

Quantum Simultaneous Protocols Without Public Coins Using Modified Equality Queries

François Le Gall ✉ 

Graduate School of Mathematics, Nagoya University, Japan

Oran Nadler ✉

Blavatnik School of Computer Science, Tel Aviv University, Israel

Harumichi Nishimura ✉

Graduate School of Informatics, Nagoya University, Japan

Rotem Oshman ✉ 

Blavatnik School of Computer Science, Tel Aviv University, Israel

Abstract

In this paper we study a quantum version of the multiparty simultaneous message-passing (SMP) model, and we show that in some cases, quantum communication can replace public randomness, even with no entanglement between the parties. This was already known for two players, but not for more than two players, and indeed, so far all that was known was a negative result. Our main technical contribution is a compiler that takes any classical public-coin simultaneous protocol based on “modified equality queries,” and converts it into a quantum simultaneous protocol without public coins with roughly the same communication complexity. We then use our compiler to derive protocols for several problems, including frequency moments, neighborhood diversity, enumeration of isolated cliques, and more.

2012 ACM Subject Classification Theory of computation → Quantum communication complexity

Keywords and phrases SMP model, multi-party communication, quantum distributed algorithms

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2024.34

Funding *François Le Gall*: JSPS KAKENHI grants Nos. JP20H05966, 20H00579, 24H00071, MEXT Q-LEAP grant No. JPMXS0120319794 and JST CREST grant No. JPMJCR24I4.

Harumichi Nishimura: JSPS KAKENHI grants Nos. JP20H05966, 22H00522, 24H00071, 24K22293, MEXT Q-LEAP grant No. JPMXS0120319794 and JST CREST grant No. JPMJCR24I4.

Rotem Oshman: ISF grants Nos. 2801/20 and 3725/24 and NSF-BSF grant No. 2022699.

1 Introduction

In the *multiparty simultaneous message-passing (SMP) model* we have k players with private inputs $x_1, \dots, x_k \in \{0, 1\}^n$, and we would like to compute a function $f(x_1, \dots, x_k)$ of the joint inputs. To this end, each player $\ell \in [k]$ computes a message M_ℓ , which it sends to a *referee*. The referee collects all k messages and produces an output, which should equal $f(x_1, \dots, x_k)$, except possibly with some small error probability. Our goal is to use as little *communication* as possible, that is, the total number of bits sent by the players to the referee should be minimized. There are several well-studied classical variants of the SMP model: a *deterministic* variant, where the participants (i.e., the players and the referee) have no randomness, and no error is allowed; a *public-coin* variant, where the participants have access to a common random string; and a *private-coin* variant, where each participant has access to its own random string. The public-coin variant is the strongest of the three, but arguably, it



© François Le Gall, Oran Nadler, Harumichi Nishimura, and Rotem Oshman; licensed under Creative Commons License CC-BY 4.0

28th International Conference on Principles of Distributed Systems (OPODIS 2024).

Editors: Silvia Bonomi, Letterio Galletta, Etienne Rivière, and Valerio Schiavoni; Article No. 34; pp. 34:1–34:20



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

is unrealistic: in the absence of prior coordination, many distributed systems do not have a source of common randomness, and must make do with the private randomness available to each participant.¹

In this paper we study a *quantum* version of the SMP model, and we show that in some cases, quantum communication can replace public randomness, even with no entanglement between the parties. This was already known for two players [9, 42], but it was not known for more than two players, and indeed, so far all that was known was a negative result [17] showing that for $k = 3$ players, there exist problems for which quantum communication with no public randomness or entanglement is exponentially weaker than classical communication *with* public randomness. Our main technical contribution is a compiler that takes any classical public-coin simultaneous protocol based on *modified equality queries* (which we define below), and converts it into a quantum *private-coin* simultaneous protocol with roughly the same communication complexity. We then use our compiler to derive protocols for several problems, including *frequency moments*, *neighborhood diversity* [31], *enumeration of isolated cliques* [30], and more.

Modified equality queries. We observe that quite a few problems studied in the literature can be solved by repeatedly executing the following type of query, $\text{MEQ}_{k,n}(i, j, y, z)$: for two indices $i, j \in [k]$, and two strings $y, z \in \{0, 1\}^n$ known only to the referee, is it the case that $x_i \oplus y = x_j \oplus z$? (Here, $x_i, x_j \in \{0, 1\}^n$ are the inputs of players i, j , respectively.) We refer to such queries as *modified equality queries*.

There is a well-known quantum simultaneous protocol [9] for “plain” equality queries, where we merely wish to determine whether $x_i = x_j$ (without the modifying strings y, z). In the protocol of [9], each player computes a short *quantum fingerprint* of its input and sends it to the referee, who then uses the quantum fingerprints to compare the players’ inputs (with some error probability; see Section 2 for the details). We observe that quantum fingerprints also allow us to implement *modified* equality queries, and moreover, this can be done even if the players do not know the strings y, z :

► **Lemma 1.** *For any $s \geq 1$, let \mathcal{Q}_s denote the set of all s -qubit quantum states.² For any $n \geq 1$ and $\varepsilon \in (0, 1)$, there is a quantum operator³ $F : \{0, 1\}^n \rightarrow \mathcal{Q}_s$ with $s = O(\log n \cdot \log(1/\varepsilon))$ such that if each player ℓ sends $F(x_\ell)$ to the referee, then for any $i, j \in [k]$ and any $y, z \in \{0, 1\}^n$, the referee can compute $\text{MEQ}_{k,n}(i, j, y, z)$ with error probability at most ε .*

We stress that Lemma 1 only states that the referee can compute $\text{MEQ}_{k,n}(i, j, y, z)$ for *one* 4-tuple (i, j, y, z) . Unlike classical protocols, in the quantum world it is not technically immediate to re-use information sent by the players to compute the value of the query for more than one 4-tuple (showing how to bypass this difficulty is indeed one of the main contributions of this paper).

Compiling MEQ decision trees into quantum protocols. Although Lemma 1 allows us to implement a single modified equality query, by itself it is not enough to obtain an efficient protocol for many of the problems we want to solve, as these problems require us to execute

¹ In *non-simultaneous* protocols it is possible to replace public randomness with private randomness [37], but this requires at least one synchronized round of communication, and synchronization comes with its own costs.

² An *s-qubit quantum state* is any quantum state that can be represented in s qubits; it is essentially a quantum superposition over classical s -bit strings. See Section 2 for the precise definition.

³ A *quantum operator* is an operation that maps one quantum state into another. In our case, we apply it to a classical string, which is slight notation abuse.

many such queries. For example, in the *distinct elements* problem, the goal is to determine the number of distinct values among x_1, \dots, x_k . Solving this problem requires $\binom{k}{2}$ “plain” equality queries, as every player’s input must be compared against all the others. In general, our goal is to work with protocols represented by an $MEQ_{k,n}$ *decision tree*: a rooted binary tree whose inner nodes are labeled by $MEQ_{k,n}$ queries, and whose leaves are labeled by output values (e.g., 0 or 1 if the tree computes a Boolean function). The tree is evaluated starting from the root, and at each step we evaluate the query written in the current node, proceeding to the left child if the answer is 0 and to right child if the answer is 1, until we eventually reach a leaf and output the value written in it.

A naïve application of Lemma 1 results in a quantum protocol whose communication cost scales linearly with the depth of the decision tree, as we must call the protocol from Lemma 1 at each step. However, we can do much better: we show that we can compile a decision tree into a quantum protocol whose communication cost depends only logarithmically on the depth of the tree. The key is to *re-use information*: instead of evaluating each modified equality query on its own, we would like to re-use the information sent by the players, so that evaluating multiple queries that involve the same player i will not require player i to send fresh information each time.

As already mentioned, unlike classical protocols, in the quantum world it is not technically immediate to re-use information sent by the players. First, quantum states cannot be duplicated (this is a consequence of the no-cloning theorem in quantum information theory). Second, if at any point we *measure* a quantum state, we may cause it to collapse, losing all the information that was stored in it (except for the outcome of the measurement), and preventing it from being re-used. This indeed happens in the protocol from Lemma 1. To avoid this pitfall, we use *gentle measurements* (see, e.g., [1, Section 1.3]), relying on the fact that a measurement whose outcome is “nearly certain” has very little effect on the quantum state we are measuring. To ensure that the outcome of each measurement we make is “nearly certain”, we *amplify* the success probability of each query $MEQ_{k,n}(i, j, y, z)$ so that if $x_i \oplus y = x_j \oplus z$ then the measurement returns 1 with probability nearly 1, and if $x_i \oplus y \neq x_j \oplus z$ then the measurement returns 1 with probability nearly 0. Finally, we observe that the quantum union bound by Gao [16] can be used and conclude that the measurements can be applied sequentially on the same state with only a small decrease of the success probability.

Ultimately, our result is the following:

► **Theorem 2.** *For any $n, k, D \geq 0$ and $\delta \in (0, 1)$, any $MEQ_{k,n}$ decision tree of depth D can be implemented by a quantum k -party SMP protocol that uses $O(k(\log D + \log(1/\delta)) \log n)$ qubits and has error probability at most δ .*

Applications. We give several applications of our compiler in Section 4. Several are technically straightforward. For instance, using “plain” equality queries, we can compare all the players’ inputs to one another, which allows us to count the number of distinct elements or compute other frequency moments of the input. Next we turn to more complex applications involving graphs: we show that in the number-in-hand network model [8] (a special case of the SMP model also sometimes called *broadcast congested clique*), we can use our compiler to obtain efficient simultaneous quantum protocols for P_3 - and P_4 -induced subgraph freeness [29, 36], computing neighborhood diversity [31], enumerating isolated cliques [30] and reconstructing distance-hereditary graphs [29, 36]. For all these problems,

we obtain efficient quantum protocols that do not require public randomness: in all of our protocols, each player only sends $\text{polylog}(n, k)$ qubits. This cost matches the cost of public-coin classical protocols and improves exponentially the cost of private-coin classical protocols.⁴

Relation with prior works on quantum distributed computing. Several works [2, 3, 4, 10, 11, 12, 14, 15, 17, 20, 22, 26, 27, 32, 33, 34, 35, 40, 41] have investigated how quantum communication can help for various computational tasks and settings in distributed computing. To our knowledge, theoretical aspects of the quantum multi-party SMP model have only been considered in Ref. [17], which we already mentioned, and Ref. [19], which focuses on a different input model (the number-on-the-forehead model). There are also a few experimental investigations of multiparty simultaneous quantum protocols [21, 39], but these works are mainly empirical and not concerned with asymptotic complexity.

2 Preliminaries

Notation and terminology. For any integer $n \geq 1$, we write $[n] = \{1, \dots, n\}$. For any strings $x, x' \in \{0, 1\}^n$, we denote by $\Delta_n(x, x')$ the Hamming distance between x and x' .

In this paper we consider undirected graphs with no self-loops $G = (V, E)$ over $k = |V|$ nodes (since the number of nodes will always match the number of players in the protocol, we use the same notation k for both). We use $\deg(v)$ to denote the degree of node $v \in V$. We often implicitly assume that $V = \{1, \dots, k\}$. Let $N(v) \subseteq V$ denote the neighbors of node $v \in V$ in G , and $\nu_v \in \{0, 1\}^k$ denote the characteristic vector of $N(v)$, where $\nu_v[u] = 1$ iff $\{v, u\} \in E$ for each $u \in [k]$. Let $e_v \in \{0, 1\}^k$ be the characteristic vector of the singleton $\{v\}$, i.e., the vector where $e_v[u] = 1$ iff $u = v$. For a subset $S \subseteq V$, we use $G[S]$ to denote the subgraph of G induced by S . For a node $v \in V$, we define $G - v$ as $G - v = G[V \setminus \{v\}]$.

A node v in G is called *pendant* if v has only one neighbor. Two nodes u, v in G are called *false twins* (resp., *true twins*) if u and v are not adjacent (resp., adjacent), and have the same neighborhood, that is, $N(u) = N(v)$ (resp., $N(u) \cup \{u\} = N(v) \cup \{v\}$, or equivalently, $N(u) \setminus \{v\} = N(v) \setminus \{u\}$). We say that u, v are *twins* if they are either true twins or false twins. Note that two non-adjacent nodes u, v cannot have $N(u) \cup \{u\} = N(v) \cup \{v\}$, and because there are no self-loops in the graph, two adjacent nodes u, v cannot have $N(u) = N(v)$. Therefore, in terms of neighborhood vectors, we have:

► **Proposition 3.** *Nodes $u \neq v$ are false twins if and only if $\nu_u = \nu_v$, and true twins if and only if $\nu_u \oplus e_u = \nu_v \oplus e_v$.*

SMP protocols and NIH network model. A *simultaneous message-passing* (SMP) protocol features k players with inputs $x_1, \dots, x_k \in \{0, 1\}^n$, respectively, and a referee, who does not know x_1, \dots, x_k . In the protocol, each player sends one message to the referee, and the referee then produces an output. The goal of the referee is to compute some function $f(x_1, \dots, x_k)$ of the inputs, and we say that the protocol *succeeds* whenever the referee's output is correct. The *communication cost* of the protocol is the maximum total number of bits sent by the players to the referee in any execution of the protocol, on any input. We say that a protocol is *bounded-error* if for any input, it outputs the correct answer with probability at least $2/3$. In this paper we do not assume that the players have shared randomness; each player's message depends only on its own input.

⁴ For private-coin classical protocols a lower bound of the form $\Omega(\sqrt{n})$ or $\Omega(\sqrt{k})$ trivially follows from the lower bound on the cost of private-coin classical protocols for the two-party equality function [5, 38].

A special case of the SMP model is the *number-in-hand (NIH) network model* [8]. Here, the input to the computation is an undirected graph $G = (V, E)$ over k nodes, and each party represents a node in the graph. The input to player $v \in [k]$ is the neighborhood vector $\nu_v \in \{0, 1\}^k$ (we thus have $n = k$ in this case), and the referee is asked to solve some graph problem on G .

In the quantum versions of the SMP model and the NIH network model, the only difference is that players are allowed to send quantum information to the referee. The *communication cost* of the protocol is the maximum total number of quantum bits (qubits) sent by the players to the referee in any execution of the protocol. We do not assume that the players have shared randomness or shared entanglement; each player's message depends again only on its own input.

Basics of quantum information. The most basic notion in quantum information is the concept of quantum bit (qubit), which represents the state of an elementary physical system that follows the laws of quantum mechanics (e.g., one photon). Qubits are physically stored in *quantum registers*. Mathematically, the state of a quantum register consisting of q qubits is described by a unit-norm complex vector of dimension m , where $m = 2^q$, and usually written using Dirac's notation as $|\psi\rangle$. By taking an orthonormal basis of the m -dimensional complex vector space and indexing these basis vectors (again using Dirac's notation) as $|j\rangle$ for all $j \in \{1, \dots, m\}$, we can write $|\psi\rangle = \sum_{j=1}^m \alpha_j |j\rangle$ for complex numbers α_j such that the state has norm 1 (i.e., satisfying $\sum_{j=1}^m |\alpha_j|^2 = 1$).

All transformations on quantum registers need to be unitary, i.e., described by unitary matrices. The main unitary matrices that will appear in the technical parts of this paper are the Hadamard gate (denoted H) and the Pauli X gate, both acting on 1 qubit, and the $CNOT$ gate acting on two qubits. The precise definition of these gates will not be necessary for understanding this paper.

Information can only be extracted from a quantum register by measurements. The most elementary type of measurements is measurement of a 1-qubit register in the computational basis. For a 1-qubit register in the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where $|\alpha|^2 + |\beta|^2 = 1$, measuring it in the computational basis gives as outcome 1 bit: the outcome is 0 with probability $|\alpha|^2$ and 1 with probability $|\beta|^2$. Importantly, the state collapses (i.e., is irreversibly modified) after the measurement: in the former case the postmeasurement state is $|0\rangle$, while in the later case the postmeasurement state is $|1\rangle$.

Several more general kinds of measurements are allowed by quantum mechanics. In this paper, we will mostly use the *2-outcome measurements* defined as follows. A 2-outcome measurement of a q -qubit register R is the following process: introduce a new 1-qubit register S initialized to $|0\rangle$, apply a unitary transform U on the whole system, and then measure Register S in the computational basis, which gives as outcome a bit $b \in \{0, 1\}$. We refer to Figure 3 in Appendix A for an illustration of the process. We denote such a 2-outcome measurement by \mathcal{M} . We will also use the following notation: for any bit $b \in \{0, 1\}$ and any q -qubit quantum state $|\psi\rangle$, we denote by $\mathcal{M}^b(|\psi\rangle)$ the probability of obtaining outcome b by \mathcal{M} when the initial state in R is $|\psi\rangle$.

Quantum union bound. We now present the quantum union bound by Gao [16]. While this bound only applies to a special type of 2-outcome measurements called 2-outcome projective measurements (defined in Appendix B), any 2-outcome measurement can actually be efficiently converted into a 2-outcome projective measurement (the conversion is described in Appendix B).

Consider several 2-outcome projective measurements $\mathcal{M}_1, \dots, \mathcal{M}_N$ acting on the same q -qubit register. Consider what happens when performing these N measurements *sequentially*. Specifically, assume that the system is initially in state $|\psi\rangle$. We first perform \mathcal{M}_1 on $|\psi\rangle$ and obtain a postmeasurement state $|\psi_1\rangle$. Then we perform \mathcal{M}_2 on $|\psi_1\rangle$ and obtain the postmeasurement state $|\psi_2\rangle$. And so it carries on, with each measurement being performed on the state resulting from the previous measurement. After N measurements, we obtain the state $|\psi_N\rangle$. For an arbitrary binary string $s \in \{0, 1\}^N$, we would like to estimate the probability that the sequence of outcomes of this measurement process is s , i.e., the probability that for all $i \in \{1, \dots, N\}$, the outcome of measurement \mathcal{M}_i is the bit s_i . The following theorem by Gao [16] shows that this probability is high when for each $i \in \{1, \dots, N\}$ applying measurement \mathcal{M}_i on the *initial state* $|\psi\rangle$ gives outcome s_i with high probability.

► **Theorem 4** (Quantum union bound [16]). *For any string $s \in \{0, 1\}^N$, the probability that the above sequential measurement process has outcome s is at least*

$$1 - 4 \sum_{i=1}^N \left(1 - \mathcal{M}_i^{s_i}(|\psi\rangle)\right).$$

The SWAP test. The SWAP test [7, 9] is a quantum protocol that checks whether two quantum states $|\psi_1\rangle$ and $|\psi_2\rangle$ stored in two q -qubit registers R_1 and R_2 , respectively, are close or not (i.e., estimates their inner product). For completeness we give a detailed description of the test in Appendix C (this detailed description is not needed to understand the claims of this paper). The main property of the SWAP test is that the test outputs 1 with probability $\frac{1}{2} + \frac{1}{2}|\langle\psi_1|\psi_2\rangle|^2$ (and outputs 0 with probability $\frac{1}{2} - \frac{1}{2}|\langle\psi_1|\psi_2\rangle|^2$), where $\langle\psi_1|\psi_2\rangle$ denotes the inner product between $|\psi_1\rangle$ and $|\psi_2\rangle$.

The SWAP test is especially useful when combined with the notion of quantum fingerprints. We first give the definition of this concept.

► **Definition 5.** *A quantum fingerprint family for the set of n -bit strings is a family $\{|h_x\rangle : x \in \{0, 1\}^n\}$ such that the following conditions hold for each $x \in \{0, 1\}^n$:*

1. $|h_x\rangle$ is a $O(\log n)$ -qubit quantum state;
2. $|\langle h_x|h_{x'}\rangle| \leq \zeta$ holds for all $x' \in \{0, 1\}^n \setminus \{x\}$, for some universal constant $\zeta \in (0, 1/2]$.

For a quantum fingerprint family $\{|h_x\rangle : x \in \{0, 1\}^n\}$, the SWAP test on states $|\psi_1\rangle = |h_x\rangle$ and $|\psi_2\rangle = |h_{x'}\rangle$ outputs 1 with probability 1 if $x = x'$ (since $\langle h_x|h_x\rangle = 1$) and outputs 1 with probability at most $\frac{1}{2} + \frac{\zeta^2}{2}$ if $x \neq x'$ (since $|\langle h_x|h_{x'}\rangle| \leq \zeta$). For later reference, we state this result in the following lemma.

► **Lemma 6.** *When $|h_x\rangle$ and $|h_{x'}\rangle$ are given in R_1 and R_2 , respectively, the SWAP test outputs 1 with probability 1 if $x = x'$, and outputs 1 with probability at most $\frac{1}{2} + \frac{\zeta^2}{2} \leq \frac{5}{8}$ if $x \neq x'$.*

Ref. [9] showed how to create quantum fingerprint families. We will actually need a special kind of quantum fingerprint families, also used in [18], that satisfies the following additional property: for any known string $y \in \{0, 1\}^n$, the fingerprint of x can be converted to the fingerprint of $x \oplus y$ (by a unitary transformation depending on y) without knowing the fingerprint of x . We call a quantum fingerprint family satisfying this additional property a *linear quantum fingerprint family*. Ref. [18] showed how to construct a linear quantum fingerprint family, which we write $\{|\Psi_x\rangle : x \in \{0, 1\}^n\}$.

For completeness, we briefly describe the construction from [18], which is based on constant rate linear error-correcting codes (the details of the construction will not be needed to understand the results in this paper). Take a linear function $E : \{0, 1\}^n \rightarrow \{0, 1\}^m$ where $m = O(n)$ such that $\Delta_m(E(x), E(x')) = \Omega(m)$ for any distinct $x, x' \in \{0, 1\}^n$. The corresponding quantum fingerprint of x is then defined as the $O(\log n)$ -qubit quantum state

$$|\Psi_x\rangle = \frac{1}{\sqrt{m}} \sum_{j=1}^m (-1)^{E(x)_j} |j\rangle,$$

where $E(x)_j$ denotes the j th bit of $E(x)$. It is easy to check that this family of states satisfies all the required conditions. In particular, for any $y \in \{0, 1\}^n$, the state $|\Psi_x\rangle$ can be converted to $|\Psi_{x \oplus y}\rangle$ by a unitary transformation (depending on y) without knowing $|\Psi_x\rangle$.

3 Quantum SMP Protocols Based on MEQ Decision Trees

In this section we prove the main technical results of this paper (Lemma 1 and Theorem 2).

3.1 Implementing a Single Query: Proof of Lemma 1

We first give a brief sketch of the proof. We use the linear quantum fingerprint family $\{|\Psi_x\rangle : x \in \{0, 1\}^n\}$ introduced at the end of Section 2. Each player sends the fingerprint corresponding to its input (i.e., player ℓ sends the state $|\Psi_{x_\ell}\rangle$). The referee then implements the SWAP test on the states $|\Psi_{x_i \oplus y}\rangle$ and $|\Psi_{x_j \oplus z}\rangle$, which can be constructed from the messages of player i and the player j due to the linearity property of the quantum fingerprint family. From Lemma 6, we know that the success probability of the SWAP test is at least $5/8$. We amplify the success probability by applying $O(\log(1/\varepsilon))$ SWAP tests in parallel, which requires each player to actually send $O(\log(1/\varepsilon))$ copies of its quantum fingerprint. We now explain all the details of the proof.

Proof of Lemma 1. Take $t = \Theta(\log(1/\varepsilon))$. For any $x \in \{0, 1\}^n$ we define $F(x) = |\Psi_x\rangle^{\otimes t}$, i.e., t copies of the (linear) quantum fingerprint of x . Since each state $|\Psi_x\rangle$ is encoded by $O(\log n)$ qubits, $F(x)$ is a quantum state of $O(t \log n) = O(\log(1/\varepsilon) \log n)$ qubits, as claimed. We now describe and analyze the referee's procedure.

Description of the referee's procedure. Remember that the referee knows the indices i, j and the strings y, z . The referee receives the quantum message $F(x_\ell)$ from player ℓ , for each $\ell \in \{1, \dots, k\}$. We assume that $F(x_\ell)$ is stored by the referee in registers $(R_{\ell,1}, \dots, R_{\ell,t})$, where each $R_{\ell,r}$ stores one copy of $|\Psi_{x_\ell}\rangle$.

The referee implements the procedure of Figure 1. Note that the conversion at Step 1 can be done locally by the referee since the referee knows y and z (remember that we are using linear quantum fingerprints, for which such a conversion is possible). Also note that Step 2 essentially implements, for each r , the SWAP test on registers $(R_{i,r}, R_{j,r})$. The only difference with the SWAP test described in Section 2 (and Appendix C) is that Register S_r is not measured. Instead, the AND of all the Registers S_1, \dots, S_t is computed in a new register, which is then measured.

1. Convert $|\Psi_{x_i}\rangle^{\otimes t}$ into $|\Psi_{x_i \oplus y}\rangle^{\otimes t}$ in Registers $(R_{i,1}, \dots, R_{i,t})$.
Convert $|\Psi_{x_j}\rangle^{\otimes t}$ into $|\Psi_{x_j \oplus z}\rangle^{\otimes t}$ in Registers $(R_{j,1}, \dots, R_{j,t})$.
2. For every $r = 1, \dots, t$:
 - 2.1 Introduce a register S_r initialized to $|0\rangle$.
 - 2.2 Apply the Hadamard gate H to S_r .
 - 2.3 (Controlled SWAP) If the content of S_r is 1, swap $R_{i,r}$ and $R_{j,r}$.
 - 2.4 Apply the Hadamard gate H and then the X gate on S_r .
3. Compute the AND of all the registers S_1, \dots, S_t in a new 1-qubit register S .
4. Measure Register S in the computational basis.

■ **Figure 1** Description of the referee's procedure for Lemma 1.

Analysis of the referee's procedure. We now analyze the success probability of the procedure. First assume that $\text{MEQ}_{k,n}(i, j, y, z) = 1$. Then from Lemma 6 we know that each SWAP test would output 1 with probability 1. This means that at Step 4, the measurement outcome is 1 with probability 1.

Now assume that $\text{MEQ}_{k,n}(i, j, y, z) = 0$. By Lemma 6, for each $r \in \{1, \dots, t\}$, the SWAP test on Registers $(R_{i,r}, R_{j,r})$, which has input $(|\Psi_{x_i}\rangle, |\Psi_{x_j}\rangle)$, would then output 1 with probability at most $\frac{5}{8}$. Thus at Step 4, the measurement outcome is 0 with probability at least $1 - \left(\frac{5}{8}\right)^t \geq 1 - \varepsilon$, where the inequality follows from our choice of t .

In both cases we thus have success probability at least $1 - \varepsilon$, as desired. ◀

3.2 Implementing an MEQ Decision Tree: Proof of Theorem 2

MEQ decision trees. We define the model of Modified Equality Query decision trees ($\text{MEQ}_{k,n}$ decision trees, or MEQ decision trees when the parameters k, n are clear from the context) and the computation associated with them as follows.

- The input consists of k n -bit strings X_1, \dots, X_k ; the output is an element of a set S .
- The computational process is described by a binary tree \mathcal{T} in which each node has either 0 or 2 children. Each internal node of the tree (i.e., each node with 2 children) is labeled by a 4-tuple (i, j, y, z) for some indices $i, j \in \{1, \dots, k\}$ and some (known) strings $y, z \in \{0, 1\}^n$. Each leaf (i.e., each node with 0 child) is labeled by an element in S .
- The computation proceeds as follows. We start at the root. At each internal node we proceed to the right child if $X_i \oplus y = X_j \oplus z$ and to the left child if $X_i \oplus y \neq X_j \oplus z$. When reaching a leaf, we stop and output the label of the leaf.

Observe that for any input X_1, \dots, X_k , the above computational process can be implemented using at most D modified equality queries, where D denotes the depth of \mathcal{T} . For a function $f: (\{0, 1\}^n)^k \rightarrow S$, we say that \mathcal{T} computes f if the output of the computational process induced by \mathcal{T} is equal to $f(X_1, \dots, X_k)$ for any $X_1, \dots, X_k \in \{0, 1\}^n$.

Converting MEQ decision trees into SMP protocols. Here is our main theorem (repeated from the introduction).

► **Theorem 3 (repeated).** *For any $n, k, D \geq 0$ and $\delta \in (0, 1)$, any $\text{MEQ}_{k,n}$ decision tree of depth D can be implemented by a quantum SMP protocol that uses $O(k(\log D + \log(1/\delta)) \log n)$ qubits and has error probability at most δ .*

Proof. For each $r \in \{1, \dots, k\}$, ℓ sends to the referee the quantum state $F(x_\ell)$ specified by Lemma 1 with $\varepsilon = \frac{\delta}{4D}$. The total communication cost is thus $O(k \log(1/\varepsilon) \log n) = O(k(\log D + \log(1/\delta)) \log n)$, as claimed.

The referee then implements the computation induced by the $\text{MEQ}_{k,n}$ decision tree, by starting from the root and then following the computational path. This requires making at most D modified equality queries sequentially.

Note that one individual modified equality query $\text{MEQ}_{k,n}(i, j, y, z)$ can be implemented using the quantum states $F(x_i)$ and $F(x_j)$ received from player i and player j . By using the procedure of Figure 1 on these two quantum states, the success probability is at least $1 - \frac{\delta}{4D}$.

The main issue is that the procedure of Figure 1 modifies the quantum states $F(x_i)$ and $F(x_j)$, which prevents reusing them for implementing the next modified equality query. To solve this issue, we convert the procedure of Figure 1 (which corresponds to a 2-outcome non-projective measurement) into a 2-outcome projective measurement using the conversion process mentioned in Section 2 and described in Appendix B. Since this conversion preserves the success probability of the measurement, the success probability of the 2-outcome projective measurement that we obtain is at least $1 - \frac{\delta}{4D}$. We implement the modified equality queries on the computation path by applying the corresponding 2-outcome projective measurements sequentially. Theorem 4 shows that the overall success probability is at least $1 - 4D(1 - (1 - \frac{\delta}{4D})) = 1 - \delta$, as claimed. ◀

4 Applications

In this section we present several applications of our compiler, ranging from statistical problems to graph problems.

4.1 Warm-up: Grouping By Equality

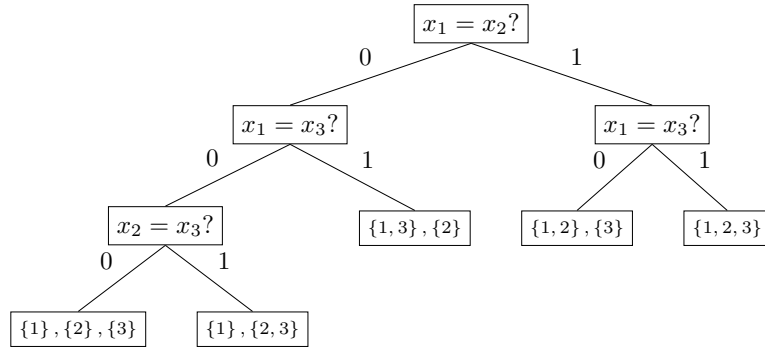
The simplest application of our compiler is to efficiently group the players by input, so that all players with the same input are in the same group: formally, the $\text{GroupByEQ}_{k,n}$ problem requires the referee to output a partition P_1, \dots, P_s of $[k]$, such that for every $i, j \in [k]$, there is an index t such that $i, j \in P_t$ if and only if $x_i = x_j$, where $x_1, \dots, x_k \in \{0, 1\}^n$ are the players' inputs.

► **Theorem 8.** *There exists a bounded-error SMP quantum protocol for $\text{GroupByEQ}_{k,n}$ with communication cost $O(k \log k \log n)$.*

Proof. The $\text{GroupByEQ}_{k,n}$ problem can be solved by an $\text{MEQ}_{k,n}$ decision tree of depth $\binom{k}{2}$, where on each path we compare players' inputs against one another until we arrive at the correct output partition. See Figure 2 for an example with $k = 3$ players. Note that not all paths have the same length, as sometimes we can deduce the answer without comparing all inputs against one another; for example, if we learn that $x_1 = x_2$, then we no longer need to compare x_2 against the other inputs, as the query answers we obtain for x_1 imply the answers for x_2 . The longest path is the leftmost path, where all queries return 0 (“not equal”), and the length of this path is exactly $\binom{k}{2}$.

The conclusion then follows from Theorem 2. ◀

Using our protocol for $\text{GroupByEQ}_{k,n}$ we can immediately solve several related problems. First, we use it to solve the $\text{AllEQ}_{k,n}$ and $\text{ExistsEQ}_{k,n}$ problems, which ask us to determine whether all inputs are the same, or whether there exist two players that have the same input, respectively. Ref. [13] showed that for any constant $\varepsilon > 0$, the classical communication costs



■ **Figure 2** An $\text{MEQ}_{k,n}$ decision tree for $\text{GroupByEQ}_{k,n}$ with $k = 3$ players. Each inner node is labeled with a query of the form “ $x_i = x_j?$ ”, which is short-hand notation for $\text{MEQ}_{k,n}(i, j, 0^n, 0^n)$. The leaves are labeled with output partitions.

of $\text{ALLEQ}_{k,n}$ and $\text{ExistsEQ}_{k,n}$ in the private-coin SMP model are $\tilde{\Theta}(\sqrt{kn} + k)$ and $\tilde{\Theta}(k\sqrt{n})$, respectively. Both problems reduce trivially to $\text{GroupByEQ}_{k,n}$; therefore Theorem 8 implies a quantum SMP protocol with communication cost $O(k \log k \log n)$ for both problems,⁵ which is an exponential improvement in the dependence on n . More generally, for any $p \geq 0$, we can compute the p -th frequency moment of the input, $F_p = \sum_{w \in \{0,1\}^n} (f_w)^p$, where f_w is the frequency of the string w in the input (i.e., the number of players whose input is w). The case $p = 0$ corresponds to counting the number of distinct inputs.

Finally, we can use our protocol for grouping by equality to solve P_3 -induced subgraph freeness: this problem is set in the NIH network model (as explained in Section 2), and requires us to determine whether the input graph contains an induced path consisting of two edges (P_3). As observed in [29], a graph G is P_3 -induced subgraph free if and only if G is a collection of node-disjoint cliques. This can be tested by grouping the nodes of the graph using the input $\nu_v \oplus e_v$ (that is, the characteristic vector of $N(v) \cup \{v\}$) for each node v , and then checking if for each node $v \in V$, the number of nodes grouped together with v (excluding v itself) is exactly $\text{deg}(v)$. To implement this test, we have each node send its degree to the referee, and then apply our protocol for $\text{GroupByEQ}_{k,k}$ to the vectors $\{\nu_v \oplus e_v\}_{v \in V}$. The total communication cost is $O(k \log^2 k)$ qubits, nearly matching the cost of the *public coin* classical protocol from [29].

4.2 Neighborhood Diversity

Our next application is to computing *neighborhood diversity* [31], a graph parameter that is used in fixed-parameter tractability to measure the density of a graph (in the same way that *treewidth*, *cliquewidth*, and other parameters are sometimes used).

The following definition is stated in the terminology of twins, for the sake of consistency with the remainder of the paper, although this is not the terminology used in [31]:

▶ **Definition 9** ([31]). *A graph $G = (V, E)$ has neighborhood diversity d if its nodes can be partitioned into d sets but no fewer, such that all nodes in each set are twins (false or true) of one another.*

⁵ We remark that for $\text{ALLEQ}_{k,n}$, this can be further improved to $O(k \log n)$ by using the permutation test [7, 9, 28] instead of the SWAP test that we use here.

We design an efficient quantum protocol for computing the neighborhood diversity of a graph, based on Proposition 3 from Section 2 (which is similar to what is used in, e.g., [36]):

► **Theorem 10.** *In the NIH network model, there exists a bounded-error quantum protocol for computing the neighborhood diversity with communication cost $O(k \log^2 k)$.*

Proof. Proposition 3 shows that nodes $v, w \in V$ are twins (false or true) if and only if $\nu_v = \nu_w$ or $\nu_v \oplus e_v = \nu_w \oplus e_w$. To compute the neighborhood diversity of G , we group nodes into sets of twins in much the same way that we used to solve GroupByEQ $_{k,n}$ above, except that to determine whether two nodes v, w are twins we need two queries: $\text{MEQ}_{k,k}(\nu_v, \nu_w, 0^k, 0^k)$ and $\text{MEQ}_{k,k}(\nu_v, \nu_w, e_v, e_w)$. The resulting decision tree has depth $2\binom{k}{2}$, and the protocol then follows by Theorem 2. ◀

We remark that computing neighborhood diversity can actually be done using “plain” equality queries alone (we do not need modified equality queries): we can implement the tests above by having each node $v \in V$ send the referee a quantum fingerprint of its neighborhood ν_v , and also of $\nu_v \oplus e_v$. However, this would fall outside the framework for our compiler, so it is simpler here to use modified equality queries and apply Theorem 2.

4.3 Reconstruction of Distance-Hereditary Graphs

The *reconstruction* task in the NIH network model requires the referee to output the entire input graph G . For information-theoretic reasons, general graphs require a total of $\Theta(k^2)$ communication to reconstruct, as this is the number of bits needed to represent an arbitrary graph on k nodes. However, for special classes of graphs, we can sometimes do much better: for example, [29] showed that there is an efficient classical public-coin SMP protocol which reconstructs the input graph if it is P_4 -induced subgraph free, and rejects if it is not P_4 -induced subgraph free. This was generalized in [36] to distance-hereditary and bounded modular-width graphs. In this subsection, we show that there is an efficient quantum private-coin SMP protocol for reconstructing distance-hereditary graphs. The protocols of [29, 36] for reconstructing P_4 -induced subgraph free graphs and bounded modular-width graphs can be adapted in a similar manner.

Distance-hereditary graphs and their properties. A graph $G = (V, E)$ is called *distance-hereditary* if the distance between two nodes v, w belonging to the same connected component in G is preserved in any induced subgraph of G that contains v and w . Distance-hereditary graphs are characterized by the existence of a decomposition called a *twin-pendant node decomposition* – a sequence (v_1, \dots, v_k) of nodes of G , such that for each $j \in [k - 1]$, one of the following conditions is true:

- (C1) v_j is a pendant node in $G[\{v_j, \dots, v_k\}]$
- (C2) v_j has a true twin in $G[\{v_j, \dots, v_k\}]$.
- (C3) v_j has a false twin in $G[\{v_j, \dots, v_k\}]$.

It is known that a graph is distance-hereditary if and only if it has a twin-pendant decomposition (see, e.g., [6]). Moreover, if the graph is distance-hereditary, then the twin-pendant decomposition can be computed by repeatedly choosing an arbitrary node satisfying one of the three conditions and removing it from the graph [29]. This forms the basis for the reconstruction protocol given in [29, 36]. The protocol of [29, 36] is stated in terms of polynomials, but we observe that it actually relies on simple algebraic properties, and can be translated to work with binary strings (interpreted as vectors over the binary field \mathbb{F}_2), as we do next. (This abstracts and simplifies the algorithm of [29, 36].)

An algebraic characterization of pendant nodes and twins. The key to reconstructing distance-hereditary graphs is to find a representation of the graph that allows us to repeatedly:

- (1) *Find* a node satisfying one of the three conditions (C1)–(C3), and
- (2) *Remove* this node from the graph and update our representation accordingly.

The requirement of Definition 11 below is an adaptation and simplification of a corresponding requirement from [29, 36]. It requires that the graph be represented by a collection of linearly-independent vectors, one for each node, such that the neighborhood of each node is the sum of the representations of its neighbors, allowing us to later “peel off” nodes from the graph by “subtracting” their representations.

► **Definition 11.** Let $G = (V, E)$ be a k -node graph and ℓ be a positive integer. A family of vectors $m = \{(a_v, b_v)\}_{v \in V}$, where $a_v, b_v \in \mathbb{F}_2^\ell$ for each $v \in V$, is a valid representation of G (or valid for G , for short) if:

1. $\{a_v\}_{v \in V}$ are linearly independent over \mathbb{F}_2^ℓ , that is, there is no non-empty subset $U \subseteq V$ such that $\bigoplus_{u \in U} a_u = 0^\ell$; and
2. For each $v \in V$, we have $b_v = \bigoplus_{u \in N(v)} a_u$.

The linear independence requirement of Definition 11 leads to an algebraic characterization of the concepts of pendant nodes and twins which will be crucial to our algorithm:

► **Proposition 12.** If $\{(a_v, b_v)\}_{v \in V}$ is a valid representation of $G = (V, E)$, then for every two nodes $w \neq u$ in G ,

1. Node w is pendant and has node u as its only neighbor if and only if $b_w = a_u$;
2. Nodes w, u are false twins if and only if $b_w = b_u$;
3. Nodes w, u are true twins if and only if $b_w \oplus a_w = b_u \oplus a_u$.

This follows from the following property of linearly-independent sets: if $\{a_v\}_{v \in V}$ is a linearly-independent set of vectors over \mathbb{F}_2^ℓ , then for any two sets $S, T \subseteq V$ we have $\bigoplus_{u \in S} a_u = \bigoplus_{u \in T} a_u$ if and only if $S = T$.

Our algorithm will work with a valid representation of the graph, and modify it as it goes along. The initial representation we will use is the following:

► **Proposition 13.** The representation $\{(e_v, \nu_v)\}_{v \in V}$ is valid for $G = (V, E)$.

Proof. Indeed, $\{e_v\}_{v \in V}$ are linearly independent, and $\nu_v = \bigoplus_{u \in N(v)} e_u$ for each $v \in V$. ◀

Next we show how to modify a valid representation after removing a node w from the graph, so that we obtain a valid representation for the remainder of the graph (the proof is very similar to [29, 36], and is omitted here):

► **Lemma 14.** Let $m = \{(a_v, b_v)\}_{v \in V}$ be valid for $G = (V, E)$ and let $w \neq u$ be nodes in G . We can obtain a valid representation $m' = \{(a'_v, b'_v)\}_{v \in V \setminus \{w\}}$ for $G - w$ as follows:

- I. If w is pendant and u is its only neighbor: then for all $v \in V \setminus \{w\}$,

$$a'_v = a_v \quad \text{and} \quad b'_v = \begin{cases} b_v & (v \neq u), \\ b_u \oplus a_w & (v = u). \end{cases}$$

- II. If w, u are false twins: then for all $v \in V \setminus \{w\}$,

$$a'_v = \begin{cases} a_v & (v \neq u), \\ a_u \oplus a_w & (v = u), \end{cases} \quad \text{and} \quad b'_v = b_v.$$

III. If w, u are true twins: then for all $v \in V \setminus \{w\}$,

$$a'_v = \begin{cases} a_v & (v \neq u), \\ a_u \oplus a_w & (v = u), \end{cases} \quad \text{and} \quad b'_v = \begin{cases} b_v & (v \neq u), \\ b_u \oplus a_w & (v = u). \end{cases}$$

Together, Propositions 12, 13 and Lemma 14 give rise to the following abstract protocol for computing a pendant-twin decomposition (or identifying that the graph is not distance-hereditary). We cannot efficiently implement this protocol in the SMP model, as it requires players to send very long messages, but we will show that we can *simulate* it using modified equality queries. From the twin- pendant decomposition output by Algorithm 1 it is easy to reconstruct the entire graph (as in [29, 36]).

■ **Algorithm 1** Abstract Protocol for Computing a Pendant-Twin Decomposition.

Input: The representation $\{(e_v, \nu_v)\}_{v \in V}$ of the graph G

- 1 Set $a_v \leftarrow e_v, b_v \leftarrow \nu_v$ for each $v \in V$
- 2 Set $decomp \leftarrow \lambda$ (an empty sequence)
- 3 **while** $|V| \geq 2$ **do**
- 4 **if** $\exists w, u$ such that $b_w = a_u$ **then** // Pendant w with neighbor u
- 5 Append (“pendant”, w, u) to $decomp$
- 6 Apply update (I) from Lemma 14 and remove w from V
- 7 **else if** $\exists w, v$ such that $b_w = b_u$ **then** // False twins w, u
- 8 Append (“false twin”, w, u) to $decomp$
- 9 Apply update (II) from Lemma 14 and remove w from V
- 10 **else if** $\exists w, u$ such that $b_w \oplus a_w = b_u \oplus a_u$ **then** // True twins w, u
- 11 Append (“true twin”, w, u) to $decomp$
- 12 Apply update (III) from Lemma 14 and remove w from V
- 13 **else** Output “Graph is not distance-hereditary”
- 14 Output $decomp$

Simulating the abstract protocol using modified equality queries. As we said above, we cannot actually afford to implement Algorithm 1: sending full neighborhood vectors ν_v requires n qubits, so we cannot even send the initial representation $\{(e_v, \nu_v)\}_{v \in V}$ to the referee. Instead, the referee works with *fingerprints* of the neighborhood vectors, and we use modified equality queries to implement the tests in Lines 4, 7 and 10.

To simulate the updates performed in Lemma 14, we rely on the following crucial property: upon removing node w , we modify the representation $\{(a_v, b_v)\}_{v \in V}$ by adding a_w to some vectors and leaving the others unchanged. By induction on the number of updates, we therefore have:

► **Proposition 15.** *After performing $t \geq 0$ steps resulting in a partial decomposition $decomp$, the resulting representation $\{(a_v^t, b_v^t)\}_{v \in V}$ of the remaining graph can be written in the form*

$$a_v^t = e_v \oplus \bigoplus_{u \in A_v^t} e_u \quad \text{and} \quad b_v^t = \nu_v \oplus \bigoplus_{u \in B_v^t} e_u,$$

where $A_v^t, B_v^t \subseteq V$ depend only on $decomp$.

This is important because the referee can explicitly construct $\bigoplus_{u \in A_v^t} e_u$ and $\bigoplus_{u \in B_v^t} e_u$ and use them as modifying vectors inside modified equality queries, as we show next:

► **Theorem 16.** *In the NIH network model, there is a bounded-error quantum protocol with communication cost $O(k \log^2 k)$ that enables the referee to reconstruct a distance-hereditary graph, or reject if the input graph is not distance-hereditary.*

Proof. The protocol is described in pseudocode in Algorithm 2. It is convenient to slightly abuse the notation by writing $\text{MEQ}_{k,k}(i, y, z)$ to denote the query “ $x_i \oplus y = z$?”, where $i \in [k]$ and $y, z \in \{0, 1\}^k$. The referee can perform this query by computing a fingerprint for the vector 0^k and then proceeding as shown in Section 3, using the fingerprint for player i ’s input, the fingerprint for 0^k , and the vectors y, z .

In the protocol, the referee explicitly maintains the vectors $\{a_v\}_{v \in V}$ of the representation, and implicitly maintains the vectors $\{b_v\}_{v \in V}$ by storing *modifier vectors* $\{c_v\}_{v \in V}$, such that $b_v = \nu_v \oplus c_v$ for each $v \in V$ (this is possible due to Proposition 15). To simulate each test in Algorithm 1 we use appropriate modified equality queries: for example, to simulate the test “ $b_w = a_u$?” in line 4 of Algorithm 1, we use the query $\text{MEQ}_{k,k}(w, c_w, a_u)$ in line 4 of Algorithm 2, which checks whether $\nu_w \oplus c_w = a_u$. Since $b_w = \nu_w \oplus c_w$, this corresponds to exactly the same test.

■ **Algorithm 2** Quantum SMP Protocol for Computing a Pendant-Twin Decomposition.

```

1 Set  $a_v \leftarrow e_v, c_v \leftarrow 0^k$  for each  $v \in V$ 
2 Set  $decomp \leftarrow \lambda$  (an empty sequence)
3 while  $|V| \geq 2$  do
4   if  $\exists w, u$  such that  $\text{MEQ}_{k,k}(w, c_w, a_u) = 1$  then // Pendant  $w$  with neighbor  $u$ 
5     Append (“pendant”,  $w, u$ ) to  $decomp$  and set  $V \leftarrow V \setminus \{w\}$ 
6     Set  $c_u \leftarrow c_u \oplus a_w$ 
7   else if  $\exists w, v$  such that  $\text{MEQ}_{k,k}(w, u, c_w, c_u) = 1$  then // False twins  $w, u$ 
8     Append (“false twin”,  $w, u$ ) to  $decomp$  and set  $V \leftarrow V \setminus \{w\}$ 
9     Set  $a_u \leftarrow a_u \oplus a_w$ 
10  else if  $\exists w, u$  s.t.  $\text{MEQ}_{k,k}(w, u, c_w \oplus a_w, c_u \oplus a_u) = 1$  then // True twins  $w, u$ 
11    Append (“true twin”,  $w, u$ ) to  $decomp$  and set  $V \leftarrow V \setminus \{w\}$ 
12    Set  $a_u \leftarrow a_u \oplus a_w$  and  $c_u \leftarrow c_u \oplus a_w$ 
13  else Output “Graph is not distance-hereditary”
14 Output  $decomp$ 

```

Each of the tests in lines 4, 7 and 10 of Algorithm 2 can be implemented using $\binom{k}{2}$ modified equality queries. The other steps do not require any queries. The whole procedure can thus be implemented by an $\text{MEQ}_{k,k}$ decision tree of depth $(k-1)3\binom{k}{2}$; the quantum protocol can be constructed from Theorem 2. ◀

4.4 Enumeration of Isolated Cliques

Finally, we turn our attention to the problem of enumerating all *isolated cliques* in a graph. Isolated cliques and pseudocliques are important concepts in complex network analysis (see, e.g., [25, 24, 30]). Concretely, we show how to enumerate max- d -isolated cliques:

► **Definition 17** ([30], Definition 3). *A subgraph S of $G = (V, E)$ is a max- d -isolated clique if the subgraph induced by S is a clique, and each node in S has at most d edges to $V \setminus S$.*

It is convenient to describe our protocol for enumerating max- d -isolated cliques using a new primitive that we call *modified bounded Hamming distance queries*, which compute the Hamming distance between two modified inputs, but only if the distance does not exceed some fixed threshold d :

$$\text{MHAM}_n^d(i, j, y, z) = \begin{cases} \Delta_n(x_i \oplus y, x_j \oplus z) & \text{if } \Delta_n(x_i \oplus y, x_j \oplus z) \leq d, \\ \perp & \text{otherwise.} \end{cases}$$

An MHAM_n^d query can be computed by an $\text{MEQ}_{k,n}$ decision tree of depth $\sum_{c=0}^d \binom{n}{c} = O(n^d)$, which checks, for all strings $e \in \{0, 1\}^n$ of Hamming weight $c \leq d$, whether $(x_i \oplus y) \oplus e = x_j \oplus z$.⁶ We note that the restriction to a fixed upper bound d is important, as in general, computing the exact Hamming distance between two strings requires linear communication, even for interactive quantum protocols where the parties have shared entanglement [23].⁷

Our protocol for enumerating max- d -isolated cliques in the NIH network model will use modified bounded Hamming distance queries with $n = k$. The protocol is motivated by the following observations. The first observation allows us to use *bounded* Hamming distance queries, as it shows that we do not need to worry about nodes with $\Delta_k(\nu_u \oplus e_u, \nu_v \oplus e_v) > 2d$:

► **Proposition 18.** *If $S \subseteq V$ contains two nodes $u \neq v$ such that $\Delta_k(\nu_u \oplus e_u, \nu_v \oplus e_v) > 2d$, then S cannot be a max- d -isolated clique.*

A symmetric difference of cardinality $> 2d$ between $N(u) \cup \{u\}$ and $N(v) \cup \{v\}$ implies that either $N(u) \cup \{u\}$ contains more than d nodes that are not in $N(v) \cup \{v\}$, or vice-versa. If S is a clique, this means that either u or v has more than d neighbors outside S , so S is not a max- d -isolated clique; if S is not a clique, then in particular it is not a max- d -isolated clique.

The next observation is useful for checking whether a given set is a clique or not:

► **Proposition 19.** *For every $u, v \in V$, if $\{u, v\} \notin E$ then $\Delta_k(\nu_u, \nu_v) = \Delta_k(\nu_u \oplus e_u, \nu_v \oplus e_v) - 2$, and if $\{u, v\} \in E$ then $\Delta_k(\nu_u, \nu_v) = \Delta_k(\nu_u \oplus e_u, \nu_v \oplus e_v) + 2$.*

This is because if $\{u, v\} \notin E$, then $N(u) \ominus N(v) = ((N(u) \cup \{u\}) \ominus (N(v) \cup \{v\})) \setminus \{u, v\}$ (where \ominus denotes the symmetric difference), whereas if $\{u, v\} \in E$, then $N(u) \ominus N(v) = \{u, v\} \cup (N(u) \cup \{u\}) \ominus (N(v) \cup \{v\})$.

One implication of Proposition 19 is that we always have $\Delta_k(\nu_u, \nu_v) \leq \Delta_k(\nu_u \oplus e_u, \nu_v \oplus e_v) + 2$. Together with Proposition 18, this gives us an upper bound of $2d + 2$ on the Hamming distance $\Delta_k(\nu_u, \nu_v)$ for nodes that might belong to the same max- d -isolated clique.

Our main result for computing max- d -isolated cliques is as follows:

► **Theorem 20.** *In the NIH network model, for any $d \geq 1$, there is a bounded-error quantum SMP protocol with communication cost $O(kd \log^2 k)$ for enumerating all the max- d -isolated cliques of the input graph.*

Proof. We compute $\text{MHAM}_k^{2d+2}(u, v, 0^k, 0^k)$ and $\text{MHAM}_k^{2d}(u, v, e_u, e_v)$ for all pairs $u, v \in G$, using an MEQ_k decision tree of depth $\binom{k}{2} \cdot O(n^d) + \binom{k}{2} \cdot O(n^{d+2}) = O(k^2 n^{d+2})$ (this consists of $2\binom{k}{2}$ “small” MEQ decision trees, one for each MHAM query, composed with one another). In addition, we have each node send its degree to the referee.

⁶ Yao [42] showed that for any constant d , one can test whether $\Delta_n(x, y) \leq d$ using $O(\log n)$ qubits in the two-party quantum SMP model. However, unlike the equality function, it is not clear how to convert this protocol into a protocol for *modified* Hamming distance queries, as it does not have the linearity property that was crucial to prove Lemma 1.

⁷ Specifically, [23] shows that testing whether two n -bit strings have Hamming distance at most d requires $\Omega(d)$ bits of communication, for any $d \leq n/2$; this implies that computing the exact Hamming distance between general n -bit strings requires $\Omega(n)$ qubits.

We label each leaf of the decision tree based on the results of the MHAM queries leading to the leaf. Each subset $S \subseteq V$ is listed as a max- d -isolated clique in a given leaf if it satisfies the following three conditions:

1. $\text{MHAM}_k^{2d}(u, v, e_u, e_v) \neq \perp$ for all pairs $u, v \in S$. By Proposition 19, this also implies that $\text{MHAM}_k^{2d+2}(u, v, 0^k, 0^k) \neq \perp$ for all $u, v \in S$.
2. $\text{MHAM}_k^{2d+2}(u, v, 0^k, 0^k) = \text{MHAM}_k^{2d}(u, v, e_u, e_v) + 2$ for all pairs $u, v \in S$. And finally,
3. $\deg(u) \leq |S| + d - 1$ for all $u \in S$.

The correctness of this enumeration algorithm follows from the two observations above, together with the fact that if S is a clique, then a node $u \in S$ has at most d edges going outside S if and only if $\deg(u) \leq |S| + d - 1$. ◀

References

- 1 Scott Aaronson. Lecture notes for the 28th McGill invitational workshop on computational complexity. *arXiv preprint*, 2016. [arXiv:1607.05256](https://arxiv.org/abs/1607.05256).
- 2 Amirreza Akbari, Xavier Coiteux-Roy, Francesco d’Amore, François Le Gall, Henrik Lievonen, Darya Melnyk, Augusto Modanese, Shreyas Pai, Marc-Olivier Renou, Václav Rozhoň, and Jukka Suomela. Online locality meets distributed quantum computing. *ArXiv:2403.01903*, 2024. [doi:10.48550/arXiv.2403.01903](https://doi.org/10.48550/arXiv.2403.01903).
- 3 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. [doi:10.1145/3519270.3538413](https://doi.org/10.1145/3519270.3538413).
- 4 Heger Arfaoui and Pierre Fraigniaud. What can be computed without communications? *SIGACT News*, 45(3):82–104, 2014. [doi:10.1145/2670418.2670440](https://doi.org/10.1145/2670418.2670440).
- 5 László Babai and Peter G. Kimmel. Randomized simultaneous messages: Solution of a problem of Yao in communication complexity. In *Proceedings of the 12th Annual IEEE Conference on Computational Complexity (CCC 1997)*, pages 239–246, 1997. [doi:10.1109/CCC.1997.612319](https://doi.org/10.1109/CCC.1997.612319).
- 6 Hans-Jürgen Bandelt and Henry Martyn Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, 41(2):182–208, 1986. [doi:10.1016/0095-8956\(86\)90043-2](https://doi.org/10.1016/0095-8956(86)90043-2).
- 7 Adriano Barenco, André Berthiaume, David Deutsch, Artur Ekert, Richard Jozsa, and Chiara Macchiavello. Stabilization of quantum computations by symmetrization. *SIAM Journal on Computing*, 26(5):1541–1557, 1997. [doi:10.1137/S0097539796302452](https://doi.org/10.1137/S0097539796302452).
- 8 Florent Becker, Pedro Montealegre, Ivan Rapaport, and Ioan Todinca. The simultaneous number-in-hand communication model for networks: Private coins, public coins and determinism. In *Proceedings of the 21st International Colloquium on Structural Information and Communication Complexity (SIROCCO 2024)*, volume 8576 of *Lecture Notes in Computer Science*, pages 83–95. Springer, 2014. [doi:10.1007/978-3-319-09620-9_8](https://doi.org/10.1007/978-3-319-09620-9_8).
- 9 Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87:167902, 2001. [doi:10.1103/PhysRevLett.87.167902](https://doi.org/10.1103/PhysRevLett.87.167902).
- 10 Xavier Coiteux-Roy, Francesco d’Amore, Rishikesh Gajjala, Fabian Kuhn, François Le Gall, Henrik Lievonen, Augusto Modanese, Marc-Olivier Renou, Gustav Schmid, and Jukka Suomela. No distributed quantum advantage for approximate graph coloring. In *Proceedings of the 56th ACM Symposium on Theory of Computing (STOC 2024)*, pages 1901–1910, 2024. [doi:10.1145/3618260.3649679](https://doi.org/10.1145/3618260.3649679).
- 11 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](https://doi.org/10.1145/1412700.1412718).

- 12 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. doi:10.1145/2611462.2611488.
- 13 Orr Fischer, Rotem Oshman, and Uri Zwick. Public vs. private randomness in simultaneous multi-party communication complexity. *Theoretical Computer Science*, 810:72–81, 2020. doi:10.1016/j.tcs.2018.04.032.
- 14 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. doi:10.4230/LIPICs.ITCS.2021.28.
- 15 Pierre Fraigniaud, Mael Luce, Frédéric Magniez, and Ioan Todinca. Even-cycle detection in the randomized and quantum CONGEST model. In *Proceedings of the 43rd ACM Symposium on Principles of Distributed Computing (PODC 2024)*, pages 209–219, 2024. doi:10.1145/3662158.3662767.
- 16 Jingliang Gao. Quantum union bounds for sequential projective measurements. *Physical Review A*, 92:052331, 2015. doi:10.1103/PhysRevA.92.052331.
- 17 Dmitry Gavinsky, Tsuyoshi Ito, and Guoming Wang. Shared randomness and quantum communication in the multi-party model. In *Proceedings of the 28th Conference on Computational Complexity (CCC 2013)*, pages 34–43, 2013. doi:10.1109/CCC.2013.13.
- 18 Dmitry Gavinsky, Julia Kempe, and Ronald de Wolf. Quantum communication cannot simulate a public coin, 2004. ArXiv:quant-ph/0411051.
- 19 Dmitry Gavinsky and Pavel Pudlák. Exponential separation of quantum and classical non-interactive multi-party communication complexity. In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity (CCC 2008)*, pages 332–339, 2008. doi:10.1109/CCC.2008.27.
- 20 Cyril Gavoille, Adrian Kosowski, and Marcin Markiewicz. What can be observed locally? In *Proceedings of the 23rd International Symposium on Distributed Computing (DISC 2009)*, volume 5805 of *LNCS*, pages 243–257. Springer, 2009. doi:10.1007/978-3-642-04355-0_26.
- 21 Hipólito Gómez-Sousa. Multi-party quantum fingerprinting with weak coherent pulses: circuit design and protocol analysis. *New Journal of Physics*, 22:113004, 2020. doi:10.1088/1367-2630/abc2e5.
- 22 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 (PODC 2024)*, pages 220–230, 2024. doi:10.1145/3662158.3662788.
- 23 Wei Huang, Yaoyun Shi, Shengyu Zhang, and Yufan Zhu. The communication complexity of the Hamming distance problem. *Information Processing Letters*, 99(4):149–153, 2006. doi:10.1016/j.ipl.2006.01.014.
- 24 Hiro Ito and Kazuo Iwama. Enumeration of isolated cliques and pseudo-cliques. *ACM Transactions on Algorithms*, 5(4):40:1–40:21, 2009. doi:10.1145/1597036.1597044.
- 25 Hiro Ito, Kazuo Iwama, and Tsuyoshi Osumi. Linear-time enumeration of isolated cliques. In *Proceedings of the 13th Annual European Symposium (ESA 2005)*, volume 3669 of *Lecture Notes in Computer Science*, pages 119–130. Springer, 2005. doi:10.1007/11561071_13.
- 26 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. doi:10.1145/3293611.3331628.
- 27 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. doi:10.4230/LIPICs.STACS.2020.23.

- 28 Masaru Kada, Harumichi Nishimura, and Tomoyuki Yamakami. The efficiency of quantum identity testing of multiple states. *Journal of Physics A: Mathematical and Theoretical*, 41:395309, 2008. doi:10.1088/1751-8113/41/39/395309.
- 29 Jarkko Kari, Martín Matamala, Ivan Rapaport, and Ville Salo. Solving the induced subgraph problem in the randomized multiparty simultaneous messages model. In *Proceedings of the 22nd International Colloquium on Structural Information and Communication Complexity (SIROCCO 2015)*, volume 9439 of *Lecture Notes in Computer Science*, pages 370–384. Springer, 2015. doi:10.1007/978-3-319-25258-2_26.
- 30 Christian Komusiewicz, Falk Hüffner, Hannes Moser, and Rolf Niedermeier. Isolation concepts for efficiently enumerating dense subgraphs. *Theoretical Computer Science*, 410(38-40):3640–3654, 2009. doi:10.1016/j.tcs.2009.04.021.
- 31 Michael Lampis. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica*, 64(1):19–37, 2012. doi:10.1007/s00453-011-9554-x.
- 32 François Le Gall, Masayuki Miyamoto, and Harumichi Nishimura. Distributed Merlin-Arthur Synthesis of Quantum States and Its Applications. In *Proceedings of the 48th International Symposium on Mathematical Foundations of Computer Science (MFCS 2023)*, pages 63:1–63:15, 2023. doi:10.4230/LIPIcs.MFCS.2023.63.
- 33 François Le Gall and Frédéric Magniez. Sublinear-time quantum computation of the diameter in CONGEST networks. In *In Proceedings of the 37th ACM Symposium on Principles of Distributed Computing (PODC 2018)*, pages 337–346, 2018. URL: <https://dl.acm.org/citation.cfm?id=3212744>, doi:10.1145/3212734.3212744.
- 34 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 63:1–63:15, 2023. doi:10.4230/LIPIcs.STACS.2023.42.
- 35 François Le Gall, Harumichi Nishimura, and Ansis Rosmanis. Quantum advantage for the LOCAL model in distributed computing. In *Proceedings of the International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 49:1–49:14, 2019. doi:10.4230/LIPIcs.STACS.2019.49.
- 36 Pedro Montealegre, Sebastian Perez-Salazar, Ivan Rapaport, and Ioan Todinca. Graph reconstruction in the congested clique. *Journal of Computer and System Sciences*, 113:1–17, 2020. doi:10.1016/j.jcss.2020.04.004.
- 37 Ilan Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 1991. doi:10.1016/0020-0190(91)90157-D.
- 38 Ilan Newman and Mario Szegedy. Public vs. private coin flips in one round communication games (extended abstract). In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing (STOC 1996)*, pages 561–570, 1996. doi:10.1145/237814.238004.
- 39 Ji-Qian Qin, Jing-Tao Wang, Yun-Long Yu, and Xiang-Bin Wang. General theory of quantum fingerprinting network. *Physical Review Research*, 3:033039, 2021. doi:10.1103/PhysRevResearch.3.033039.
- 40 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.
- 41 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. doi:10.1145/3519270.3538441.
- 42 Andrew Chi-Chih Yao. On the power of quantum fingerprinting. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC 2003)*, pages 77–81, 2003. doi:10.1145/780542.780554.

A Illustration of a 2-Outcome Measurement

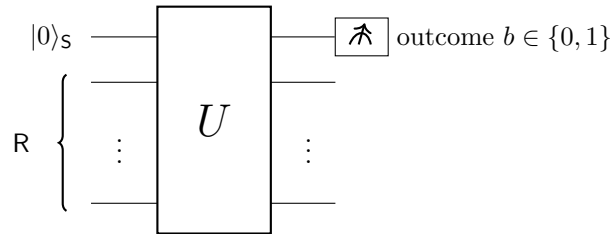


Figure 3 A 2-outcome measurement of a quantum register R .

B Projective measurements

Projective measurements are a special kind of measurements allowed by the laws of quantum mechanics. We explain below this concept and how to convert a 2-outcome measurement (as introduced in Section 2) into a 2-outcome projective measurement.

The 2-outcome projective measurement corresponding to the 2-outcome measurement associated with the unitary U is the measurement described in Figure 4. There are two key properties. First, for any $b \in \{0, 1\}$, the probability of obtaining b is exactly the same as the probability of obtaining b in the standard measurement associated with U . The second, and crucial, property is that for any $b \in \{0, 1\}$, if the probability of obtaining b in the 2-outcome projective measurement is very close to 1, then the postmeasurement state is very close to the initial state $|\psi\rangle$, which means that this state can be “reused” in later computation. In this paper, we will not need a formal statement of this second property. We will use instead (as a black box) the quantum union bound from Theorem 4.

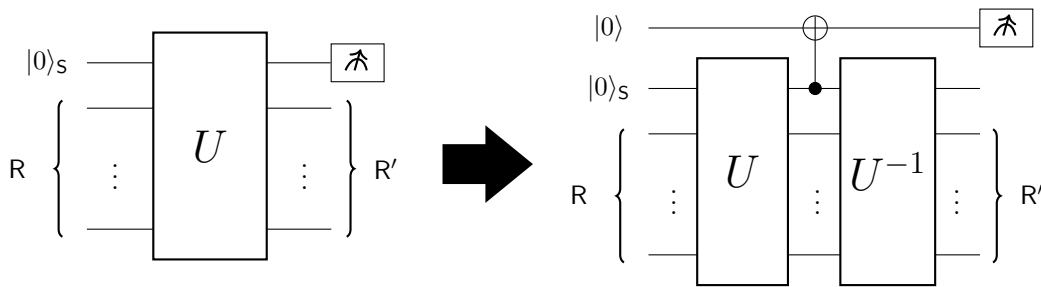


Figure 4 Conversion from a quantum circuit implementing a (non-projective) 2-outcome measurement associated with the unitary U (left) to a quantum circuit implementing a 2-outcome projective measurement (right). Register R stores the initial state and Register R' stores the postmeasurement state. In the right picture, the 2-qubit gate between U and U^{-1} represents the CNOT gate, where the X gate (also called NOT gate) is applied on the \oplus -part conditioned on the content of the black-circle part being 1.

C Description of the SWAP test

In this appendix we give the technical description of the SWAP test presented in Section 2: we describe the test in Figure 5 and give the corresponding quantum circuit in Figure 6.

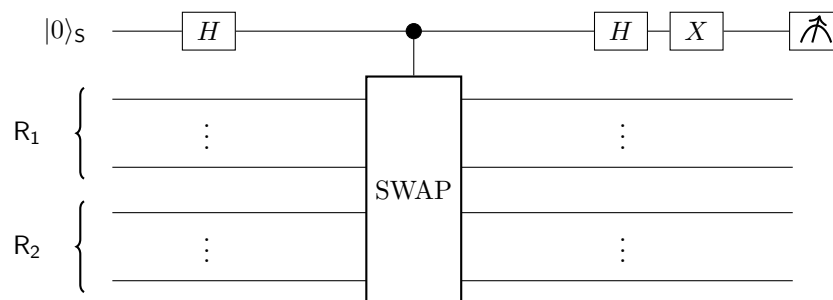
34:20 Quantum Simultaneous Protocols Using Equality Queries

SWAP test

Input: two quantum states in Registers R_1 and R_2 , respectively

1. Introduce a 1-register S initialized to $|0\rangle_S$.
2. Apply the Hadamard gate H on S .
3. (Controlled SWAP) If the content of S is 1, swap R_1 and R_2 .
4. Apply the Hadamard gate H and then the X gate on S .
5. Measure Register S in the computational basis and output the outcome.

■ **Figure 5** Description of the SWAP test.



■ **Figure 6** Quantum circuit for the SWAP test.