by Nagoya University and JST, the establishment of university fellowships towards the creation of science technology innovation, Grant Number JPMJFS212. HN was supported by the JSPS KAKENHI grants JP19H04066, JP20H05966, JP21H04879, JP22H00522 and by the MEXT Q-LEAP grants JPMXS0120319794.

1 Introduction

While quantum computational complexity has so far mostly investigated the complexity of classical problems (e.g., computing Boolean functions) in the quantum setting, recent works [1, 16, 20, 24, 29, 35] have started investigating the complexity of *quantum* problems (e.g., generating quantum states). For instance, Ji, Liu and Song [20] and Kretschmer [24]

have investigated the concept of quantum pseudorandom states from complexity-theoretic and cryptographic perspectives. Irani, Natarajan, Nirkhe, Rao, and Yuen [16] have made in-depth investigations of the complexity of the *state synthesis problem* in a setting first introduced by Aaronson [1] where the goal is to generate a quantum state by making queries to a classical oracle encoding the state. Rosenthal and Yuen [35] and Metger and Yuen [29] have considered interactive proofs for synthesizing quantum states (and also for implementing unitaries). Here the main goal is to generate complicated quantum states (e.g., quantum states described by an exponential-size generating quantum circuit) efficiently with the help of an all-powerful but untrusted prover. Note that in settings where an all-powerful prover is present, the task of quantum state synthesis is closely related to the task of quantum state verification (since the prover can simply send the quantum state that needs to be synthesized).

In this paper, we investigate the task of state generation and verification in the setting of quantum distributed computing. Quantum distributed computing is a fairly recent research topic: despite early investigations in the 2000s and the 2010s [3, 8, 9, 13, 36], it is only in the past five years that significant advances have been done in understanding the power of quantum distributed algorithms [2, 10, 17, 18, 25, 27, 37]. Fraigniaud, Le Gall, Nishimura, and Paz [10], in particular, have investigated the power of distributed quantum proofs in distributed computing, which is the natural quantum version of the concept of distributed classical proofs (also called locally-checkable proofs [14] or proof-labeling schemes [23]): each node of the network receives, additionally to its input, a quantum state (called a quantum proof) from an all-powerful but untrusted party called the prover. The main result from [10] shows that there exist classical problems that can be solved by quantum protocols using quantum proofs of length exponentially smaller than in the classical case.

We present two main results about state generation and verification in the setting where an all-powerful but untrusted prover helps the nodes in a non-interactive way, and apply these results to design new quantum protocols for concrete problems studied recently in [10, 33].

1.1 First result and applications: State Generation with Distributed Inputs

One of the main conceptual contributions of this paper is introducing the following problem: In a network of $r+1$ nodes $v_0, v_1, \ldots, v_r$, node v_0 is given as input an n-qubit quantum state $|\psi\rangle$. The goal is to generate the quantum state $U|\psi\rangle$ at node v_r, where U is a unitary matrix whose description is distributed over the nodes of the network. For concreteness, in this paper we focus on the case where the network is a path of length r and the nodes v_0, v_r are both extremities of the path.[1]

Here is the precise description of the problem. The parties $v_0, v_1, \ldots, v_r$ are the nodes of a line graph of length r: the left-end extremity is v_0, the right-end extremity is v_r, and nodes v_j and v_{j+1} are connected for $j = 0, 1, \ldots, r-1$. Node v_0 receives as input the classical description of an n-qubit state $|\psi\rangle$, as a 2^n-dimensional vector.[2] The other nodes v_j for $j = 1, 2, \ldots, r$ receive as input the description of an n-qubit unitary transformation: each node v_j receives the description of a unitary transformation U_j acting on n qubits. In this setting, the aim is to generate the quantum state

[1] In distributed computing it is standard to first investigate the complexity of computational problems on simple network topologies such as a path or a ring. A solution on the path can often be extended to networks of more complex topology, or be used as a building block for solving problems on network of arbitrary topology.

[2] Our protocol actually only requires v_0 to be able to generate many copies of $|\psi\rangle$, and thus also works when the input is a description of a quantum circuit generating $|\psi\rangle$, or even a black box generating $|\psi\rangle$.

$$|\varphi_r\rangle := U_r \cdots U_1 |\psi\rangle$$

at the right-end extremity v_r. We call this problem n-qubit *State Generation with Distributed Inputs* on the line of length r (n-qubit SGDI_r). Without a prover, this problem is clearly not solvable in less than r rounds of communications between neighbors (this can be seen easily by considering the case where $U_1 = \cdots = U_r = I$).

We consider the setting where a prover (an all-powerful but untrusted party) helps the nodes in a non-interactive way: at the very beginning of the protocol the prover sends to node v_j a quantum state ρ_j of at most s_c qubits, for each $j \in \{0, 1, \ldots, r\}$. Here s_c is called the certificate size of the protocol and the state ρ_j is called the certificate to v_j. The nodes then run a one-round[3] distributed quantum algorithm (called the verification algorithm). More precisely, the nodes first perform one round of (synchronous) communication: each node sends one quantum message of at most s_m qubits to its neighbors (s_m is called the message size of the protocol). Each node then decides to either accept or reject. Such protocols, which have been introduced and studied in [10], are called distributed Quantum Merlin-Arthur (dQMA) protocols (see Section 2 for details). Additionally, when considering dQMA protocols for n-qubit SGDI_r, we add the requirement that node v_r outputs an n-qubit quantum state at the end of the protocol.

Here is our main result:

▶ **Theorem 1.** *For any constant $\varepsilon > 0$, there exists a dQMA protocol for n-qubit SGDI_r with certificate size $O(n^2 r^5)$ and message size $O(nr^2)$ satisfying the following: (***completeness***) There are certificates $\rho_0, \ldots, \rho_r$ such that all the nodes accept and node v_r outputs $|\varphi_r\rangle$ with probability 1; (***soundness***) If all the nodes accept with probability at least ε, then the output state ρ of node v_r satisfies $\langle\varphi_r|\rho|\varphi_r\rangle \geq 1 - \varepsilon$.*

The protocol of Theorem 1 is a dQMA protocol with perfect completeness and soundness ε. Indeed, when receiving appropriate certificates from the prover, all nodes accept with probability 1 and node v_r outputs the state $|\varphi_r\rangle$. On the other hand, if the state ρ is far from $|\varphi_r\rangle$, the soundness condition guarantees that for any certificates $\rho_0, \ldots, \rho_r$ received from the prover (including the case of entangled certificates), the probability that at least one node rejects is at least $1 - \varepsilon$ (remember that the quantity $\langle\varphi_r|\rho|\varphi_r\rangle$ represents the square root of the fidelity between $|\varphi_r\rangle\langle\varphi_r|$ and ρ).

As an application of Theorem 1, we construct a quantum protocol for a concrete computational task called Set Equality, which was introduced in Ref. [33]. Here is the formal definition over a network of arbitrary topology (represented by an arbitrary graph $G = (V, E)$).

▶ **Definition 1** ($\mathsf{SetEquality}_{\ell,U}$ [33])**.** *Let ℓ be a positive integer and U be a finite set. Each node u of a graph $G = (V, E)$ holds two lists of ℓ elements $(a_{u,1}, \ldots, a_{u,\ell})$ and $(b_{u,1}, \ldots, b_{u,\ell})$ as input, where $a_{u,i}, b_{u,i} \in U$ for all $i \in \{1, 2, \ldots, \ell\}$. Define $A = \{a_{u,i} \mid u \in V,\ i \in \{1, 2, \ldots, \ell\}\}$ and $B = \{b_{u,i} \mid u \in V,\ i \in \{1, 2, \ldots, \ell\}\}$. The output of $\mathsf{SetEquality}_{\ell,U}$ is 1 (yes), if $A = B$ as multisets and 0 (no) otherwise.*

Using Theorem 1 we obtain the following result:

[3] As in almost all prior works on (classical or quantum) distributed proofs, in this paper we consider only one-round verification algorithms.

▶ **Theorem 2.** *For any small enough constant $\varepsilon > 0$, there exists a dQMA protocol for* $\mathsf{SetEquality}_{\ell,U}$ *on the line graph of length r with completeness $1-\varepsilon$ and soundness ε that has certificate size $O(r^5 \log^2(\ell r) \log^2 |U|)$ and message size $O(r^2 \log(\ell r) \log |U|)$.*

While Ref. [33] considered the special case of $\mathsf{SetEquality}_{\ell,U}$ and showed efficient distributed *interactive* protocols with small certificate and message size (see Section 1.4), no (nontrivial) classical dMA protocol (or lower bound) is known before this paper to our best knowledge. We complement the result in Theorem 2 by showing classical lower bounds and upper bounds of distributed Merlin-Arthur (dMA) protocols for $\mathsf{SetEquality}_{\ell,U}$.

▶ **Theorem 3.** *For any dMA protocol for* $\mathsf{SetEquality}_{\ell,U}$ *on a line graph of length r with certificate size s_c, completeness $3/4$, and soundness $1/4$,*

- *if $|U| < \ell$, then $s_c = \Omega(|U| \log(\ell/|U|))$;*
- *if $|U| = \Omega(\ell)$, then $s_c = \Omega(\ell)$;*
- *if $|U| = \Omega(r\ell)$, then $s_c = \Omega(r\ell)$.*

▶ **Theorem 4.** *There exists a dMA protocol for* $\mathsf{SetEquality}_{\ell,U}$ *on a line graph of length r with completeness 1 and soundness 0 whose certificate size and message size are both $O(\min\{r\ell \log |U|, |U| \log(r\ell)\})$.*

Although the dependence in r is worse than in the classical dMA protocol of Theorem 4, the dependence of the dQMA protocol of Theorem 2 in ℓ (the number of elements each node receives) and $|U|$ (the size of the universal set) are polylogarithmic. On the other hand, in classical case, we have linear lower bounds with respect to ℓ and $|U|$ as in Theorem 3. Therefore Theorem 2 gives a significant improvement for sufficiently large ℓ and $|U|$. This assumption about the input parameters seems reasonable when considering applications similar to those of the dQMA protocol for the equality problem proposed in Ref. [10]. Note that our bounds of classical certificate size in Theorem 3 and Theorem 4 are tight up to $\operatorname{poly}\log(\ell, |U|, r)$ factors when $|U| < \ell$ or $|U| = \Omega(r\ell)$.

1.2 Second result and applications: EPR-pairs generation and LOCC dQMA protocols

Our second contribution is a protocol, based on a recent work by Zhu and Hayashi [41], to create EPR-pairs between adjacent nodes of a network without quantum communication in the same setting as above, where a prover helps the nodes in a non-interactive way. As an application of this protocol, we prove a general result showing how to convert any dQMA protocol on an arbitrary network into another dQMA protocol where the verification algorithm uses only classical communication (instead as quantum communication, as allowed in the definition of dQMA protocols and used in all dQMA protocols of Ref. [10] and Theorems 1 and 2 above).

More precisely, we say a dQMA protocol is an LOCC (Local Operation and Classical Communication) dQMA protocol if the verification algorithm can be implemented only by local operations at each node and classical communication between neighboring nodes (i.e., no quantum communication is allowed). Our protocol for generating EPR-pairs enables us to show the following theorem:

▶ **Theorem 5.** *For any constants p_c and p_s such that $0 \leq p_s < p_c \leq 1$, let $\mathcal{P}$ be a dQMA protocol for some problem on a network G with completeness p_c, soundness p_s, certificate size $s_c^{\mathcal{P}}$ and message size $s_m^{\mathcal{P}}$. For any small enough constant $\gamma > 0$, there exists an LOCC dQMA protocol $\mathcal{P}'$ for the same problem on G with completeness p_c, soundness $p_s + \gamma$, certificate size $s_c^{\mathcal{P}} + O(d_{\max} s_m^{\mathcal{P}} s_{tm}^{\mathcal{P}})$, and message size $O(s_m^{\mathcal{P}} s_{tm}^{\mathcal{P}})$, where $d_{\max}$ is the maximum degree of G, and $s_{tm}^{\mathcal{P}}$ is the total number of qubits sent in the verification stage of $\mathcal{P}$.*

As an application of Theorem 5, we consider the equality problem studied in Ref. [10]. In this problem, denoted EQ_n^t, a collection of n-bit strings $x_1, x_2, \ldots, x_t$ is given as input to t specific nodes $u_1, u_2, \ldots, u_t$ (called terminals) of an arbitrary network $G = (V, E)$ as follows: node u_i receives x_i, for $i \in \{1, 2, \ldots, t\}$. The goal is to check whether the t strings are equal, i.e., whether $x_1 = \cdots = x_t$. By applying Theorem 5 to the main result in Ref. [10] (a dQMA protocol for EQ_n^t with certificate size $O(tr^2 \log n)$ and message size $O(tr^2 \log(n + r))$), we obtain the following corollary:

▶ **Corollary 1.** *For any small enough constant $\varepsilon > 0$, there is an LOCC dQMA protocol for EQ_n^t with completeness 1, soundness ε, certificate size $O(d_{\max}|V|t^2r^4 \log^2(n + r))$ and messages size $O(|V|t^2r^4 \log^2(n + r))$, where r is the radius of the set of the t terminals and $|V|$ is the number of nodes of the network $G = (V, E)$.*

We can also apply Theorem 5 to the dQMA protocol of Theorem 2, leading to the following corollary:

▶ **Corollary 2.** *For any small enough constant $\varepsilon > 0$, there is an LOCC dQMA protocol for $\mathsf{SetEquality}_{\ell,U}$ on the line graph of length r with completeness $1 - \varepsilon$, soundness ε, certificate size $O(r^5 \log^2(\ell r) \log^2 |U|)$ and messages size $O(r^5 \log^2(\ell r) \log^2 |U|)$.*

Note that these LOCC dQMA protocols still have good dependence in the main parameters we are interested in: the parameter n for EQ_n^t (for which the dependence is still exponentially better than any classical dMA protocols) and the parameters ℓ and $|U|$ for $\mathsf{SetEquality}_{\ell,U}$ (for which the dependence is still exponentially better than any classical dMA protocols, due to Theorem 3).

1.3 Overview of our proofs

To explain the proof idea of Theorem 1, we only consider the simplified case $U_1 = \cdots = U_r = I$. The general case can be proved similarly by a slightly more complicated analysis.

The dQMA protocol to prove Theorem 1 is based on the dQMA protocol on the line of length r by Fraigniaud et al. [10]. In the setting of Ref. [10], the left-end extremity v_0 has an n-bit string x, the right-end extremity v_r has an n-bit string y, and the other intermediate nodes have no input. The goal is to verify whether $x = y$. The dQMA protocol in Ref. [10] checks whether the fingerprint state $|\psi_0\rangle = |\psi_x\rangle$ [5] prepared by v_0 is equal to the fingerprint state $|\psi_r\rangle = |\psi_y\rangle$ prepared by v_r ($x = y$), or $|\psi_0\rangle$ is almost orthogonal to $|\psi_r\rangle$ ($x \neq y$). For this, node v_j ($2 \leq j \leq r - 1$) receives a subsystem whose reduced state is ρ_j as a certificate from the prover. At the verification stage, any node (except for v_r) chooses keeping its certificate by itself, or sending it to the right neighboring node with probability 1/2 to check if the reduced states of the two neighboring nodes, ρ_j and ρ_{j+1}, are close, which can be checked by the SWAP test [5]. If $x = y$, then the prover can send $|\psi_0\rangle$ ($= |\psi_r\rangle$) for every intermediate node to pass all the SWAP tests done at the verification stage, which means accept. Otherwise, the SWAP test done at some node rejects with a reasonable probability since $|\psi_x\rangle$ is very far from $|\psi_y\rangle$, and hence the distance between ρ_j and ρ_{j+1} should be far at some j.

Now the case that $U_1 = \cdots = U_r = I$ (which means that all nodes except v_0 have no input) in the setting of SGDI_r (then the goal state $|\varphi_r\rangle$ at v_r is the same as the state $|\psi\rangle$ of v_0) is similar to the setting of Ref. [10], except that v_r also has no input. The difficulty is that v_r has no state that can be generated by itself, and thus the analysis of Ref. [10] cannot be used as it is.

To overcome this difficulty, we utilize an idea from the verification of graph states [15, 32], in particular, the idea by Morimae, Takeuchi, and Hayashi [32]. They used the following basic idea for their protocol in order to verify an arbitrary graph state $|G\rangle$ sent from the prover (or prepared by a malicious party): (i) the verifier receives $(m+k+1)$ subsystems, in which each subsystem ideally contains $|G\rangle$, from the prover; (ii) the verifier chooses m subsystems uniformly at random, and discards them; (iii) the verifier chooses one subsystem, and some test that $|G\rangle$ should pass (stabilizer test) is done for each of the remaining k subsystems; and (iv) if all the tests passed, the chosen subsystem in (iii) should be close to $|G\rangle$, which is proved by using a quantum de Finetti theorem with some measurement condition [28] (exponentially better in the dimension of the subsystem than the standard quantum de Finetti theorem [6]). Note that (ii) and (iii) are necessary since the assumption that the total system is permutation-invariant is needed to apply the quantum de Finetti theorem.

Our protocol applies the idea of Ref. [32] to the verification protocol of Ref. [10] explained above. Namely, the parties $v_1, v_2, \ldots, v_r$ first receives $(m+k+1)$ subsystems, where each subsystem ideally contains $|\psi\rangle^{\otimes r}$, sent from the prover. For k subsystems that are randomly chosen, we apply the verification protocol of Ref. [10]. Actually, we have a subtle problem with the corresponding steps of (ii) and (iii) in the idea of Ref. [32], since $v_0, v_1, \ldots, v_r$ do not have any shared randomness, and thus those steps cannot be implemented jointly. Fortunately, this problem can be overcome since the permutation-invariant property is satisfied by the random permutations of $(m+k+1)$ subsystems on *each* party.

The dQMA protocol for Theorem 2 is based on the distributed interactive protocol by Naor, Parter, and Yogev [33] using shared randomness[4]. In our setting (line of length r), the distributed interactive protocol of Ref. [33] is as follows with two polynomials $\alpha_j(x) := \prod_i (x - a_{j,i})$ and $\beta_j(x) := \prod_i (x - b_{j,i})$: with shared randomness s (taken from a large field), (i) v_0 prepares $A_0(s) := \alpha_0(s)$ and $B_0(s) := \beta_0(s)$; (ii) v_j $(j = 1, 2, \ldots, r)$ ideally receives $A_j(s) := \alpha_0(s) \cdots \alpha_j(s)$ and $B_j(s) := \beta_0(s) \cdots \beta_j(s)$ from the prover; (iii) $A_j(s) = \alpha_j(s) A_{j-1}(s)$ and $B_j(s) = \beta_j(s) B_{j-1}(s)$ are checked for consistency by communication from v_{j-1} to v_j. We can see that when $A = B$, $A_r(s) = B_r(s)$ for any s, and thus this protocol accepts with probability 1 by the ideal certificates from the prover, while when $A \neq B$, $A_r(s) \neq B_r(s)$ for most of s, and thus some node rejects with reasonable probability.

Actually, neither interaction nor shared randomness is available in our setting. Instead, we reduce the protocol by Naor et al. to SGDI_r with $|\psi\rangle = |\psi_A\rangle \otimes |\psi_B\rangle$ where $|\psi_A\rangle = \sum_s |s\rangle|\alpha_0(s)\rangle$, and $|\psi_B\rangle = \sum_s |s\rangle|\beta_0(s)\rangle$, and $U = U_{j,A} \otimes U_{j,B}$, where $U_{j,A}$ roughly[5] maps $|s\rangle|t\rangle$ to $|s\rangle|\alpha_j(s)t\rangle$ $(j = 1, 2, \ldots, r)$ and $U_{j,B}$ roughly maps $|s\rangle|t\rangle$ to $|s\rangle|\beta_j(s)t\rangle$ $(j = 1, 2, \ldots, r)$. Then, Theorem 1 guarantees that v_r receives $\sum_s |s\rangle|A_r(s)\rangle$ and $\sum_s |s\rangle|B_r(s)\rangle$ with high fidelity as long as every node accepts with at least the probability guaranteed by Theorem 1. The SWAP test between these at v_r checks if $A = B$ with high probability.

For the classical lower bound of $\mathsf{SetEquality}_{\ell,U}$ in Theorem 3, we utilize the lower bound for EQ^2_n of [10]. Ref. [10] showed that for any classical protocol for EQ^2_n on the line graph, at least one internal node requires a certificate of linear size. We show that EQ^2_n can be reduced to $\mathsf{SetEquality}_{\ell,U}$ in three cases depending on the size of U. Here we explain the simplest case: $|U| = \Omega(\ell)$. For a line graph with the left-end extremity v and the right-end extremity v', let $x = x_1 x_2 \cdots x_n$ be the input of EQ^2_n for v and $y = y_1 y_2 \cdots y_n$ be the input of EQ^2_n for v'. Then we consider an injection f from $\{0,1\}^n$ to the set of 3ℓ-bit strings with

[4] While there is no shared randomness in their setting, shared randomness can be simulated by two interactions between the prover and the verifier.

[5] We actually need some modifications for $U_{j,A}$ to be unitary.

Hamming weight ℓ such that the input list $(a_{v,1}, \ldots, a_{v,\ell})$ of $\mathsf{SetEquality}_{\ell,U}$ for v includes the j-th element of the universal set U for $|U| > 3\ell$ if and only if $f(x)_j = 1$, and the input list $(b_{v',1}, \ldots, b_{v',\ell})$ of $\mathsf{SetEquality}_{\ell,U}$ for v' includes the j-th element of the universal set U if and only if $f(y)_j = 1$. Now these two sets are identical if and only if $x = y$, which means a reduction from EQ_n^2 to $\mathsf{SetEquality}_{\ell,U}$ for $\ell = \Theta(n)$. We thus get a lower bound of $\Omega(\ell)$ from the $\Omega(n)$ lower bound of EQ_n^2 mentioned above.

The classical upper bound of $\mathsf{SetEquality}_{\ell,U}$ in Theorem 4 is fairly simple: the prover can send all of inputs A and B to each node to achieve the first upper bound $O(r\ell \log |U|)$. For the second upper bound $O(|U| \log(r\ell))$, the node v_i on the line graph $\{v_0, \ldots, v_r\}$ is given the information of inputs of $v_j, j \in \{0, \ldots, i-1\}$ as the certificate in the form of the number of each element of U in the corresponding inputs.

The basic proof idea of Theorem 5 is standard: we replace one qubit communicated between any two nodes u and v by two bits using quantum teleportation [4], assuming that they share an EPR pair $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ sent from the prover. The problem is that the prover may be malicious, and u and v should then verify that the pair sent from the prover is $|\Phi^+\rangle$. In order to obtain $|\Phi^+\rangle$ with high fidelity, we actually ask the prover to send $N+1$ copies of the EPR pairs. An honest prover will send the state $|\Phi^+\rangle^{\otimes(N+1)}$, but a malicious prover may naturally send an arbitrary state. Nodes u and v use N among the $N+1$ pairs for the verification. If the verification succeeds, they are guaranteed that the remaining pair has high fidelity with $|\Phi^+\rangle$.

This type of verification of $|\Phi^+\rangle$ in an adversarial scenario by the malicious prover was considered in a remarkable work by Zhu and Hayashi [41]. Extending the previous result [34] in a less adversarial scenario, they showed that by taking $N = O(\frac{1}{\varepsilon}\log(\frac{1}{\delta}))$, if the verification test succeeds with probability at least δ, the state σ of the last pair has a high fidelity with $|\Phi^+\rangle$ such that $\langle\Phi^+|\sigma|\Phi^+\rangle \geq 1-\varepsilon$. Furthermore, the measurements in their verification protocol (essentially the same as those in Ref. [34]) are local, namely, they do not need any entangled measurement between the two qubits of each pair.

Now the proof idea of Theorem 5 uses the verification protocol of Ref. [41] in our setting. To do so, we first observe that the amount of classical communication needed between u and v can be upper-bounded by $O(N)$ (which is the same as the certificate size from the prover), by rewriting the protocol of Ref. [41] with a slight modification in our setting. Then we replace the quantum bits sent among the nodes in the original dQMA protocol $\mathcal{P}$ by classical communication. However, it needs not only a single EPR pair but a lot of EPR pairs to be verified. Thus, we need further analysis to convert $\mathcal{P}$ into an LOCC dQMA protocol and to evaluate the message size of classical communication and the certificate size.

1.4 Related work

The concept of *distributed Merlin-Arthur protocols* (dMA), which is very similar to the concept of *randomized proof-labeling schemes* [12] was introduced by [11] as a randomized version of locally checkable proofs (LCPs). In a dMA protocol, as in LCPs, the prover assigns each node a short certificate. The nodes then perform a 1-round distributed algorithm, i.e., exchange messages with their neighbors through incident edges. The difference is that in dMA, this algorithm can be a randomized algorithm, instead of a deterministic algorithm as in LCPs. This randomization is helpful to reduce the size of certificates for some problems.

The recent paper [22] introduced the interactive extension of dMA, *distributed interactive proofs*, in which the prover and the verifier can perform more interaction. They showed that interaction is also useful to reduce the size of certificates. This concept has recently been explored in depth by several studies: distributed interactive proofs that utilize quantum certificates [26], the role of shared and private randomness [7, 30], and more efficient protocols

for concrete problems [19, 31, 33]. In particular, [33] introduced $\mathsf{SetEquality}_{\ell,U}$, which is one of the problems we study in this paper, and showed efficient interactive protocols for $\mathsf{SetEquality}_{\ell,U}$ when $\ell = |V|$ and $|U| = O(|V|)$ that require two interactions between the prover and the verifier with certificate size[6] $O(\log |V|)$, and five interactions between the prover and the verifier with certificate size $O(\log \log |V|)$.

The technique we used in this paper from Refs. [32, 41] belongs to a broad and hot topic called "state certification (state verification)" [21, 38]. One conceptual contribution of this paper is providing the first concrete example of the effective use of these techniques for quantum distributed verification.

2 dQMA protocols

We consider a decision problem on a connected graph (called the network) $G = (V, E)$, where t inputs $x_1, x_2, \ldots, x_t$ are assigned to t nodes $v_1, v_2, \ldots, v_t \in V$. We interpret the decision problem as a Boolean function f, where $f(x_1, x_2, \ldots, x_t) = 1$ is interpreted as "yes" and $f(x_1, x_2, \ldots, x_t) = 0$ is interpreted as "no".

The concept of distributed quantum Merlin-Arthur (dQMA) protocols on a graph $G = (V, E)$ is a quantum version of the concept of distributed Merlin-Arthur (dMA) protocols. The aim of a dMA protocol is to verify whether $f(x_1, x_2, \ldots, x_t) = 1$ or not. As briefly explained in Section 1.1, the nodes of G (which correspond to the verifier) first receive a message from a powerful but possibly malicious party (the prover). The nodes then enter a verification phase, in which they communicate together (but do not communicate with the prover anymore). The communication is possible only if two nodes are connected: each node can send one message to each of its neighbors. In the case of dQMA protocols, the only difference is that the message from the prover and the communication among the nodes may be quantum. Note that neither randomness nor entanglement are shared among the nodes in advance.

Formally, in a dQMA protocol $\mathcal{P}$ on $G = (V, E)$, each node $u \in V$ first receives a quantum register M_u from the prover. Then the nodes move to the verification stage, which consists of the following steps: (i) u applies a local quantum (or classical) operation on the composite system of M_u and its private register V_u; (ii) u sends a quantum (or classical) register M_{uv} to any neighboring node v, and (iii) u applies a local quantum (or classical) operation on M_u, V_u, and $\otimes_{v \in N(u)} \mathsf{M}_{vu}$, and either accepts or rejects (we call this the decision of u), where $N(u)$ denotes the set of nodes that are neighbors of u. When local operations at each node and communication among the nodes in the verification stage are classical, the dQMA protocol is called *LOCC (Local Operation and Classical Communication)*.

The two main complexity measures of $\mathcal{P}$ are the certificate size and the message size. The certificate size of $\mathcal{P}$, denoted as $s_c^{\mathcal{P}}$, is the maximum number of qubits that are sent to each node from the prover, that is, $s_c^{\mathcal{P}} := \max_{u \in V} |\mathsf{M}_u|$, where $|\mathsf{R}|$ denotes the number of qubits of R. The message size of $\mathcal{P}$, denoted as $s_m^{\mathcal{P}}$, is the maximum number of qubits sent on edges of G, namely, $s_m^{\mathcal{P}} := \max_{(u,v) \in E} (|\mathsf{M}_{uv}| + |\mathsf{M}_{vu}|)$.

A dQMA protocol $\mathcal{P}$ for a decision problem f on G with completeness p_c and soundness p_s is defined as a dQMA protocol satisfying the following two conditions:

(completeness) If $f(x_1, x_2, \ldots, x_t) = 1$, there exists some quantum state $|\chi\rangle$ on $\mathsf{M} := \otimes_{u \in V} \mathsf{M}_u$ such that $\Pr[\text{all nodes accept}] \geq p_c$;

(soundness) If $f(x_1, x_2, \ldots, x_t) = 0$, for any quantum state $|\chi\rangle$ on M, $\Pr[\text{all nodes accept}] \leq p_s$.

[6] For $\mathsf{SetEquality}_{\ell,U}$, the certificate size of their protocol can be written as $O(\log |U| + \log(\ell |V|))$.

In this paper, we consider the problem of generating a quantum state $|\varphi\rangle$ on a network $G = (V, E)$. In this problem, some initially specified nodes $w_1, \ldots, w_\kappa$ not only make their decisions (accept or reject) but also output the quantum state $|\varphi\rangle$ jointly (if they accept). In our specific problem, the n-qubit SGDI_r, all nodes of the line graph with nodes $v_0, v_1, \ldots, v_r$ have an input (v_0 has a classical description of $|\psi\rangle$ and v_j for $j = 1, 2, \ldots, r$ has a classical description of U_j), $|\varphi\rangle = |\varphi_r\rangle$ ($:= U_r \cdots U_1|\psi\rangle$), $\kappa = 1$, and $w_1 = v_r$.

In a dQMA protocol for the problem of generating $|\varphi\rangle$ on G, the completeness and soundness conditions are slightly different from the case of decision problems. For our purpose we actually only need to discuss perfect-completeness protocols. We say that the dQMA protocol has perfect completeness and (δ, ε)-soundness if the following completeness and soundness are satisfied:

(completeness) There exists a quantum state $|\chi\rangle$ on M such that

$$\Pr[\text{all nodes accept and } w_1, \ldots, w_\kappa \text{ output } |\varphi\rangle \text{ jointly}] = 1;$$

(soundness) If all nodes accept with probability at least δ, then the output $\tilde{\rho}$ of $w_1, \ldots, w_\kappa$ (under the condition that all nodes accept) satisfies

$$\langle\varphi|\tilde{\rho}|\varphi\rangle \geq 1 - \varepsilon.$$

The soundness condition is regarded as a kind of hypothesis testing (i.e., if the verifier's test passes with probability greater than a threshold, then the state would be close to the ideal one). A similar completeness-soundness condition is used for the interactive proofs for synthesizing quantum states [35].

3 dQMA Protocol for State Generation with Distributed Inputs

In this section we present our dQMA protocol for the n-qubit State Generation with Distributed Inputs over the line of length r (n-qubit SGDI_r) and prove Theorem 1.

3.1 dQMA protocol for SGDI

The following is our dQMA protocol for n-qubit SGDI_r.

Protocol $\mathcal{P}_{\mathsf{SGDI}}$: Let $k = 144cr^{2+\eta}$ and $m = 2cnk^2(r+1)^{1+\eta}$ for any constant $c > 0$ and any small constant $\eta \geq 0$.

1. v_0 prepares $(m+k+1)$ copies of $|\psi\rangle$ in n-qubit registers $\mathsf{R}_{0,j}$ $(j = 1, 2, \ldots, m+k+1)$.
2. The prover sends each v_l, where $l = 1, 2, \ldots, r$, $(m + k + 1)$ n-qubit registers $\mathsf{R}_{l,1}, \mathsf{R}_{l,2}, \ldots, \mathsf{R}_{l,m+k+1}$.
3. Each v_l $(l = 1, 2, \ldots, r)$ permutes the $(m+k+1)$ registers $\mathsf{R}_{l,1}, \mathsf{R}_{l,2}, \ldots, \mathsf{R}_{l,m+k+1}$ by a permutation π on $\{1, 2, \ldots, m + k + 1\}$ taken uniformly at random, and renames $\mathsf{R}_{l,j} := \mathsf{R}_{l,\pi(j)}$.
4. The parties $v_0, v_1, \ldots, v_r$ implement the following subprotocol $\mathcal{P}_{\mathsf{SGDIV}}$ (a modification of the verification steps in Ref. [10]) on registers $\mathsf{R}_{0,j}, \mathsf{R}_{1,j}, \ldots, \mathsf{R}_{r,j}$ for each $j = 2, 3, \ldots, k+1$ in order. If some party rejects for some j, the protocol rejects.
5. v_r outputs $\mathsf{R}_{r,1}$.

Protocol $\mathcal{P}_{\mathsf{SGDIV}}$: Assume that v_0 has $|\psi\rangle$ on n-qubit register R_0, and v_l $(l = 1, 2, \ldots, r)$ receives n-qubit register R_l.

1. For every $j = 0, 1, \ldots, r-1$, party v_j chooses a bit b_j uniformly at random, and sends its register R_j to the right neighbor v_{j+1} whenever $b_j = 0$.
2. For every $j = 1, 2, \ldots, r$, if v_j receives a register from the left neighbor v_{j-1}, and if $b_j = 1$, then v_j applies U_j on register R_{j-1}, and performs the SWAP test on the registers $(\mathsf{R}_{j-1}, \mathsf{R}_j)$, and accepts or rejects accordingly; Otherwise, v_j accepts.

We can show the following theorem, which induces Theorem 1 by a special case with $\eta = 0$.

▶ **Theorem 6.** *Protocol $\mathcal{P}_{\mathsf{SGDI}}$ has perfect completeness and $(\frac{1}{(cr^\eta)^{1/4}}, \frac{1}{(cr^\eta)^{1/4}})$-soundness. The certificate size of $\mathcal{P}_{\mathsf{SGDI}}$ is $O(n^2 r^{5+3\eta})$ and the message size is $O(nr^{2+\eta})$.*

3.2 Proof of Theorem 6

We can see that the certificate size of $\mathcal{P}_{\mathsf{SGDI}}$ is $(m + k + 1)n = O(n^2 r^{5+3\eta})$ from step 2 of $\mathcal{P}_{\mathsf{SGDI}}$. Since $\mathcal{P}_{\mathsf{SGDI}}$ implements $\mathcal{P}_{\mathsf{SGDIV}}$ $(k + 1)$ times, and the message size of $\mathcal{P}_{\mathsf{SGDI}}$ is $O(n)$, the message size of $\mathcal{P}_{\mathsf{SGDI}}$ is $O(nk) = O(nr^{2+\eta})$.

The completeness clearly holds: since the prover honestly sends

$$|\varphi_l\rangle := U_l \cdots U_1 |\psi\rangle$$

as the content of $\mathsf{R}_{l,j}$ for each $j \in \{1, 2, \ldots, m + k + 1\}$ and then all the SWAP tests in $\mathcal{P}_{\mathsf{SGDIV}}$ accept with probability 1.

The proof of the soundness can be found in the full version.

4 Application: dQMA Protocol for Set Equality

In this section we prove Theorem 2 by constructing a protocol for $\mathsf{SetEquality}_{\ell,U}$ based on the protocol for SGDI_r developed in Section 3.

Proof of Theorem 2. We consider $\mathsf{SetEquality}_{\ell,U}$ (Definition 1) for the line graph of length r with nodes $v_0, v_1, \ldots, v_r$, where v_j has $a_{j,1}, \ldots, a_{j,\ell}$ and $b_{j,1}, \ldots, b_{j,\ell}$. Let $\alpha_j(s) := \prod_{i \in \{1,2,\ldots,\ell\}} (s - a_{j,i})$ and $\beta_j(s) := \prod_{i \in \{1,2,\ldots,\ell\}} (s - b_{j,i})$ for each $j \in \{0, 1, \ldots, r\}$. We identify $a_{u,i}, b_{u,i}$ as elements in a finite field $\mathbb{F}$ with size $|\mathbb{F}| \geq \tilde{c}\ell(r+1)2^{\log |U|}$ for some (sufficiently large) constant $\tilde{c} > 0$. Our goal is the same as of the interactive protocol of [33] – the node v_r checks if two polynomials $p_A(s) := \prod_{j \in \{0,1,\ldots,r\}} \alpha_j(s)$ and $p_B(s) := \prod_{j \in \{0,1,\ldots,r\}} \beta_j(s)$ take the same value for uniform randomly chosen $s \in \mathbb{F}$, where s is distributed by the interaction. In order to implement this idea in a non-interactive way, we utilize the framework of SGDI as follows: Let

$$|\psi\rangle := \frac{1}{\sqrt{|\mathbb{F}|}} \sum_{s \in \mathbb{F}} |s\rangle |\alpha_0(s)\rangle$$

be the initial state that the node v_0 can locally produce. Each node $v_j, j \in \{1, 2, \ldots, r\}$ has a unitary transformation U_j which maps $|s\rangle|\alpha_{j-1}(s)\rangle$ to $|s\rangle|\alpha_j(s)\rangle$ (to be precise, we also need to deal with the case where $\alpha_j(s)$ is a zero polynomial). Using the protocol for SGDI in Theorem 1, the node v_r outputs the state $|\varphi_r\rangle = U_r \cdots U_1 |\psi\rangle$, which is the uniform

superposition of $|s\rangle|p_A(s)\rangle$ over all $s \in \mathbb{F}$. Similarly, v_r also outputs the uniform superposition of $|s\rangle|p_B(s)\rangle$ over all $s \in \mathbb{F}$. Finally, v_r does the SWAP test for these states, which accepts with high probability if and only if two polynomials are identical (i.e., $A = B$ as multisets) since the two polynomials $p_A(s)$ and $p_B(s)$ take different values for most $s \in \mathbb{F}$ if $A \neq B$. In the full version we complete the proof by describing the details of the protocol (which we denote $\mathcal{P}_{\mathsf{seteq}}$) and analyzing it rigorously. ◀

4.1 Classical bounds for $\mathsf{SetEquality}_{\ell,U}$

Here we prove the classical lower bounds shown in Theorem 3. The proof of Theorem 4 is deferred to the full version.

Proof of Theorem 3. In order to prove our lower bounds for $\mathsf{SetEquality}_{\ell,U}$, we utilize reductions from EQ_n^2 to $\mathsf{SetEquality}_{\ell,U}$, then apply the following lower bound of EQ_n^2 that appears in [10].

▶ **Lemma 1** (Theorem 9 of [10]). *Let $r \geq 3$ be a positive integer. Consider an instance of EQ_n^2 where the two nodes v_0 and v_r on a line graph $v_0, \ldots, v_r$ are provided with inputs $x \in \{0,1\}^n$ and $y \in \{0,1\}^n$. Then, for any dMA protocol that solves EQ_n^2 for this instance with completeness $1-p$ and soundness $1-2p-\varepsilon$ for any $p, \varepsilon > 0$, there exists $i \in \{1, \ldots, r-1\}$ such that the certificate size of v_i is $\Omega(n)$.*

We show three different reductions depending on the size of $|U|$. Due to space constraints, we only show the simplest case: $|U| = \Omega(\ell)$. The reductions for the other two cases are deferred to the full version.

Let $\mathcal{P}$ be a dMA protocol for $\mathsf{SetEquality}_{\ell,U}$ with the certificate size s_c which appears in the statement of the theorem. We consider the following instance of EQ_n^2 on a line graph $v_0, \ldots, v_r$ of length $r \geq 3$: the node v_0 is provided with x, and the node v_r is provided with y. Let ℓ be the minimum integer satisfying $\binom{3\ell}{\ell} \geq 2^n$. Since $\binom{3\ell}{\ell} = 2^{3\ell H(1/3) - O(\log \ell)}$ where $H(\cdot)$ is the binary entropy function, we have $\ell = \Theta(n)$. Let $S = \{s \in \{0,1\}^{3\ell} : |s| = \ell\}$ be the set of 3ℓ-bit strings so that $|S| = \binom{3\ell}{\ell}$. We arbitrarily choose one injection $f : \{0,1\}^n \to S$ (this kind of injection exists since we have $|S| \geq 2^n$). The network constructs the following instance of $\mathsf{SetEquality}_{\ell,U}$ for the universal set $U = \{0, 1, 2, \ldots, 3\ell\}$ without communication:

- The inputs x and y are converted to $f(x), f(y) \in S$. Let $X = \{i : f(x)_i = 1\}$ and $Y = \{i : f(y)_i = 1\}$ be two sets of ℓ elements from the universal set $\{1, 2, \ldots, 3\ell\}$. X and Y are regarded as the inputs $(a_{v_0,1}, \ldots, a_{v_0,\ell})$ and $(b_{v_r,1}, \ldots, b_{v_r,\ell})$ of $\mathsf{SetEquality}_{\ell,U}$. Furthermore, we set $(b_{v_0,1}, \ldots, b_{v_0,\ell}) = (a_{v_r,1}, \ldots, a_{v_r,\ell}) = (0, \ldots, 0)$.
- The inputs to each internal node $v_1, \ldots, v_{r-1}$ are set to $(0, \ldots, 0), (0, \ldots, 0)$.

Now the set A and B of this instance of $\mathsf{SetEquality}_{\ell,U}$ are identical as multisets if and only if $f(x) = f(y)$. Since f is an injection, we have $f(x) = f(y) \Leftrightarrow x = y$ and thus the output of $\mathsf{SetEquality}_{\ell,U}$ on this instance is identical to that of EQ_n^2 on the input x and y. Now we can use the protocol $\mathcal{P}$ to solve EQ_n^2 for $n = \Theta(\ell)$. Thus by Lemma 1, we have $s_c = \Omega(\ell)$. ◀

5 Conversion of dQMA protocols into LOCC dQMA protocols

In this section we show how to create an EPR pair $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ between two parties without quantum communication in the setting where a prover helps the nodes in a non-interactive way. Our protocol is based on the verification protocol of the EPR pair in the adversarial setting proposed by Zhu and Hayashi [41] (see also [39, 40]), who showed that a verifier V can check whether a two-qubit state sent from a (possibly malicious) prover is $|\Phi^+\rangle$.

The following is the verification protocol given in Ref. [41].

Protocol $\mathcal{P}_{\mathsf{ZH}}$: Let $\mathsf{R}_1, \mathsf{R}_2, \ldots, \mathsf{R}_N, \mathsf{R}_{N+1}$ be $(N+1)$ two-qubit registers from the prover. Here, $|+\rangle := \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|-\rangle := \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, $|+'\rangle := \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ and $|-'\rangle := \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$.

1. Perform a random permutation π on the $(N+1)$ two-qubit registers, and rename $\mathsf{R}_j := \mathsf{R}_{\pi(j)}$ for $j = 1, 2, \ldots, N+1$.
2. For each $j = 1, 2, \ldots, N$, the verifier V does one of the following three POVMs on register R_j with probability $1/3$ for each:
   - $M_1 = \{E_1, I - E_1\}$ with $E_1 = |00\rangle\langle 00| + |11\rangle\langle 11|$.
   - $M_2 = \{E_2, I - E_2\}$ with $E_2 = |++\rangle\langle ++| + |--\rangle\langle --|$.
   - $M_3 = \{E_3, I - E_3\}$ with $E_3 = |+'-'\rangle\langle +'-'| + |-'+'\rangle\langle -'+'|$.
3. Reject if the second components in the POVMs are obtained. Otherwise, the test passes and outputs R_{N+1}.

Ref. [41] describes $E_1 = \frac{I+Z^{\otimes 2}}{2}$, $E_2 = \frac{I+X^{\otimes 2}}{2}$ and $E_3 = \frac{I-Y^{\otimes 2}}{2}$, where

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

while we rewrite them as above (which is a similar expression to the protocol in Ref. [34]) since it would be easy to see for our purpose. Importantly, step 2 implements the POVM $\{\Omega, I - \Omega\}$ on R_j with $\Omega = \frac{2}{3}|\Phi^+\rangle\langle\Phi^+| + \frac{1}{3}I$ for each j, but it is implemented by local measurements on each qubit of R_j.

The following result was shown for the protocol $\mathcal{P}_{\mathsf{ZH}}$ in Ref. [41].

▶ **Theorem 7** (Zhu-Hayashi). *There is a number $N = O(\frac{1}{\varepsilon}\log(\frac{1}{\delta}))$ such that if the test passed with probability at least δ, then the output state $\tilde{\sigma}$ of $\mathcal{P}_{\mathsf{ZH}}$ (under the condition that the test passes) satisfies $\langle\Phi^+|\tilde{\sigma}|\Phi^+\rangle \geq 1 - \varepsilon$.*

The protocol $\mathcal{P}_{\mathsf{ZH}}$ uses only local measurements, and thus, it can be used for verifying the sharing of an EPR pair by two parties who only use local operations and classical communication (LOCC).

Let V_1 and V_2 be neighboring parties who expect to receive $|\Phi^+\rangle$ jointly from the prover. The following protocol is a simple implementation of $\mathcal{P}_{\mathsf{ZH}}$ with LOCC by V_1 and V_2.

Protocol $\mathcal{P}_{\mathsf{ZHLOCC}}$: Let $\mathsf{R}_{1,1}, \ldots, \mathsf{R}_{N,1}, \mathsf{R}_{N+1,1}$ be $(N+1)$ one-qubit registers from the prover to V_1, and $\mathsf{R}_{1,2}, \ldots, \mathsf{R}_{N,2}, \mathsf{R}_{N+1,2}$ be $(N+1)$ one-qubit registers from the prover to V_2, respectively.

1. V_1 chooses a random permutation π on $\{1, 2, \ldots, N+1\}$ and sends it to V_2, and then both perform π on the $(N+1)$ two-qubit registers $(\mathsf{R}_{1,1}, \mathsf{R}_{1,2}), \ldots, (\mathsf{R}_{N,1}, \mathsf{R}_{N,2}), (\mathsf{R}_{N+1,1}, \mathsf{R}_{N+1,2})$. Rename $\mathsf{R}_{j,1} := \mathsf{R}_{\pi(j),1}$ and $\mathsf{R}_{j,2} := \mathsf{R}_{\pi(j),2}$.
2. V_1 chooses N random numbers $k_1, k_2, \ldots, k_N \in \{1, 2, 3\}$ and sends them to V_2. For each $j = 1, 2, \ldots, N$, V_1 and V_2 implement one of the POVMs M_1, M_2, M_3 on register $(\mathsf{R}_{j,1}, \mathsf{R}_{j,2})$ jointly as follows.
   - when $k_j = 1$, they jointly implement $M_1 = \{E_1, I - E_1\}$; V_1 and V_2 measure $\mathsf{R}_{j,1}$ and $\mathsf{R}_{j,2}$ in the Z basis $\{|0\rangle, |1\rangle\}$, respectively, and V_1 sends the measurement value to V_2, who rejects iff it differs from the measurement value of V_2.

- when $k_j = 2$, they jointly implement $M_2 = \{E_2, I - E_2\}$; V_1 and V_2 measure $\mathsf{R}_{j,1}$ and $\mathsf{R}_{j,2}$ in the X basis $\{|+\rangle, |-\rangle\}$, respectively, and V_1 sends the measurement value to V_2, who rejects iff it differs from the measurement value of V_2.
- when $k_j = 3$, they jointly implement $M_3 = \{E_3, I - E_3\}$; V_1 and V_2 measure $\mathsf{R}_{j,1}$ and $\mathsf{R}_{j,2}$ in the Y basis $\{|+'\rangle, |-'\rangle\}$, respectively, and V_1 sends the measurement value to V_2, who rejects iff it is same as the measurement value of V_2.

3. The test passes and V_1 and V_2 output $\mathsf{R}_{N+1,1}$ and $\mathsf{R}_{N+1,2}$, respectively.

It is easy to see that $\mathcal{P}_{\mathsf{ZHLOCC}}$ simulates $\mathcal{P}_{\mathsf{ZH}}$ exactly in a distributed manner. The protocol $\mathcal{P}_{\mathsf{ZHLOCC}}$ does not use any quantum communication between V_1 and V_2, while the amount of classical communication used between V_1 and V_2 is $\lceil \log(N+1)! \rceil + \lceil \log 3^N \rceil + N = O(N \log N)$.

Furthermore, we can replace a random permutation π in step 1 of $\mathcal{P}_{\mathsf{ZHLOCC}}$ by switching the jth two-qubit register $(\mathsf{R}_{j,1}, \mathsf{R}_{j,2})$ and the $(N+1)$th register $(\mathsf{R}_{N+1,1}, \mathsf{R}_{N+1,2})$ by choosing j uniformly at random from $\{1, 2, \ldots, N+1\}$ (actually, doing nothing when $j = N+1$) since the output state by such change is the same as protocol $\mathcal{P}_{\mathsf{ZHLOCC}}$. We call the protocol by such change $\mathcal{P}^+_{\mathsf{ZHLOCC}}$. Now the amount of classical communication used between V_1 and V_2 in $\mathcal{P}^+_{\mathsf{ZHLOCC}}$ is improved to $\lceil \log(N+1) \rceil + \lceil \log 3^N \rceil + N = O(N)$.

Thus the following theorem holds for $\mathcal{P}^+_{\mathsf{ZHLOCC}}$.

▶ **Theorem 8.** *For the same number $N = O(\frac{1}{\varepsilon} \log(\frac{1}{\delta}))$ as Theorem 7, if the test passed with at least probability δ, then the two-qubit state $\tilde{\sigma}$ output by V_1 and V_2 in $\mathcal{P}^+_{\mathsf{ZHLOCC}}$ satisfies $\langle \Phi^+ | \tilde{\sigma} | \Phi^+ \rangle \geq 1 - \varepsilon$.*

In the full version, we prove Theorem 5 by showing how to use the protocol $\mathcal{P}^+_{\mathsf{ZHLOCC}}$.

References

1 Scott Aaronson. The complexity of quantum states and transformations: from quantum money to black holes, 2016. arXiv:1607.05256.

2 Joran van Apeldoorn and Tijn de Vos. A framework for distributed quantum queries in the CONGEST model. In *Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing (PODC 2022)*, pages 109–119, 2022.

3 Heger Arfaoui and Pierre Fraigniaud. What can be computed without communications? *SIGACT News*, 45(3):82–104, 2014. doi:10.1145/2670418.2670440.

4 Charles H. Bennett, Gilles Brassard, Claude Crépeau, Richard Jozsa, Asher Peres, and William K. Wootters. Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Physical Review Letters*, 70:1895–1899, 1993.

5 Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87:167902, 2001. doi:10.1103/PhysRevLett.87.167902.

6 Matias Christandl, Robert König, Graeme Mitchison, and Renato Renner. One-and-a-half quantum de finetti theorem. *Communications in Mathematical Physics*, 273:473–498, 2007.

7 Pierluigi Crescenzi, Pierre Fraigniaud, and Ami Paz. Trade-offs in distributed interactive proofs. In *Proceedings of the 33rd International Symposium on Distributed Computing (DISC 2019)*, pages 13:1–13:17, 2019. doi:10.4230/LIPIcs.DISC.2019.13.

8 Vasil S. Denchev and Gopal Pandurangan. Distributed quantum computing: a new frontier in distributed systems or science fiction? *SIGACT News*, 39(3):77–95, 2008. doi:10.1145/1412700.1412718.

9 Michael Elkin, Hartmut Klauck, Danupon Nanongkai, and Gopal Pandurangan. Can quantum communication speed up distributed computation? In *Proceedings of the 33rd ACM Symposium on Principles of Distributed Computing (PODC 2014)*, pages 166–175, 2014.

10 Pierre Fraigniaud, François Le Gall, Harumichi Nishimura, and Ami Paz. Distributed quantum proofs for replicated data. In *Proceedings of the 12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, pages 28:1–28:20, 2021.

11 Pierre Fraigniaud, Pedro Montealegre, Rotem Oshman, Ivan Rapaport, and Ioan Todinca. On distributed Merlin-Arthur decision protocols. In *Proceedings of the 26th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2019)*, pages 230–245, 2019.

12 Pierre Fraigniaud, Boaz Patt-Shamir, and Mor Perry. Randomized proof-labeling schemes. *Distributed Computing*, 32(3):217–234, 2019. doi:10.1007/s00446-018-0340-8.

13 Cyril Gavoille, Adrian Kosowski, and Marcin Markiewicz. What can be observed locally? In *Proceedings of the 23rd International Symposium on Distributed Computing (DISC 2009)*, pages 243–257, 2009.

14 Mika Göös and Jukka Suomela. Locally checkable proofs in distributed computing. *Theory of Computing*, 12:19:1–19:33, 2016. doi:10.4086/toc.2016.v012a019.

15 Masahito Hayashi and Tomoyuki Morimae. Verifiable measurement-only blind quantum computing with stabilizer testing. *Physical Review Letters*, 115:220502, 2015.

16 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 (CCC 2022)*, pages 5:1–5:19, 2022.

17 Taisuke Izumi and François Le Gall. Quantum distributed algorithm for the All-Pairs Shortest Path problem in the CONGEST-CLIQUE model. In *Proceedings of the 38th ACM Symposium on Principles of Distributed Computing (PODC 2019)*, pages 84–93, 2019.

18 Taisuke Izumi, François Le Gall, and Frédéric Magniez. Quantum distributed algorithm for triangle finding in the CONGEST model. In *Proceedings of the 37th International Symposium on Theoretical Aspects of Computer Science (STACS 2020)*, pages 23:1–23:13, 2020.

19 Benjamin Jauregui, Pedro Montealegre, and Ivan Rapaport. Distributed interactive proofs for the recognition of some geometric intersection graph classes. In *Proceedings of 29th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2022)*, pages 212–233, 2022.

20 Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In *Proceedings of the 38th Annual International Cryptology Conference (CRYPTO 2018), Part III*, pages 126–152, 2018.

21 Martin Kliesch and Ingo Roth. Theory of quantum system certification. *PRX quantum*, 2(1):010201, 2021.

22 Gillat Kol, Rotem Oshman, and Raghuvansh R. Saxena. Interactive distributed proofs. In *Proceedings of the 37th ACM Symposium on Principles of Distributed Computing (PODC 2018)*, pages 255–264, 2018.

23 Amos Korman, Shay Kutten, and David Peleg. Proof labeling schemes. *Distributed Computing*, 22(4):215–233, 2010. doi:10.1007/s00446-010-0095-3.

24 William Kretschmer. Quantum pseudorandomness and classical complexity. In *Proceedings of the 16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*, pages 2:1–2:20, 2021.

25 François Le Gall and Frédéric Magniez. Sublinear-time quantum computation of the diameter in CONGEST networks. In *Proceedings of the 37th ACM Symposium on Principles of Distributed Computing (PODC 2018)*, pages 337–346, 2018.

26 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 (STACS 2023)*, pages 42:1–42:21, 2023.

27 François Le Gall, Harumichi Nishimura, and Ansis Rosmanis. Quantum advantage for the LOCAL model in distributed computing. In *Proceedings of the 36th International Symposium on Theoretical Aspects of Computer Science (STACS 2019)*, pages 49:1–49:14, 2019.

28 Ke Li and Graeme Smith. Quantum de Finetti theorem under fully-one-way adaptive measurements. *Physical Review Letters*, 114:160503, 2015.

29 Tony Metger and Henry Yuen. stateQIP = statePSPACE. `arXiv:2301.07730`. Presented at the 26th Conference on Quantum Information Processing (QIP2023).

30 Pedro Montealegre, Diego Ramírez-Romero, and Ivan Rapaport. Shared vs private randomness in distributed interactive proofs. In *Proceedings of the 31st International Symposium on Algorithms and Computation (ISAAC 2020)*, pages 51:1–51:13, 2020.

31 Pedro Montealegre, Diego Ramírez-Romero, and Ivan Rapaport. Compact distributed interactive proofs for the recognition of cographs and distance-hereditary graphs. In *Proceedings of the International Symposium on Stabilizing, Safety, and Security of Distributed Systems (SSS 2021)*, pages 395–409, 2021.

32 Tomoyuki Morimae, Yuki Takeuchi, and Masahito Hayashi. Verification of hypergraph states. *Physical Review A*, 96:062321, 2017.

33 Moni Naor, Merav Parter, and Eylon Yogev. The power of distributed verifiers in interactive proofs. In *Proceedings of the 31st ACM-SIAM Symposium on Discrete Algorithms (SODA 2020)*, pages 1096–115, 2020. `doi:10.1137/1.9781611975994.67`.

34 Sam Pallister, Noah Linden, and Ashley Montanaro. Optimal verification of entangled states with local measurements. *Physical Review Letters*, 120:170502, 2018. `doi:10.1103/PhysRevLett.120.170502`.

35 Gregory Rosenthal and Henry Yuen. Interactive proofs for synthesizing quantum states and unitaries. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, pages 112:1–112:4, 2022.

36 Seiichiro Tani, Hirotada Kobayashi, and Keiji Matsumoto. Exact quantum algorithms for the leader election problem. *ACM Transactions on Computation Theory*, 4(1):1:1–1:24, 2012. `doi:10.1145/2141938.2141939`.

37 Xudong Wu and Penghui Yao. Quantum complexity of weighted diameter and radius in CONGEST networks. In *Proceedings of the 42nd ACM Symposium on Principles of Distributed Computing (PODC 2022)*, pages 120–130, 2022.

38 Xiao-Dong Yu, Jiangwei Shang, and Otfried Gühne. Statistical methods for quantum state verification and fidelity estimation. *Advanced Quantum Technologies*, 5(5):2100126, 2022.

39 Huangjun Zhu and Masahito Hayashi. Efficient verification of pure quantum states in the adversarial scenario. *Physical Review Letters*, 123:260504, 2019.

40 Huangjun Zhu and Masahito Hayashi. General framework for verifying pure quantum states in the adversarial scenario. *Physical Review A*, 100:062335, 2019.

41 Huangjun Zhu and Masahito Hayashi. Optimal verification and fidelity estimation of maximally entangled states. *Physical Review A*, 99:052346, 2019.