Quantum Advantage for the LOCAL Model in Distributed Computing

François Le Gall  
Graduate School of Informatics, Kyoto University  
Yoshida-Hommachi, Sakyo-ku, Kyoto 606-8501, Japan

Harumichi Nishimura  
Graduate School of Informatics, Nagoya University  
Chikusa-ku, Nagoya, Aichi 464-8601, Japan

Ansis Rosmanis  
Centre for Quantum Technologies, National University of Singapore  
Block S15, 3 Science Drive 2, 117543, Singapore

Abstract

There are two central models considered in (fault-free synchronous) distributed computing: the CONGEST model, in which communication channels have limited bandwidth, and the LOCAL model, in which communication channels have unlimited bandwidth. Very recently, Le Gall and Magniez (PODC 2018) showed the superiority of quantum distributed computing over classical distributed computing in the CONGEST model. In this work we show the superiority of quantum distributed computing in the LOCAL model: we exhibit a computational task that can be solved in a constant number of rounds in the quantum setting but requires $\Omega(n)$ rounds in the classical setting, where $n$ denotes the size of the network.

2012 ACM Subject Classification  Theory of computation → Quantum computation theory

Keywords and phrases  Quantum computing, distributed computing, LOCAL model

1 Introduction

Classical distributed computing. A central topic in distributed computing is the study of synchronous network algorithms. Here processors and communication channels are modeled using nodes and edges, respectively, and executions proceed with round-based synchrony, where each node can transfer one message to each adjacent node per round. The main quantity of interest is typically the number of rounds needed to solve a computational task. Two fundamental models considered in the literature are the LOCAL model, introduced by Linial [19, 20], and the CONGEST model, introduced by Peleg [22].

The LOCAL model does not put any limitation on the size of the messages sent at each round, and thus mainly characterizes the locality of the problem considered and the hardness of breaking symmetry between nodes. Obviously all computational problems can be solved with $O(D)$ rounds in the LOCAL model, where $D$ is the diameter of the network, by first collecting all the information about the network (including the inputs of all nodes) at some node. Many important problems have significantly more efficient algorithms—we refer to [22] for examples and to [15] for a recent classification.

In the CONGEST model, on the other hand, each message has restricted length (the length is typically restricted to $O(\log n)$ bits, where $n$ is the number of nodes in the network). This corresponds to the situation of communication channels with limited bandwidth, in which case congestions can arise. A simple example showing the striking difference between these two models is deciding whether the diameter of the network is 2 or 3. This problem
Quantum Advantage for the LOCAL Model in Distributed Computing

requires $\Theta(n)$ rounds in the CONGEST model \cite{11 16 23}, while in the LOCAL model it can be trivially solved with a constant number of rounds.

Quantum distributed computing. Quantum versions of both models can be naturally defined by replacing classical channels by quantum channels between the processors (which are now quantum processors, i.e., processors that can process quantum information). Gavoille, Kosowski and Markiewicz \cite{12} first considered quantum distributed computing in the LOCAL model, and showed that for several fundamental problems, such as Graph Coloring or Maximal Independent Set, allowing quantum communication cannot lead to any significant advantage. More recently, Arfaoui and Fraigniaud \cite{2} observed that several lower bound techniques for the classical LOCAL model hold in the quantum model as well.

The power of distributed network computation in the CONGEST model, where each node can send $O(\log n)$ qubits per round to each neighbor, has been first investigated by Elkin, Klauck, Nanongkai and Pandurangan \cite{10}. The main conclusions reached in that paper were that for many fundamental problems in distributed computing, such as computing minimum spanning trees or minimum cuts, quantum communication does not, again, offer significant advantages over classical communication. Recently, Le Gall and Magniez nevertheless showed the superiority of quantum distributed computing in the CONGEST model for a concrete problem \cite{18}: they showed that the diameter of the network can be computed in $\tilde{O}(\sqrt{nD})$ rounds in the quantum setting, where $n$ is the number of nodes of the network and $D$ is the diameter of the network. In comparison, as mentioned above $\Omega(n)$ rounds are necessary in the classical setting even if $D$ is constant. It should be mentioned that from a purely complexity-theoretic perspective most known separations between two-party classical and quantum communication complexities (e.g., separations in the bounded-error setting for the disjointness function \cite{1 7 17}) can be converted in a straightforward way into similar separations in the CONGEST model. The contribution of \cite{18} is actually to give a separation for an important problem in distributed computing.

A pressing open question is to understand whether a similar quantum speedup in distributed computing can be obtained in the LOCAL model. The only known gap is a factor 2: for each integer $t \geq 1$, Gavoille, Kosowski and Markiewicz \cite{12} constructed a computational problem (inspired by the work by Greenberger, Horne and Zeilinger \cite{13}) that can be solved in $t$ rounds in the quantum setting but requires $2t$ rounds in the classical setting. The quantum upper bound comes from the observation that $t$ rounds are enough to create entanglement between two nodes at distance $2t$ from each other. In this perspective, as mentioned in \cite{12}, the speed-up factor of 2 may “look like a natural limit”. Note that, contrary to the CONGEST model, known separations between two-party (or multiparty) quantum and classical communication complexities seem meaningless to prove separations in the LOCAL model due to the unlimited bandwidth between nodes.

Our results. In this work we show the existence of a large gap between the round complexities of quantum and classical distributed computation in the LOCAL model.

\begin{itemize}
  \item Theorem 1. There exists a computational problem that can be solved with 2 rounds in the quantum LOCAL model, but requires $\Omega(n)$ rounds in the classical LOCAL model, where $n$ denotes the number of nodes in the network. The classical lower bound holds even if arbitrary prior randomness is allowed.
\end{itemize}

\footnote{A much larger gap is shown in \cite{12} for the setting where the nodes of the network initially share a globally entangled state. In the present paper, however, we consider the arguably more natural setting where no prior entanglement is allowed.}
The computational problem we construct to prove Theorem 1 is inspired by a construction from [3], which was initially used to show the non-locality of measurement outcomes of graph states. The same construction was recently also used by Bravyi, Gosset and König [5] to prove their separation between quantum and classical constant-depth circuit complexities. The problem, defined in Section 4, can be informally described as follows: on an $n$-node ring, the nodes should output one of the possible outcomes that arise when measuring the graph state corresponding to the ring in a basis depending on the input each node receives. We are currently not aware of any applications of Theorem 1 for constructing quantum algorithms for problems of interest to the distributed computing community, but nevertheless consider this result as a valuable proof of concept showing that the quantum LOCAL model can be arbitrarily more powerful than the classical LOCAL model.

The computational problem considered in Theorem 1 is a relation (i.e., for each input there are multiple valid outputs). It is fairly easy to show that for any function (i.e., for each input there is only one valid output at each node) the quantum and classical round complexities are equal in the LOCAL model: we give a proof of this property in Appendix B.

We then investigate whether a separation similar to the separation of Theorem 1 can be obtained for a computational problem without input. Such kinds of computational problems (seen as sampling problems or computations of probability distributions) are the main targets of the field of quantum supremacy (see [14] for a recent survey). Indeed, a major open problem left in the work by Bravyi, Gosset and König [5] mentioned above is to prove the superiority of constant-depth quantum circuits for the computation of a probability distribution. We show that in the LOCAL model of distributed computing such a goal can be achieved.

**Theorem 2.** There exists a sampling problem that can be solved with 2 rounds in the quantum LOCAL model, but requires $\Omega(n)$ rounds in the classical LOCAL model. The classical lower bound holds even for constant-error additive approximation.

Theorem 2 is proved by considering the same computational problem as used in Theorem 1 but replacing the inputs by random bits. The proof nevertheless requires several adjustments, in particular a careful analysis of the classical randomness shared during the execution of the protocol.

**Other relevant works.** It is well known that quantum communication can offer significant advantages over classical communication in several settings such as communication complexity or quantum games (see, e.g., [6, 9, 25]). Concerning problems of interest to the distributed computing community, the main works not already mentioned are quantum algorithms for byzantine agreements [4] and for distributed computing over anonymous networks, and in particular the design of zero-error quantum algorithms for leader election [24] (see also [9]).

# 2 Preliminaries

## 2.1 Notations and definitions

**Quantum gates.** We assume that the reader is familiar with the basis of quantum computation and refer to [21] for a standard reference. We will use the Hadamard gate $H$ and the phase gate $S$ acting on one qubit:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix},$$
where \( i \) denotes the imaginary unit of complex numbers. We will also use the CNOT gate acting on two qubits (called the control qubit and the target qubit) that maps the basis state \( |a \rangle |b \rangle \), for any two bits \( a, b \in \{0, 1\} \), to the state \( |a \rangle |a \oplus b \rangle \) where \( \oplus \) denotes the exclusive OR. Finally, we will need the following two 2-qubit gates (Controlled-Z and Controlled-S gates):

\[
\begin{align*}
\text{CZ} &= \begin{pmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & -1 \end{pmatrix}, \\
\text{CS} &= \begin{pmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & i \end{pmatrix}.
\end{align*}
\]

**Graph-theoretic notation.** In this work all the graphs will be undirected and unweighted. For any graph \( G = (V, E) \) and any node \( u \in V \), we use \( N(u) \) to denote the set of neighbors of \( u \).

**Graph states.** Graph states are a special type of quantum states that are associated with graphs [15]. Let \( G = (V, E) \) be any undirected graph. The graph state associated with \( G \) is the quantum state on \( |V| \) qubits constructed in the following way. Let \( \{Q_u\}_{u \in V} \) denote the \( |V| \) registers used to store the qubits of the graph state (each register stores one qubit). First construct the quantum state

\[
\bigotimes_{u \in V} |0\rangle_{Q_u}
\]

in these registers. Then apply a Hadamard gate on each register. Finally, for each edge \( \{u, v\} \in E \), apply the gate CZ on the couple of registers \((Q_u, Q_v)\).

**The total variation distance.** Given two probability distributions \( p, q : X \to [0, 1] \) over a finite set \( X \), the total variation distance (also called statistical distance) between \( p \) and \( q \) is defined as

\[
\frac{1}{2} \sum_{x \in X} |p(x) - q(x)|.
\]

## 2.2 Classical and quantum LOCAL models

In this paper we consider the LOCAL communication model in both the classical and quantum scenarios. The topology of the network is represented by a graph. Executions proceed with round-based synchrony and each node can transfer one message to each adjacent node per round. Initially the nodes of the network share neither any randomness nor, in the quantum scenario, any entanglement. In this paper all the networks are undirected and unweighted. All links and nodes of the network (corresponding to the edges and nodes of the graph, respectively) are reliable and suffer no faults. Each node has a distinct identifier. Initially, each node knows nothing about the topology of the network except the set of edges incident to itself and the number of nodes of the graph.

The processors at each node operate probabilistically in the classical LOCAL model, and they operate quantumly in the quantum LOCAL model. The messages exchanged between them are, respectively, classical and quantum. In this paper we do not consider the running time of the processors, as we are only interested in the round complexity. While the classical lower bound of Theorem 2 is proved using a relatively informal definition of the classical LOCAL model, we include its formal definition in Appendix A for completeness.

\[2\] The classical lower bound of our first result (Theorem 6) actually holds even if the nodes of the network initially share arbitrary randomness.
2.3 The construction from prior works

We now describe the construction introduced in [3], and also used in [5], that shows that non-locality can arise when measuring graph states. For any even integer \( d \geq 2 \), we define the graph \( G_d \) as a ring consisting of \( 3d \) nodes, and denote the nodes \( v_0, v_1, \ldots, v_{3d-1} \) (see Figure 1). It will be convenient to consider this graph as a triangle, with the three nodes \( v_0, v_{2d} \) and \( v_{2d} \) as corners. We define \( V_R = \{ v_i \mid i \in \{1, \ldots, d-1\} \} \), \( V_B = \{ v_i \mid i \in \{d+1, \ldots, 2d-1\} \} \) and \( V_L = \{ v_i \mid i \in \{2d+1, \ldots, 3d-1\} \} \) as the set of nodes on the right side, bottom side and left side, respectively, of the triangle. We also define \( V_{\text{even}} \) as the set of all nodes of the graph with even label, and \( V_{\text{odd}} \) as the set of all nodes with odd label.

\[ v_0 \quad v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \quad v_7 \quad v_8 \quad v_9 \quad v_{10} \quad v_{11} \]

\[ V_L \quad V_{\text{even}} \quad V_R \]

\[ m_E = \bigoplus_{v \in V_{\text{even}}} m_v, \quad m_R = \bigoplus_{v \in V_R \cap V_{\text{odd}}} m_v, \]

\[ m_B = \bigoplus_{v \in V_B \cap V_{\text{odd}}} m_v, \quad m_L = \bigoplus_{v \in V_L \cap V_{\text{odd}}} m_v. \]

Refs. [3, 5] characterized which combinations of these four bits can arise as an outcome of the process \( \mathcal{P}(b_0, b_1, b_2) \):

**Proposition 3.** ([3, 5]) For any bits \( b_0, b_1, b_2 \) and any measurement outcome of the process
\(\mathcal{P}_d(b_0, b_1, b_2)\), the identity \(m_R \oplus m_B \oplus m_L = 0\) holds. Additionally, we have:

\[
\begin{align*}
    m_E &= 0 & \text{if } (b_0, b_1, b_2) &= (0, 0, 0), \\
    m_E \oplus m_R \oplus m_L &= 1 & \text{if } (b_0, b_1, b_2) &= (0, 1, 1), \\
    m_E \oplus m_R \oplus m_B &= 1 & \text{if } (b_0, b_1, b_2) &= (1, 0, 1), \\
    m_E \oplus m_B \oplus m_L &= 1 & \text{if } (b_0, b_1, b_2) &= (1, 1, 0).
\end{align*}
\]

It will be convenient to represent a measurement outcome \(m_v\) as the binary string \(m \in \{0,1\}^{3d}\) where the \(i\)-th bit is \(m_i\), for each \(i \in \{0, \ldots, 3d - 1\}\). We define the support of the process \(\mathcal{P}_d(b_0, b_1, b_2)\), and denote it \(\Lambda_d(b_0, b_1, b_2)\), as the set of all binary strings in \(\{0,1\}^{3d}\) corresponding to measurement outcomes arising (with non-zero probability) from the process \(\mathcal{P}_d(b_0, b_1, b_2)\).

Finally, our lower bounds will rely on the following lemma, which essentially shows that the quantum correlations from the process \(\mathcal{P}_d(b_0, b_1, b_2)\) cannot be simulated classically by local affine functions.

\[\textbf{Lemma 4.} (E, G)\] Consider any affine function \(q_E: \{0,1\}^2 \rightarrow \{0,1\}\) and any three affine functions \(q_R: \{0,1\}^2 \rightarrow \{0,1\}\), \(q_L: \{0,1\}^2 \rightarrow \{0,1\}\), \(q_B: \{0,1\}^2 \rightarrow \{0,1\}\) such that

\[q_E(b_1, b_2) \oplus q_B(b_1, b_2) \oplus q_L(b_0, b_2) = 0\]

holds for any \((b_1, b_2, b_3) \in \{0,1\}^3\). Then at least one of the four following equalities does not hold:

\[
\begin{align*}
    q_E(0,0,0) &= 0, \\
    q_E(0,1,1) \oplus q_R(0,1) \oplus q_L(0,1) &= 1, \\
    q_E(1,0,1) \oplus q_R(1,0) \oplus q_B(0,1) &= 1, \\
    q_E(1,1,0) \oplus q_B(1,0) \oplus q_L(1,0) &= 1.
\end{align*}
\]

\section{Efficient Construction of Graph States}

In this section we consider the construction of graph states in the distributed setting. More precisely, we consider the following problem that we call the \textit{subgraph state construction problem}. The problem is defined on an arbitrary network \(G = (V, E)\). Each node \(v \in V\) receives a bit \(c_v \in \{0, 1\}\) as input. Let \(G' = (V', E')\) denote the subgraph of \(G\) induced by the node set \(V' = \{v \in V \mid c_v = 1\}\). The problem asks to create the graph state corresponding to \(G'\), shared over the nodes in \(V'\): each node \(v \in V'\) of the network should own the corresponding 1-qubit register of the graph state (which is Register \(Q_v\) in the notations of Section 2.2).

The following theorem shows that this problem can be done efficiently.

\[\textbf{Theorem 5.} \textit{In the quantum LOCAL model, the subgraph state construction problem can be solved in 2 rounds.}\]

\[\textbf{Proof.}\] The protocol is presented in Figure 3 and illustrated, for a path of two nodes, in Figure 4. This is clearly a 2-round protocol: one round is used at Step 1(c) and one round is used at Step 2(b).

We now prove that the protocol is correct. At the end of Step 1(b), the state of the whole network is:

\[
|\varphi\rangle = \bigotimes_{v \in V} \left( \frac{1}{\sqrt{2}} \sum_{j=0}^{1} |j\rangle_{Q_v} \bigotimes_{e \in N(v)} |j\rangle_{R_e} \right).
\]
Input: each node $u \in V$ receives a bit $c_u$

1. Each node $u \in V$ does the following:
   (a) it prepares one 1-qubit register $Q_u$ and, for each neighbor $v \in N(u)$, one 1-qubit register denoted $R^u_v$ (all these registers are initialized to the quantum state $|0\rangle$);
   (b) it applies a Hadamard gate on $Q_u$, and then a CNOT gate on $(Q_u, R^u_v)$ with $Q_u$ as control qubit, for each $v \in N(u)$;
   (c) it sends, for each $v \in N(u)$, the register $R^u_v$ and the bit $c_u$ to node $v$.

2. Each node $u \in V$ (who now owns the registers $Q_u$ and the registers $R^u_v$ just received) does the following:
   (a) it applies the gate CS to the pair of registers $(Q_u, R^u_v)$ for each $v \in N(u)$ such that $c_u \wedge c_v = 1$;
   (b) it sends back Register $R^u_v$ to node $v$, for each $v \in N(u)$.

3. Each node $u \in V$ (who now owns the registers $Q_u$ and the registers $R^u_v$) does the following:
   (a) it applies a CNOT gate on $(Q_u, R^u_v)$ with $Q_u$ as control qubit, for each $v \in N(u)$;
   (b) it disregards the registers $R^u_v$ for all $v \in N(u)$.

Figure 3 The quantum distributed algorithm solving the subgraph state construction problem.

Let us fix any two nodes $u$ and $v$ such that $\{u, v\} \in E$. The state $|\varphi\rangle$ can be rewritten as

$$\frac{1}{2} \sum_{j=0}^{1} \sum_{k=0}^{1} |j\rangle_{Q_u} |k\rangle_{Q_v} |j\rangle_{R^u_v} |k\rangle_{R^v_u} |\psi_{j,u,k,v}\rangle$$

where

$$|\psi_{j,u,k,v}\rangle = \bigotimes_{v' \in N(u) \setminus \{v\}} |j\rangle_{R^u_{v'}} \bigotimes_{w' \in N(v) \setminus \{u\}} |k\rangle_{R^v_{w'}}.$$

If node $u$ applies the gate CS to the pair of registers $(Q_u, R^u_v)$ and node $v$ applies the gate CS to the pair of registers $(Q_v, R^v_u)$, then the state $|\varphi\rangle$ is mapped to the quantum state

$$\text{CZ}(Q_u, Q_v) |\varphi\rangle,$$

where $\text{CZ}(Q_u, Q_v)$ denotes the gate CZ applied to the pair of registers $(Q_u, Q_v)$. Since $c_u \wedge c_v = 1$ if and only if $\{u, v\} \in E'$, at the end of Step 2, the whole state of the network is

$$\left( \prod_{(u,v) \in E'} \text{CZ}(Q_u, Q_v) \right) |\varphi\rangle.$$

Since Step 3 simply disentangles and then disregards the registers $R^u_v$ for all $(u, v) \in V \times V$, at the end of Step 3 we obtain the desired graph state shared by the nodes in $V'$.

4 Separation between the Classical and Quantum LOCAL Models

In this section we prove Theorem $\Box$
For any even integer \( d \geq 2 \), remember the network \( G_d = (V, E) \) defined in Section 2.3 where \( V = \{v_i, \ldots, v_{3d-1}\} \). In this section we will consider the network \( Q_d \) with node set \( V \cup \{w_0, w_1, w_2\} \) and edge set \( E \cup \{\{v_0, w_0\}, \{v_{3d}, w_1\}, \{v_{2d}, w_2\}\} \), which is obtained from \( G_d \) by adding one node to each corner (see Figure 5).

We now describe the computational problem used to prove our separation. The network considered is \( Q_d \), for any even integer \( d \geq 2 \). The input consists of three bits \( b_0, b_1 \) and \( b_2 \): node \( w_0 \) is given \( b_0 \), node \( w_1 \) is given \( b_1 \), and node \( w_2 \) is given \( b_2 \) (the other nodes have no input). The output is defined as follows: for each \( i \in \{0, 1, \ldots, 3d-1\} \), the node \( v_i \) should output one bit \( x_i \). The nodes \( w_0 \), \( w_1 \) and \( w_2 \) do not output anything. The output can thus be seen as a binary string \( (x_0, \ldots, x_{3d-1}) \) of length \( 3d \). We say that this string is valid if it is in the set \( \Lambda_d(b_0, b_1, b_2) \).

The following theorem shows an upper bound on the complexity of this problem in the quantum LOCAL model and a lower bound in the classical LOCAL model.

\[ \textbf{Theorem 6.} \text{ There exists a 2-round quantum algorithm that always outputs a valid string.} \]
For any integer $T \leq d/2$, no $T$-round classical algorithm can output a valid string with probability greater than $7/8$ on all inputs $(b_0, b_1, b_2) \in \{0, 1\}^3$, even if arbitrary prior randomness is allowed.

**Proof.** The considered computational problem can easily be solved in two rounds in the quantum setting by implementing the following process.

- **Process 1.** The nodes of the network first apply the 2-round algorithm of Theorem 6 with input $c_{w_0} = c_{w_1} = c_{w} = 0$ for each $v \in V$. This constructs the graph state over the subgraph $G_d$ of $\mathcal{G}_d$. Moreover, for each $i \in \{0, 1, 2\}$, the node $w_i$ concurrently sends its input $b_i$ to its neighbor $v_{di}$ (the messages can be appended to the messages of the algorithm of Theorem 3). Finally, the nodes of $V$ implement Steps 2–4 of the process $\mathcal{P}_d(b_0, b_1, b_2)$, which can be done without communication, and output their measurement outcomes.

Note that implementing Process 1 requires each node to know whether it is an input node ($w_0, w_1$ or $w_2$), a corner node on the ring ($v_0, v_d$ or $v_{2d}$) or a non-corner node on the ring (all the other nodes). This is not a problem since each node knows its degree and the type of the nodes depends only on their degrees: the nodes $w_0, w_1$ and $w_2$ are the nodes of degree 1, the nodes $v_0, v_d$ and $v_{2d}$ are the nodes of degree 3, and all the other nodes have degree 2.

We now show the classical lower bound, which uses the same argument as in [3] and holds even if the nodes of the network share prior randomness. Consider any classical distributed algorithm $A$ and fix its randomness $r$ (the string $r$ represents both the shared prior randomness and the random bits used by the algorithm). This defines a deterministic algorithm that we denote $A(r)$. Let us write $q_v(b_0, b_1, b_2)$ the bit output at node $v$ by $A(r)$, for each $v \in V$. Let us define

$$
q_E(b_0, b_1, b_2) = \sum_{v \in V_{even}} q_v(b_0, b_1, b_2), \quad q_R(b_0, b_1, b_2) = \sum_{v \in V_R \cap V_{odd}} q_v(b_0, b_1, b_2),
$$

$$
q_B(b_0, b_1, b_2) = \sum_{v \in V_B \cap V_{odd}} q_v(b_0, b_1, b_2), \quad q_L(b_0, b_1, b_2) = \sum_{v \in V_L \cap V_{odd}} q_v(b_0, b_1, b_2).
$$

Assume that the algorithm uses at most $d/2$ rounds. Then $q_E, q_R, q_B$ and $q_L$ are affine functions of $b_0, b_1, b_2$. Moreover, $q_R$ can only depend on $b_0$ and $b_1$, $q_B$ can only depend on $b_1$ and $b_2$, and $q_L$ can only depend on $b_0$ and $b_2$. From Proposition 3 and Lemma 4 we get that, at least for one choice of $(b_0, b_1, b_2) \in \{0, 1\}^3$, the output of $A(r)$ is not a valid string (i.e., does not correspond to a possible measurement outcome of the process $\mathcal{P}_d(b_0, b_1, b_2)$). A simple counting argument then shows that there exists at least one choice of $(b_0, b_1, b_2)$ for which the original randomized protocol $A$ fails to output a valid string with probability at least $1/8$.

We are now ready to prove Theorem 1.

**Proof of Theorem 1.** Theorem 6 implies that any classical algorithm that outputs a valid string with probability greater than $7/8$ requires a number of rounds linear in the size of the network (since $d$ is a linear function of the size of network $\mathcal{G}_d$).

We now show how to reduce the success probability from $7/8$ to an arbitrary small value: for any constant $\varepsilon > 0$ we construct a new computational problem, which can still be solved in two rounds in the quantum setting, such that any classical algorithm solving this problem with probability at least $\varepsilon$ requires a number of rounds linear in the size of the network. Let $k$ be an integer. The problem considered is simply $k$ independent copies of the problem considered so far: the network considered has $3k(d+1)$ nodes and consists of $k$ copies of the network $\mathcal{G}_d$. Each copy receives three bits and outputs a string of $3d$ bits. The output of
the whole network is correct if the strings output by each copy are all valid. This problem can obviously be solved using two rounds in the quantum setting by constructing the graph state over the whole network. Theorem 6 implies that for any integer $T \leq d/2$, no $T$-round classical algorithm can give a correct output with probability greater than $(7/8)^{k}$ on all inputs, even if arbitrary prior randomness is allowed. Setting $k = \Theta(\log(1/\varepsilon))$ concludes the proof.

5 Separation for a Distribution

In this section we prove Theorem 2. The idea is to convert the relation of the previous section into a distribution by requiring that each input is taken uniformly at random (and requiring that the three nodes with an input output their inputs as well).

Recall Process 1 in the proof of Theorem 6. There, the actions of every node of $G$ depend only on the degree of the node, namely, whether its degree is 1, 2 or 3. The same is true for the 2-round sampling protocol in the quantum LOCAL model described below, which also uses the same network $G$. Therefore, for notational convenience, let us assume that every node knows its global location in $G$.

Consider the probability distribution $\Gamma_{d}$ generated by the following 2-round quantum protocol. First, for each $i \in \{0, 1, 2\}$, the node $w_{i}$ chooses an unbiased random bit $b_{i}$. Then Process 1 is implemented, at the end of which, as specified, no descriptors each return one bit. Meanwhile, the nodes $w_{0}, w_{1}, w_{2}$ output, respectively, $b_{0}, b_{1}, b_{2}$.

Theorem 2 immediately follows from the following result.

Theorem 7. Every $T \leq d/4$ round algorithm on $G_{d}$ in the classical LOCAL model generates a probability distribution that is at least $1/11$ away from $\Gamma_{d}$ in the total variation distance.

Proof. The proof proceeds as follows. Starting from the classical LOCAL model, we present a series of increasingly powerful models on the network $G_{d}$. Each model receives no input and returns one bit per node. Then we show that the last, the most powerful among these models cannot generate a probability distribution that is closer than distance $1/11$ away from $\Gamma_{d}$.

Consider the classical LOCAL model on the network $G_{d}$. We assume that the randomness of each node stems from a finite random bit string that it receives as an input, and all further operations of the node are deterministic (see Appendix A.1 for technical details). We now present a series of steps, where each step either strengthens the model or maintains its power while making it easier to analyze.

1. We assume that all the nodes know their location in the global topology.
2. We allow certain nodes to share randomness. In particular, for each $i \in \{0, 1, 2\}$, let $V_{i}$ be the set consisting of $w_{i}$ and all the nodes $u \in V$ at distance at most $T$ away from $w_{i}$. And let $V_{\perp} = V \setminus (V_{0} \cup V_{1} \cup V_{2})$. We assume that, for $i \in \{0, 1, 2, \perp\}$, all nodes within $V_{i}$ share randomness, namely, they all start with the same random string $Q_{i}$, which we think of as a random variable.

Here it is worth pausing the steps to note that, in a $T$-round protocol, the bit $b_{i}$ output by the node $w_{i}$ depends only on $Q_{i}$, thus we may write it as a function $b_{i}(Q_{i})$. Let $p_{i}$ be the probability that $b_{i} = 1$. If there exists $i \in \{0, 1, 2\}$ with $p_{i} \notin [5/11, 6/11]$, then the marginal distribution over $b_{i}$ is already at total variation distance greater than $1/11$ away from the corresponding marginal distribution in $\Gamma_{d}$, and the whole distributions ($\Gamma_{d}$ and the one generated by the classical protocol) can be only even farther apart. Thus let us
assume that $p_i \in [5/11, 6/11]$ for all $i \in \{0, 1, 2\}$. Since $Q_0, Q_1, Q_2$ are independent, each $(b_0, b_1, b_2) \in \{0, 1\}^3$ is output with probability at least $(5/11)^3 > 1/11$.

3. For $i \in \{0, 1, 2\}$, let $B_i$ be a random variable that takes value 1 with probability $p_i$ and value 0 with probability $1 - p_i$. For both $\beta \in \{0, 1\}$, let $Q_\beta^i$ be a random variable that equals each value $q$ of $Q_i$ such that $b_i(q) = \beta$ with probability $\Pr[Q_i = q]/\Pr[B_i = \beta]$. We replace the shared randomness $Q_i$ by $(Q_0^i, Q_1^i, B_i)$—each of the three variables being independent—with an extra requirement that the node $w_i$ always outputs $B_i$. This is clearly without loss of power, because we can recover $Q_i$ as $Q_0^i$, for which $b_i(Q_i) = B_i$.

4. We share all the randomness except $B_0, B_1, B_2$ among all the nodes. More precisely, we assume that all nodes start with the randomness $r = (Q_\perp^0, Q_0^1, Q_0^3, Q_0^2, Q_1^3, Q_2^3)$. In addition, for each $i \in \{0, 1, 2\}$, nodes in $V_i$ start with an additional random bit $B_i$ and we preserve the requirement that $w_i$ must output $B_i$.

Now we need to show that the final model cannot generate a probability distribution that is closer than total variation distance 1/11 away from $\Gamma_d$. Note that, at the beginning of the protocol, the value $B_i$ is only known to nodes at distance at most $T - 1$ away from $v_{d,i}$, and, after the protocol, it can be known only to nodes at distance $2T - 1 < d/2$ away from $v_{d,i}$. In particular, at the end of the protocol, each node of the network will know no more than one of the values $B_0, B_1, B_2$. All other communicated information is useless, as, aside from $B_0, B_1, B_2$, all other randomness is global.

The remainder of the proof is almost equivalent to that of the classical lower bound in Theorem 6 with the sole difference of the counting argument: instead of each choice of $(b_0, b_1, b_2)$ being given with probability exactly $1/8$, now each choice of $(b_0, b_1, b_2)$ is given with probability at least $1/11$.

Acknowledgments

The authors are grateful to Richard Cleve, Hirotada Kobayashi and Tomoyuki Morimae for helpful comments and discussions. FLG was partially supported by the JSPS KAKENHI grants No. 15H01677, No. 16H01705 and No. 16H05853. HN was partially supported by the JSPS KAKENHI grants No. 26247016, No. 16H01705 and No. 16K00015. AR was partially supported by the Singapore Ministry of Education and the National Research Foundation under grant R-710-000-012-135. Part of this work was done while AR was visiting Kyoto University, and AR would like to thank FLG for hospitality.

References

1. Scott Aaronson and Andris Ambainis. Quantum search of spatial regions. *Theory of Computing*, 1(1):47–79, 2005. [doi:10.4086/toc.2005.v001a004]
2. Heger Arfaoui and Pierre Fraigniaud. What can be computed without communications? *SIGACT News*, 45(3):82–104, 2014. [doi:10.1145/2670418.2670440]
3. Jonathan Barrett, Carlton M. Caves, Bryan Eastin, Matthew B. Elliott, and Stefano Pironio. Modeling Pauli measurements on graph states with nearest-neighbor classical communication. *Physical Review A*, 75:012103, 2007. [doi:10.1103/PhysRevA.75.012103]
4. Michael Ben-Or and Avinatan Hassidim. Fast quantum byzantine agreement. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 481–485, 2005. [doi:10.1145/1060590.1060662]
5. Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. arXiv:1704.00690. Presented as an invited talk at STOC 2018.
Quantum Advantage for the LOCAL Model in Distributed Computing

6 Anne Broadbent and Alain Tapp. Can quantum mechanics help distributed computing? *SIGACT News*, 39(3):67–76, 2008. doi:10.1145/1412700.1412717

7 Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation. In *Proceedings of the 30th ACM Symposium on the Theory of Computing*, pages 63–68, 1998. doi:10.1145/276698.276713

8 Yi-Jun Chang and Seth Pettie. A time hierarchy theorem for the LOCAL model. In *Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science*, pages 156–167, 2017. doi:10.1109/FOCS.2017.23

9 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

10 Michael Elkin, Hartmut Klauck, Danupon Nanongkai, and Gopal Pandurangan. Can quantum communication speed up distributed computation? In *Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing*, pages 166–175, 2014. doi:10.1145/2611462.2611488

11 Silvio Frischknecht, Stephan Holzer, and Roger Wattenhofer. Networks cannot compute their diameter in sublinear time. In *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1150–1162, 2012. doi:10.1137/1.9781611973099.91

12 Cyril Gavoille, Adrian Kosowski, and Marcin Markiewicz. What can be observed locally? In *Proceedings of the 23rd International Symposium on Distributed Computing*, pages 243–257, 2009. doi:10.1007/978-3-642-04355-0_26

13 Daniel M. Greenberger, Michael A. Horne, and Anton Zeilinger. Going beyond Bell’s theorem. In *Bell’s Theorem, Quantum Theory, and Conceptions of the Universe*, volume 37 of *Fundamental Theories of Physics*, pages 69–72. Springer, Dordrecht, 1989. doi:10.1007/978-94-017-0849-4_10

14 Aram W. Harrow and Ashley Montanaro. Quantum computational supremacy. *Nature*, 549:203–209, 2017. doi:10.1038/nature23458

15 Marc Hein, Jens Eisert, and Hans J. Briegel. Multiparty entanglement in graph states. *Physical Review A*, 69:062311, Jun 2004. doi:10.1103/PhysRevA.69.062311

16 Stephan Holzer and Roger Wattenhofer. Optimal distributed all pairs shortest paths and applications. In *Proceedings of the 2012 ACM Symposium on Principles of Distributed Computing*, pages 355–364, 2012. doi:10.1145/2332432.2332504

17 Peter Høyer and Ronald de Wolf. Improved quantum communication complexity bounds for disjointness and equality. In *Proceedings of the 19th Annual Symposium on Theoretical Aspects of Computer Science*, pages 299–310, 2002. doi:10.1007/3-540-45841-7_24

18 François Le Gall and Frédéric Magniez. Sublinear-time quantum computation of the diameter in CONGEST networks. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, pages 337–346, 2018. doi:10.1145/3212734.3212744

19 Nathan Linial. Distributive graph algorithms—global solutions from local data. In *Proceedings of the 28th Annual Symposium on Foundations of Computer Science*, pages 331–335, 1987. doi:10.1109/SFCS.1987.20

20 Nathan Linial. Locality in distributed graph algorithms. *SIAM Journal on Computing*, 21(1):193–201, 1992. doi:10.1137/0221015

21 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2011. doi:10.1017/CBO9780511976667

22 David Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, 2000. doi:10.1137/1.9780898719772

23 David Peleg, Liam Roditty, and Elad Tal. Distributed algorithms for network diameter and girth. In *Proceedings of the 39th International Colloquium on Automata, Languages, and Programming*, pages 660–672, 2012. doi:10.1007/978-3-642-31585-5_58
We formalize a $T$-round classical LOCAL network as follows. We model each node $u \in V$ as a special Turing machine with a work tape, a message tape $M_{u,v}$ for each neighbor $v \in N(u)$, and a read-only random tape. Initially, the work tape contains the input of $u$ (if there is any), the message tapes are blank, and the random tape is initialized to unbiased random bits, independent from one another and from the content of other tapes.

The set of states of each Turing machine is a disjoint union $S_0 \cup \ldots \cup S_T \cup \{q_{\text{fin}}\}$, with one designated “starting” state $q_t \in S_t$ for each $t \in \{0, \ldots, T\}$. The state $q_{\text{fin}}$ is the final state, and, for convenience, we define $q_{T+1} = q_{\text{fin}}$. The Turing machine starts in $q_0$, and, for every $t \in \{0, \ldots, T\}$, we require that a state in $S_t$ can only transition into a state in $S_{t+1} \cup \{q_{t+1}\}$. In addition, we require that the transition from $S_t$ to $q_{t+1}$ occurs with probability 1, regardless of the content of the work and the message tapes when the Turing machine first enters $q_t$.

We formalize the exchange of messages as follows. In round $t \in \{0, \ldots, T\}$, all Turing machines start in their corresponding state $q_t$ and run until they all have reached their corresponding state $q_{t+1}$. Then, if $t < T$, the configuration of message tapes $M_{u,v}$ and $M_{v,u}$ are swapped for every $\{u, v\} \in E$, and all Turing machines start round $t + 1$. Otherwise, if $t = T$, the work tape of $u \in V$ contains the output of that node.

### A.1 Restriction to finite and initial randomness

In the proof of Theorem 2 we are essentially assuming that the random tapes are of finite length. That is without loss of generality because, given any protocol on a finite network and any $\epsilon > 0$, there exists a positive integer $L$ such that, with probability at least $1 - \epsilon$, no Turing machine of the protocol ever visits more than $L$ cells of its random tape. Thus, since $\epsilon$ can be chosen arbitrarily small, we can assume all random tapes to be of some finite length $L$. Via similar reasoning, we can assume that all the randomness is provided at the beginning of the protocol, instead of fresh randomness being provided at each round.

### B The Case of Functions

A well-known fact in classical distributed computing is that randomness does not help when computing functions in the LOCAL model. In this appendix we show that this argument extends to the quantum case: we prove that any $T$-round quantum protocol computing a function can be converted into a $T$-round classical protocol computing the same function.

Suppose, in the LOCAL model, we have a $T$-round quantum protocol $P$ with the network structure given by a graph $G = (V, E)$. And suppose that $P$ computes some function $f: D \rightarrow \Sigma^{|V|}$, where $\Sigma$ is the input-output alphabet and $D \subseteq \Sigma^{|V|}$. More precisely, we assume that, for every input $x \in D$, with probability strictly larger than $1/2$ all nodes $u \in V$ output $f(x)_u$.

For a node $u \in V$ and an integer $i \geq 0$, let the $i$-neighborhood of $u$, denoted $N_i(u)$, be the set of nodes in $V$ at distance at most $i$ away from $u \in V$. And, for an input $x \in D$, let $x_{u,i}$ denote the restriction of $x$ to $N_i(u)$. 

---

F. Le Gall, H. Nishimura and A. Rosmanis

A Technical Definition of the Classical LOCAL Model

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

25 Ronald de Wolf. Quantum communication and complexity. Theoretical Computer Science, 287(1):337–353, 2002. doi:10.1016/S0304-3975(02)00377-8
Claim 8. For every $x \in D$ and every $u \in V$, the output of node $u$ is a random variable $O_u(x)$ whose probability distribution depends only on $x_{u,T}$. (This holds true even in a more powerful model where nodes are allowed to share any entanglement prior to receiving the input.)

Since the quantum protocol $\mathcal{P}$ computes $f$, for every $x \in D$ and every $u \in V$, the random variable $O_u(x)$ takes the value $f(x)_u$ with probability larger than $1/2$. Now consider the following classical $T$-round deterministic protocol: each node $u \in V$ collects the inputs from nodes in its $T$-neighborhood, which suffices to locally reproduce $O_u(x)$, and then it outputs the most probable value of $O_u(x)$. The correctness of this protocol follows from Claim 8.

Proof of Claim 8. For $t \in \{0, 1, \ldots, T\}$, let $\rho_t$ be the reduced density state of the $(T-t)$-neighborhood of $u$ after $t$ rounds of communication. By induction, we argue that the states $\rho_0, \rho_1, \ldots, \rho_T$—which we can think of forming the past light cone of $\rho_T$—all depend only on $x_{u,T}$, and no values of $x$ outside $N_T(u)$. As the base case, it clearly holds for $\rho_0$ (even in the presence of prior entanglement). For the inductive step, let us assume that, for some $t \geq 0$, $\rho_t$ depends only on $x_{u,T}$. Then the reduced density state of the $(T-t)$-neighborhood of $u$ just before the $(t+1)$-th round of communication depends only on $x_{u,T}$. In that round of communication, nodes in the $(T-t-1)$-neighborhood of $u$ receive messages only from within the $(T-t)$-neighbourhood of $u$, and thus the state $\rho_{t+1}$ also depends only on $x_{u,T}$. When $\rho_T$, the final state of the node $u$, is measured, the probabilities of various outcomes are completely determined by $\rho_T$. Hence, these probabilities depend only on $x_{u,T}$.  \[\]