Can Quantum Communication Speed Up Distributed Computation?*

Michael Elkin†
Department of Computer Science Ben-Gurion University, Israel
elkinm@cs.bgu.ac.il

Hartmut Klauck‡
Nanyang Technological University & Centre for Quantum Technologies, Singapore
hklauck@gmail.com

Danupon Nanongkai§
Faculty of Computer Science University of Vienna, Austria
danupon@gmail.com

Gopal Pandurangan¶
Nanyang Technological University, Singapore & Brown University, Providence, USA.
gopalpandurangan@gmail.com

ABSTRACT

The focus of this paper is on *quantum distributed* computation, where we investigate whether quantum communication can help in *speeding up* distributed network algorithms. Our main result is that for certain fundamental network problems such as minimum spanning tree, minimum cut, and shortest paths, quantum communication *does not* help in substantially speeding up distributed algorithms for these problems compared to the classical setting.

In order to obtain this result, we extend the technique of Das Sarma et al. [SICOMP 2012] to obtain a uniform approach to prove non-trivial lower bounds for quantum distributed algorithms for several graph optimization (both exact and approximate versions) as well as verification problems, some of which are new even in the classical setting, e.g. tight randomized lower bounds for Hamiltonian cycle and spanning tree verification, answering an open problem of Das Sarma et al., and a lower bound in terms of the weight aspect ratio, matching the upper bounds of Elkin [STOC 2004]. Our approach introduces the *Server model* and *Quantum Simulation Theorem* which together provide a connection between distributed algorithms and communication complexity. The Server model is the standard two-party communication complexity model augmented with additional power; yet, most of the hardness in the two-party model is carried over to this new model. The Quantum Simulation Theorem carries this hardness further to quantum distributed computing. Our techniques, except the proof of the hardness in the Server model, require very little knowledge in quantum computing, and this can help overcoming a usual impediment in proving bounds on quantum distributed algorithms. In particular, if one can prove a lower bound for distributed algorithms for a certain problem using the technique of Das Sarma et al., it is likely that such lower bound can be extended to the quantum setting using tools provided in this paper and without the need of knowledge in quantum computing.

Categories and Subject Descriptors

C.2.4 [**Computer Systems Organization**]: Computer Communication Networks — *Distributed Systems*; F.2.0 [**Analysis of Algorithms and Problem Complexity**]: General; G.2.2 [**Mathematics of Computing**]: Discrete Mathematics — *Graph Theory*

General Terms

Algorithms,Theory

Keywords

distributed computing; graph algorithms; quantum communication; time complexity; lower bound, CONGEST model

1. INTRODUCTION

The power and limitations of distributed (network) computation have been studied extensively over the last three

*The full version of this paper is available at `http://arxiv.org/abs/1207.5211`

†Supported by the Israeli Academy of Science, grant 593/11, and by the Binational Science Foundation, grant 2008390.

‡This work is funded by the Singapore Ministry of Education (partly through the Academic Research Fund Tier 3 MOE2012-T3-1-009) and by the Singapore National Research Foundation.

§Work done while at Nanyang Technological University, Singapore, and Brown University, USA.

¶Supported in part by the following research grants: Nanyang Technological University grant M58110000, Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 2 grant MOE2010-T2-2-082, Singapore MOE AcRF Tier 1 grant MOE2012-T1-001-094, and a grant from the US-Israel Binational Science Foundation (BSF).

PODC'14, July 15–18, 2014, Paris, France.

http://dx.doi.org/10.1145/2611462.2611488.

decades or so. In a distributed network, each individual node can communicate only with its neighboring nodes. Some distributed problems can be solved entirely via local communication, e.g., maximal independent set, maximal matching, coloring, dominating set, vertex cover, or approximations thereof. These are considered "local" problems, as they can be shown to be solved using *small (i.e., polylogarithmic)* communication (e.g., see [44, 53, 61]). For example, a maximal independent set can be computed in $O(\log n)$ time [45]. However, many important problems are "global" problems (which are the focus of this paper) from the distributed computation point of view. For example, to count the total number of nodes, to elect a leader, to compute a spanning tree (ST) or a minimum spanning tree (MST) or a shortest path tree (SPT), information necessarily must travel to the farthest nodes in a system. If exchanging a message over a single edge costs one time unit, one needs $\Omega(D)$ time units to compute the result, where D is the network diameter [53]. If message size was unbounded, one can simply collect all the information in $O(D)$ time, and then compute the result. However, in many applications, there is *bandwidth restriction* on the size of the message (or the number of bits) that can be exchanged over a communication link in one time unit. This motivates studying global problems in the CONGEST model [53], where each node can exchange at most B bits (typically B is small, say $O(\log n)$) among its neighbors in one time step. This is one of the central models in the study of distributed computation. The design of efficient algorithms for the CONGEST model, as well as establishing lower bounds on the time complexity of various fundamental distributed computing problems, has been the subject of an active area of research called (locality-sensitive) *distributed computing* for the last three decades (e.g., [53, 20, 18, 32, 61, 13]). In particular, it is now established that $\tilde{\Omega}(D+\sqrt{n})$ [1] is a fundamental lower bound on the running time of many important graph optimization (both exact and approximate versions) and verification problems such as MST, ST, shortest paths, minimum cut, ST verification etc [13].

The main focus of this paper is studying the power of distributed network computation in the *quantum setting*. More precisely, we consider the CONGEST model in the quantum setting, where nodes can use quantum processing, communicate over quantum links using quantum bits, and use exclusively quantum phenomena such as *entanglement* (e.g., see [17, 6, 25]). A fundamental question that we would like to investigate is whether quantumness can help in speeding up distributed computation for graph optimization problems; in particular, whether the above mentioned lower bound of $\tilde{\Omega}(D+\sqrt{n})$ (that applies to many important problems in the classical setting) also applies to the quantum setting.

Lower bounds for local problems (where the running time is $O(\text{poly}\log n)$) in the quantum setting usually follow directly from the same arguments as in the classical setting. This is because these lower bounds are proved using the "limited sight" argument: The nodes do not have time to get the information of the entire network. Since entanglement *cannot be used to replace communication* (by, e.g., Holevo's theorem [28] (also see [51, 50])), the same argument holds in the quantum setting with prior entanglement. This argument is captured by the notion of *physical locality* defined by Gavoille et al. [25], where it is shown that for many *local* problems, quantumness does not give any significant speedup in time compared to the classical setting.

The above limited sight argument, however, does not seem to be extendible to *global* problems where the running time is usually $\Omega(D)$, since nodes have enough time to see the whole network in this case. In this setting, the argument developed in [13] (which follows the line of work in [54, 43, 21, 36]) can be used to show tight lower bounds for many problems in the classical setting. However, this argument does not always hold in the quantum setting because it essentially relies on network "congestion": Nodes cannot communicate fast enough (due to limited bandwidth) to get important information to solve the problem. However, we know that the quantum communication and entanglement can potentially *decrease the amount of communication* and thus there might be some problems that can be solved faster. One example that illustrates this point is the following *distributed verification of disjointness function* defined in [13, Section 2.3].

EXAMPLE 1.1. *Suppose we give b-bit string x and y to node u and v in the network, respectively, where $b = \sqrt{n}$. We want to check whether the inner product $\langle x, y\rangle$ is zero or not. This is called the* Set Disjointness *problem (*Disj*). It is easy to show that it is impossible to solve this problem in less than $D/2$ rounds since there will be no node having the information from both u and v if u and v are of distance D apart. (This is the very basic idea of the limited sight argument.) This argument holds for both classical and quantum setting and thus we have a lower bound of $\Omega(D)$ on both settings. [13, Lemma 4.1] shows that this lower bound can be significantly improved to $\tilde{\Omega}(b) = \tilde{\Omega}(\sqrt{n})$ in the classical setting, even when the network has diameter $O(\log n)$. This follows from the communication complexity of $\Omega(b)$ of* Disj *[2, 31, 3, 57] and the* Simulation Theorem *of [13]. This lower bound, however, does not hold in the quantum setting since we can simulate the known $O(\sqrt{b})$-communication quantum protocol of [1] in $O(\sqrt{b}D) = O(n^{1/4}D)$ rounds.* □

Thus we have an example of a global problem that quantum communication gives an advantage over classical communication. This example also shows that the previous techniques and results from [13] does not apply to the quantum setting since [13] heavily relies on the hardness of the above distributed disjointness verification problem. A fundamental question is: *"Does this phenomenon occur for natural global distributed network problems?"*

Our paper answers the above question where we show that this phenomenon does not occur for many global graph problems. Our main result is that for fundamental global problems such as minimum spanning tree, minimum cut, and shortest paths, quantum communication *does not* help significantly in speeding up distributed algorithms for these problems compared to the classical setting. More precisely, we show that $\tilde{\Omega}(D + \sqrt{n})$ is a lower bound for these problems in the quantum setting as well. An $\tilde{O}(D + \sqrt{n})$ time algorithm for MST problem in the classical setting is well-known [38]. Recently, it has been shown that minimum cut also admits a distributed $(1+\epsilon)$-approximation algorithm in the same time in the classical setting [27, 60, 46, 49]. Also, recently it has been shown that shortest paths admits an $\tilde{O}(D + \sqrt{n}D^{1/4})$-time $(1+\epsilon)$-approximation and

[1] $\tilde{\Omega}$ and $\tilde{O}$ notations hide polylogarithmic factors.

$\tilde{O}(\sqrt{n}+D)$-time $O(\log n)$-approximation distributed classical algorithms [41, 47]. Thus, our quantum lower bound shows that quantum communication does not speed up distributed algorithms for MST and minimum cut, while for shortest paths the speed up, if any, is bounded by $O(D^{1/4})$ (which is small for small diameter graphs).

In order to obtain our quantum lower bound results, we develop a uniform approach to prove non-trivial lower bounds for quantum distributed algorithms. This approach leads us to several non-trivial quantum distributed lower bounds (which are the first-known quantum bounds for problems such as minimum spanning tree, shortest paths etc.), some of which are new even in the classical setting. Our approach introduces the *Server model* and *Quantum Simulation Theorem* which together provide a connection between distributed algorithms and communication complexity. The Server model is simply the standard two-party communication complexity model augmented with a powerful *Server* who can communicate for free but receives no input (cf. Def. 3.1). It is more powerful than the two-party model, yet captures most of the hardness obtained by the current quantum communication complexity techniques. The Quantum Simulation Theorem (cf. Theorem 3.5) is an extension of the Simulation Theorem of Das Sarma et al. [13] from the classical setting to the quantum one. It carries this hardness from the Server model further to quantum distributed computing. Most of our techniques require very little knowledge in quantum computing, and this can help overcoming a usual impediment in proving bounds on quantum distributed algorithms. In particular, if one can prove a lower bound for distributed algorithms in the classical setting using the technique of Das Sarma et al., then it is possible that one can also prove the same lower bound in the quantum setting in essentially the same way – the only change needed is that the proof has to start from problems that are hard on the server model that we provide in this paper.

2. THE SETTING

2.1 Quantum Distributed Computing Model

We study problems in a natural quantum version of the CONGEST(B) model [53] (or, in short, the B-model), where each node can exchange at most B bits (typically B is small, say $O(\log n)$) among its neighbors in one time step. The main focus of the current work is to understand the time complexity of fundamental graph problems in the B-model in the *quantum setting*. We now explain the model. We refer the readers to full version for a more rigorous and formal definition of our model.

Consider a synchronous network of processors modeled by an undirected n-node graph, where nodes model the processors and edges model the links between the processors. The processors (henceforth, nodes) are assumed to have unique IDs. Each node has limited topological knowledge; in particular, it only knows the IDs of its neighbors and knows no other topological information (e.g., whether its neighbors are linked by an edge or not). The node may also accept some additional inputs as specified by the problem at hand.

The communication is synchronous, and occurs in discrete pulses, called *rounds*. All the nodes wake up simultaneously at the beginning of each round. In each round each node u is allowed to send an arbitrary message of B bits through each edge $e=(u,v)$ incident to u, and the message will arrive at v at the end of the current round. Nodes then perform an internal computation, which finishes instantly since nodes have infinite computation power. There are several measures to analyze the performance of distributed algorithms, a fundamental one being the *running time*, defined as the worst-case number of rounds of distributed communication.

In the quantum setting, a distributed network could be augmented with two additional resources: *quantum communication* and *shared entanglement* (see e.g., [17]). Quantum communication allows nodes to communicate with each other using *quantum bits (qubits)*; i.e., in each round at most B qubits can be sent through each edge in each direction. Shared entanglement allows nodes to possess qubits that are entangled with qubits of other nodes[2]. Quantum distributed networks can be categorized based on which resources are assumed (see, e.g., [25]). In this paper, we are interested in the *most powerful model*, where both quantum communication and the *most general form of shared entanglement* are assumed: in a technical term, we allow nodes to share an arbitrary n-partite entangled state as long as it does not depend on the input (thus, does not reveal any input information). Throughout the paper, we simply refer to this model as quantum distributed network (or just distributed network, if the context is clear). All lower bounds we show in this paper hold in this model, and thus also imply lower bounds in weaker models.

2.2 Distributed Graph Problems

We focus on solving graph problems on distributed networks. We are interested in two types of graph problems: optimization and verification problems. In both types of problems, we are given a distributed network N modeled by a graph and some property $\mathcal{P}$ such as "Hamiltonian cycle", "spanning tree" or "connected component".

In optimization problems, we are additionally given a (positive) weight function $w : E(N) \to \mathbb{R}_+$ where every node in the network knows weights of edges incident to it. Our goal is to find a subnetwork M of N of minimum weight that satisfies $\mathcal{P}$ (e.g. minimum Hamiltonian cycle or MST) where every node knows which edges incident to it are in M in the end of computation. Algorithms can sometimes depend on the *weight aspect ratio* W defined as $W = \frac{\max_{e\in E(N)} w(e)}{\min_{e\in E(N)} w(e)}$.

In verification problems, we are additionally given a subnetwork M of N as the problem input (each node knows which edges incident to it are in M). We want to determine whether M has some property, e.g., M is a Hamiltonian cycle ($\mathsf{Ham}(N)$), a spanning tree ($\mathsf{ST}(N)$), or a connected component ($\mathsf{Conn}(N)$), where every node knows the answer in the end of computation.

[2]Roughly speaking, one can think of shared entanglement as a "quantum version" of shared randomness. For example, a well-known entangled state on two qubits is the *EPR* pair [19, 4] which is a pair of qubits that, when measured, will either both be zero or both be one, with probability 1/2 each. An EPR pair shared by two nodes can hence be used to, among other things, generate a shared random bit for the two nodes. Assuming entanglement implies shared randomness (even among all nodes), but also allows for other operations such as quantum teleportation [51], which replaces quantum communication by classical communication plus entanglement.

We use[3] $Q^{*,N}_{\epsilon_0,\epsilon_1}(\mathsf{Ham}(N))$ to refer to the quantum time complexity of Hamiltonian cycle verification problem on network N where for any 0-input M (i.e. M is not a Hamiltonian cycle), the algorithm has to output zero with probability at least $1-\epsilon_0$ and for any 1-input M (i.e. M is a Hamiltonian cycle), the algorithm has to output one with probability at least $1-\epsilon_1$. (We call this type of algorithm (ϵ_0, ϵ_1)-*error*.) When $\epsilon_0 = \epsilon_1 = \epsilon$, we simply write $Q^{*,N}_{\epsilon}(\mathsf{Ham}(N))$. Define $Q^{*,N}_{\epsilon_0,\epsilon_1}(\mathsf{ST}(N)))$ and $Q^{*,N}_{\epsilon_0,\epsilon_1}(\mathsf{Conn}(N))$ similarly.

We also study the *gap versions* of verification problems. For any integer $\delta \geq 0$, property $\mathcal{P}$ and a subnetwork M of N, we say that M is δ-*far*[4] from $\mathcal{P}$ if we have to add at least δ edges *from* N and remove any number of edges in order to make M satisfy $\mathcal{P}$. We denote the problem of distinguishing between the case where the subnetwork M satisfies $\mathcal{P}$ and is δ-far from satisfying $\mathcal{P}$ *the* δ-$\mathcal{P}$ *problem* (it is promised that the input is in one of these two cases). When we do not want to specify δ, we write Gap-$\mathcal{P}$. Other graph problems that we are interested in are those in [13] and their gap versions. See full version for precise definitions.

3. OUR CONTRIBUTIONS

Our first contribution is lower bounds for various fundamental verification and optimization graph problems, some of which are new even in the classical setting and answers some previous open problems (e.g. [13]). We explain these lower bounds in detail in Section 3.2. The main implication of these lower bounds is that quantum communication does *not* help in substantially speeding up distributed algorithms for many of these problems compared to the classical setting. Notable examples are MST, minimum cut, s-source distance, shortest path tree, and shortest s-t paths. In Corollary 3.9, we show a lower bound of $\Omega(\sqrt{\frac{n}{B\log n}})$ for these problems which holds against any quantum distributed algorithm with any approximation guarantee. Due to the seminal paper of Kutten and Peleg [38], we know that MST can be computed exactly in $\tilde{O}(\sqrt{n}+D)$ time in the classical setting, and thus we cannot hope to use quantum communication to get a significant speed up for MST. Recently, Ghaffari and Kuhn [27] showed that minimum cut can be $(2+\epsilon)$-approximated in $\tilde{O}(\sqrt{n}+D)$ time in the classical setting, and Su [60] and Nanongkai [46] independently improved the approximation ratio to $(1+\epsilon)$; this implies that, again, quantum communication does not help. More recently, Nanongkai [47] showed that s-source distance, shortest path tree, and shortest s-t paths, can be $(1+o(1))$-approximated in $\tilde{O}(\sqrt{n}D^{1/4}+D)$ time in the classical setting; thus, the speedup that quantum communication can provide for these problems, if any, is bounded by $O(D^{1/4})$. Moreover, if we allow higher approximation factor, the result of Lenzen and Patt-Shamir [41] implies that we can $O(\log n)$-approximate these problems in $\tilde{O}(\sqrt{n}+D)$ time; this upper bound together with our lower bound leaves no room for quantum algorithms to improve the time complexity. Besides the above lower bounds for optimization problems, we show the same lower bound of $\Omega(\sqrt{\frac{n}{B\log n}})$ for verification problems in Corollary 3.7. Das Sarma et al. [13] showed that these problems, except least-element list verification, can be solved in $\tilde{O}(\sqrt{n}+D)$ time in the classical setting; thus, once again, quantum communication does not help.

Our second contribution is the systematic way to prove lower bounds of quantum distributed algorithms. The high-level idea behind our lower bound proofs is by establishing a connection between quantum communication complexity and quantum distributed network computing. Our work is inspired by [13] (following a line of work in [54, 43, 21, 36]) which shows lower bounds for many graph verification and optimization problems in the classical distributed computing model. The main technique used to show the classical lower bounds in [13] is the *Simulation Theorem* (Theorem 3.1 in [13]) which shows how one can use lower bounds in the standard two-party classical communication complexity model [37] to derive lower bounds in the "distributed" version of communication complexity. We provide techniques of the same flavor for proving quantum lower bounds. In particular, we develop the *Quantum Simulation Theorem*. However, due to some difficulties in handling quantum computation (especially the entanglement) we need to introduce one more concept: instead of applying the Quantum Simulation Theorem to the standard two-party communication complexity model, we have to apply it to a slightly stronger model called *Server model*. We show that working with this stronger model does not make us lose much: several hard problems in two-party communication complexity remain hard in this model, so we can still prove hardness results using these problems. Quantum Simulation Theorem together with the Server model give us a tool to bring the hardness in the quantum two-party setting to the distributed setting. In Section 3.1, we give a more comprehensive overview of our techniques. Along the way, we also obtain new results in the standard communication complexity model, which we explain in Section 3.3.

3.1 Lower Bound Techniques for Quantum Distributed Computing

The high-level idea behind our lower bound proofs is establishing a connection between quantum communication complexity and quantum distributed network computing via a new communication model called the *Server model*, as shown in two middle columns of Fig. 1. This model is a generalization of the standard two-party communication complexity model in the sense that the Server model can simulate the two-party model; thus, lower bounds on this model imply lower bounds on the two-party network models. More importantly, we show that lower bounds on this model imply lower bounds on the quantum distributed model as well. This is depicted by the rightmost arrows in Fig. 1. In addition, we prove quantum lower bounds in the server model, some of which also imply *new* lower bounds in the two-party model for problems such as Hamiltonian cycle and spanning tree, even in the classical setting. This is done by

[3]We mention the reason behind our complexity notations. First, we use $*$ as in Q^* in order to emphasize that our lower bounds hold even when there is a shared entanglement, as usually done in the literature. Since we deal with different models in this paper, we put the model name after $*$. Thus, we have $Q^{*,N}$ for the case of distributed algorithm on a distributed network N, and $Q^{*,cc}$ and $Q^{*,sv}$ for the case of the standard communication complexity and the Server model (cf. Subsection 3.1), respectively.

[4]We note that the notion of δ-far should not be confused with the notion of ϵ-*far* usually used in property testing literature where we need to add and remove at least ϵ *fraction* of edges in order to achieve a desired property. The two notions are closely related. The notion that we chose makes it more convenient to reduce between problems on different models.

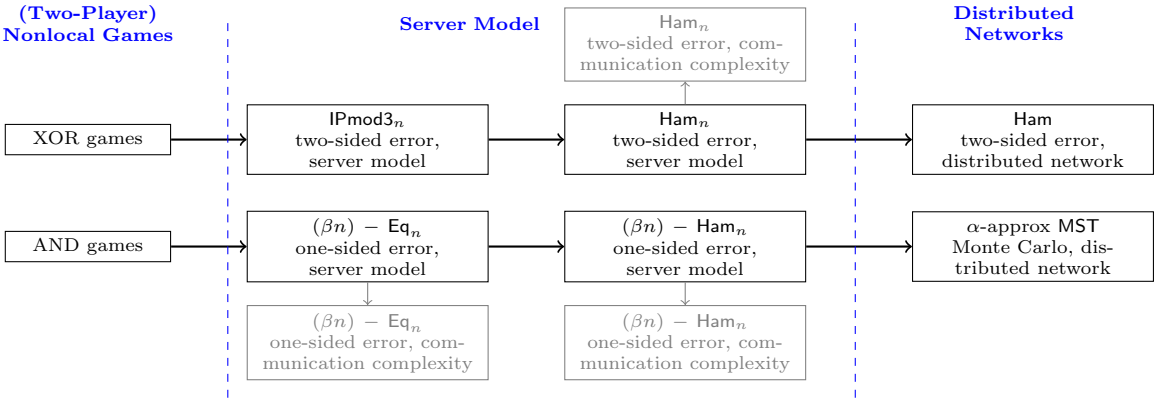


Figure 1: *Our proof structure. Lines in gray show the implications of our results in communication complexity.*

showing that certain techniques based on *nonlocal games* can be extended to prove lower bounds on the Server model as depicted by leftmost arrows in Fig. 1, and by reductions between problems in the Server models (cf. Section 5) as depicted by middle arrows in Fig. 1.

Definition 3.1 (Server Model). *There are three players in the server model: Carol, David and the server. Carol and David receive the inputs x and y, respectively, and want to compute $f(x,y)$ for some function f. (Observe that the server receives no input.) Carol and David can talk to each other. Additionally, they can talk to the server. The catch here is that the server can send messages for* free. *Thus, the communication complexity in the server model counts only messages sent by Carol and David.*

We let $Q^{*,sv}_{\epsilon_0,\epsilon_1}(f)$ denote the communication complexity — in the quantum setting with entanglement — of computing function f where for any i-input (an input whose correct output is $i \in \{0,1\}$) the algorithm must output i with probability at least $1-\epsilon_i$. We will write $Q^{*,sv}_{\epsilon}(f)$ instead of $Q^{*,sv}_{\epsilon,\epsilon}(f)$. For the standard two-party communication complexity model [37], we use $Q^{*,cc}_{\epsilon_0,\epsilon_1}(f)$ to denote the communication complexity in the quantum setting with entanglement.

To the best of our knowledge, the Server model is different from other known models in communication complexity. Clearly, it is different from multi-party communication complexity since the server receives no input and can send information for *free*. Moreover, it is easy to see that the Server model, even without prior entanglement, is at least as strong as the standard quantum communication complexity model with shared entanglement, since the server can dispense any entangled state to Carol and David. Interestingly, it turns out that the Server model is *equivalent* to the standard two-party model in the classical communication setting, while it is not clear if this is the case in the quantum communication setting. This is the main reason that proving lower bounds in the quantum setting is more challenging in its classical counterpart.

To explain some issues in the quantum setting, let us sketch the proof of the fact that the two models are *equivalent* in the classical setting. Let us first consider the deterministic setting. The proof is by the following "simulation" argument. Alice will simulate Carol and the server. Bob will simulate David and the server. In each round, Alice will see all messages sent from the server to Carol and thus she can keep simulating Carol. However, she does not see the message sent from David to the server which she needs to simulate the server. So, she must get this message from Bob. Similarly, Bob will get from Alice the message sent from Carol to the server. These are the only messages we need in each round in order to be able to simulate the protocol. Observe that the total number of bits sent between Alice and Bob is exactly the number of bits sent by Carol and David to the server. Thus, the complexities of both models are exactly the same in the deterministic case. We can conclude the same thing for the public coin setting (where all parties share a random string) since Alice and Bob can use their shared coin to simulate the shared coin of Carol, David and the server.

The above argument, however, does not seem to work in the quantum setting. The main issue with a simulation along the lines of the one sketched above is that Alice and Bob cannot simulate a "copy" of the server each. For instance one could try to simulate the server's state in a distributed way by maintaining the state that results by applying CNOT to every qubit of the server and a fresh qubit, and distribute these qubits to Alice and Bob. But then if the server sends a message to Carol, Bob would have to disentangle the corresponding qubits in his copy, which would require a message to Alice.

While we leave as an open question whether the two models are equivalent in the quantum setting, we prove that many lower bounds in the two-party model extend to the server model, via a technique called nonlocal games.

Lower Bound Techniques on the Server Model. We show that many hardness results in the two-party model (where there is no server) carry over to the Server model. This is the only part that the readers need some background in quantum computing. The main difficulty in showing this is that, the Server model, even *without* prior entanglement, is clearly at least *as strong as* the standard quantum communication complexity model (where there is no server) *with* shared entanglement, since the server can dispense any entangled state to Carol and David. Thus, it is a challenging problem, which could be of an independent interest, whether *all* hard problems in the standard model remain hard in the server model.

While we do not fully answer the above problem, we identify a set of lower bound techniques in the standard quantum communication complexity model that can be carried over to the Server model, and use them to show that *many* problems remain hard. Specifically, we show that techniques based on the (two-player) *nonlocal* games (see, e.g., [39, 40, 33]) can be extended to show lower bounds on the Server model.

Nonlocal games are games where two players, Alice and Bob, receive an input x and y from some distribution that is known to them and want to compute $f(x, y)$. Players cannot talk to each other; instead, they output one bit, say a and b, which are then combined to be an output. For example, in *XOR games* and *AND games*, these bits are combined as $a \oplus b$ and $a \wedge b$, respectively. The players' goal is to maximize the probability that the output is $f(x, y)$. We relate nonlocal games to the server model by showing that the XOR- and AND-game players can use an efficient server-model protocol to guarantee a good winning chance:

LEMMA 3.2. (SERVER MODEL LOWER BOUNDS VIA NONLOCAL GAMES) *For any boolean function f and $\epsilon_0, \epsilon_1 \geq 0$, there is an (two-player nonlocal) XOR-game strategy $\mathcal{A}'$ (respectively, AND-game strategy $\mathcal{A}''$) such that, for any input (x, y), with probability $4^{-2Q^{*,sv}_{\epsilon_0,\epsilon_1}(f)}$, $\mathcal{A}'$ (respectively, $\mathcal{A}''$) outputs $f(x, y)$ with probability at least $1 - \epsilon_{f(x,y)}$ (i.e. it outputs 1 with probability at least $1-\epsilon_1$ and 0 with probability at least $1-\epsilon_0$); otherwise (with probability $1-4^{-2Q^{*,sv}_{\epsilon_0,\epsilon_1}(f)}$), $\mathcal{A}'$ outputs 0 and 1 with probability 1/2 each (respectively, $\mathcal{A}''$ outputs 0 with probability 1).*

Roughly speaking, the above lemma compares two cases: in the "good" case $\mathcal{A}'$ outputs the correct vaue of $f(x, y)$ with high probability (the probability controlled by ϵ_0 and ϵ_1) and in the "bad case" $\mathcal{A}'$ simply outputs a random bit. It shows that if $Q^{*,sv}_{\epsilon_0,\epsilon_1}(f)$ is small, then the "good" case will happen with a non-negligible probability. In other words, the lemma says that if $Q^{*,sv}_{\epsilon_0,\epsilon_1}(f)$ is small, then the probability that the nonlocal game players win the game will be high.

This lemma gives us an access to several lower bound techniques via nonlocal games. For example, following the γ_2-norm techniques in [42, 59, 40] and the recent method of [33], we show one- and two-sided error lower bounds for many problems on the server model (in particular, we can obtain lower bounds in general forms as in [58, 59, 40]). These lower bounds match the two-party model lower bounds.

Graph Problems and Reductions between Server-Model Problems (Details in Section 5). To bring the hardness in the Server model to the distributed setting, we have to prepare hardness for the right problems in the Server model so that it is easy to translate to the distributed setting. In particular, the problems that we need are the following graph problems.

DEFINITION 3.3 (SERVER-MODEL GRAPH PROBLEMS). *Let G be a graph of n nodes[5]. We partition edges of G to $E_C(G)$ and $E_D(G)$, which are given to David and Carol, respectively. The two players have to determine whether G has some property, e.g., G is a Hamiltonian cycle (Ham_n)[6], a spanning tree (ST_n), or is connected (Conn_n). For the purpose of this paper in proving lower bounds for distributed algorithms, we restrict the problem and assume that in the case of the Hamiltonian cycle problem $E_G(C)$ and $E_D(C)$ are both perfect matchings.*

We also consider the gap version in the case of communication complexity. The notion of δ-far is slightly different from the distributed setting (cf. Section 2.2) in that we can add *any* edges to G instead of adding *only* edges in N to M. The *main challenge* in showing hardness results for these graph problems is that some of them, e.g. Hamiltonian cycle and spanning tree verification, are not known to be hard, even in the classical two-party model (they are left as open problems in [13]). To get through this, we derive several new reductions (using novel gadgets) to obtain this:

THEOREM 3.4. (SERVER-MODEL LOWER BOUNDS FOR Ham_n) *There exist some constants $\epsilon, \beta > 0$ such that for any n, $Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{Ham}_n)$ and $Q^{*,sv}_{0,\epsilon}((\beta n)\text{-}\mathsf{Ham}_n)$ are $\Omega(n)$.*

We prove Theorem 3.4 using elementary (but intricate) gadget-based reductions. Thus, no knowledge in quantum computing is required to understand this proof. Theorem 3.4 also leads to lower bounds that are new even in the classical two-party model. We discuss this in Section 3.3.

Quantum Simulation Theorem: From Server Model to Distributed Algorithms. To show the role of the Server model in proving distributed algorithm lower bounds, we prove a *quantum version* of the *Simulation Theorem* of [13] which shows that the hardness of graph problems of our interest in the Server model implies the hardness of these problems in the quantum distributed setting (the theorem below holds for several graph problems but we state it only for the Hamiltonian Cycle verification problem since it is sufficient for our purpose):

THEOREM 3.5 (QUANTUM SIMULATION THEOREM). *For any B, L, $\Gamma \geq \log L$, $\beta \geq 0$ and $\epsilon_0, \epsilon_1 > 0$, there exists a B-model quantum network N of diameter $\Theta(\log L)$ and $\Theta(\Gamma L)$ nodes such that if $Q^{*,N}_{\epsilon_0,\epsilon_1}((\beta\Gamma)\text{-}\mathsf{Ham}(N)) \leq \frac{L}{2} - 2$ then $Q^{*,sv}_{\epsilon_0,\epsilon_1}((\beta\Gamma)\text{-}\mathsf{Ham}_\Gamma) = O((B \log L) Q^{*,N}_{\epsilon_0,\epsilon_1}((\beta\Gamma)\text{-}\mathsf{Ham}(N)))$.*

In words, the above theorem states that if there is an (ϵ_0, ϵ_1)-error quantum distributed algorithm that solves the Hamiltonian cycle verification problem on N in at most $(L/2) - 2$ time, i.e. $Q^{*,N}_{\epsilon_0,\epsilon_1}(\mathsf{Ham}(N)) \leq (L/2) - 2$, then the (ϵ_0, ϵ_1)-error communication complexity in the Server model of the Hamiltonian cycle problem on Γ-node graphs is $Q^{*,sv}_{\epsilon_0,\epsilon_1}(\mathsf{Ham}_\Gamma) = O((B \log L) Q^{*,N}_{\epsilon_0,\epsilon_1}(\mathsf{Ham}(N)))$. The same statement also holds for its gap version $((\beta\Gamma)\text{-}\mathsf{Ham}(N))$. We note that the above theorem can be extended to a large class of graph problems. The proof of the above theorem does not need any knowledge in quantum computing to follow. In fact, it can be viewed as a simple modification of the Simulation Theorem in the classical setting [13]. The main difference, and the most difficult part to get our Quantum Simulation Theorem to work, is to realize that we must start from the Server model instead of the two-party model.

[5]To avoid confusion, throughout the paper we use G to denote the input graph in the Server model and N and M to denote the distributed network and its subnetwork, respectively, unless specified otherwise. For any graph H, we use $V(H)$ and $E(H)$ to denote the set of nodes and edges in H, respectively.

[6]Ham_n is used for the Hamiltonian cycle verification problem in the Server models, where n denotes the size of input graphs, and $\mathsf{Ham}(N)$ is used for the Hamiltonian cycle verification problem on a distributed network N (defined in Section 2.2).

	Problems	Previous results	Our results
B-model distributed network	Ham, ST, MST verification	$\Omega(\sqrt{n/B\log n})$ deterministic, classical communication [13, 36]	$\Omega(\sqrt{n/B\log n})$ two-sided error, quantum communication with entanglement
	Conn and other verification problems from [13]	$\Omega(\sqrt{n/B\log n})$ two-sided error, classical communication [13]	
	α-approx MST and other optimization problems from [13]	$\Omega(\sqrt{n/B\log n})$ Monte Carlo, classical communication for $W=\Omega(\alpha n)$ [13]	$\Omega(\min(\sqrt{n}, W/\alpha)/\sqrt{B\log n})$ Monte Carlo, quantum communication with entanglement
Communication Complexity	Ham, ST, and other verification problems	$\Omega(n)$ one-sided error, classical communication [56]	$\Omega(n)$ two-sided error, quantum communication with entanglement
	Gap-Ham, Gap-ST, Gap-Conn, and other gap problems for $\Omega(n)$ gap	unknown	$\Omega(n)$ one-sided error, quantum communication with entanglement

Figure 2: *Previous and our new lower bounds. We note that n is the number of nodes in the network in the case of distributed network and the number of nodes in the input graph in the case of communication complexity.*

3.2 Quantum Distributed Lower Bounds

We present specific lower bounds for various fundamental verification and optimization graph problems. Some of these bounds are new even in the classical setting. To the best of our knowledge, our bounds are the first non-trivial lower bounds for fundamental global problems.

1. Verification problems. We prove a *tight* two-sided error quantum lower bound of $\tilde{\Omega}(\sqrt{n})$ time, where n is the number of nodes in the distributed network and $\tilde{\Theta}(x)$ hides $\mathrm{poly}\log x$, for the *Hamiltonian cycle* and *spanning tree verification problems*. Our lower bound holds even in a network of small ($O(\log n)$) diameter.

THEOREM 3.6 (VERIFICATION LOWER BOUNDS). *For any B and large n, there exists $\epsilon > 0$ and a B-model n-node network N of diameter $\Theta(\log n)$ such that any (ϵ,ϵ)-error quantum algorithm with prior entanglement for Hamiltonian cycle and spanning tree verification on N requires $\Omega(\sqrt{\frac{n}{B\log n}})$ time. That is, $Q^{*;N}_{\epsilon,\epsilon}(\mathsf{Ham}(N))$ and $Q^{*;N}_{\epsilon,\epsilon}(\mathsf{ST}(N))$ are $\Omega(\sqrt{\frac{n}{B\log n}})$.*

Our bound implies a new bound on the classical setting which answers the open problem in [13], and is the *first randomized lower bound* for both graph problems, subsuming the deterministic lower bounds for Hamiltonian cycle verification [13], spanning tree verification [13] and minimum spanning tree verification [36]. It is also shown in [13] that Ham can be reduced to several problems via deterministic classical-communication reductions. Since these reductions can be simulated by quantum protocols, we can use these reductions straightforwardly to show that all lower bounds in [13] hold even in the quantum setting.

COROLLARY 3.7. *The statement in Theorem 3.6 holds for the following verification problems: Connected component, spanning connected subgraph, cycle containment, e-cycle containment, bipartiteness, s-t connectivity, connectivity, cut, edge on all paths, s-t cut and least-element list. (See [13] and the full version for definitions.)*

Fig. 2 compares our results with previous results for verification problems.

2. Optimization problems. We show a *tight* $\tilde{\Omega}(\min(W/\alpha, \sqrt{n}))$-time lower bound for any α-approximation quantum randomized (Monte Carlo and Las Vegas) distributed algorithm for the MST problem.

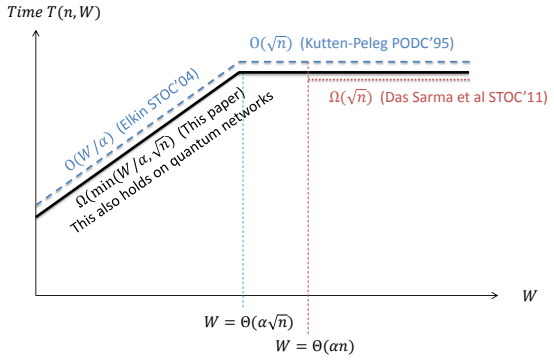


Figure 3: *Previous and our new bounds (cf. Theorem 3.8 and Corollary 3.9) for approximating the MST problem in distributed networks when N and α are fixed. The dashed line (in blue) represents the deterministic upper bounds (Algorithms). The dotted line (in red) is the previous lower bound for randomized algorithms. The solid line (in black) represents the bounds shown in this paper. Note that the previous lower bounds hold only in the classical setting while the new lower bounds hold in the quantum setting even when entanglement is allowed.*

THEOREM 3.8 (OPTIMIZATION LOWER BOUNDS). *For any n, B, W and $\alpha < W$ there exists $\epsilon > 0$ and a B-model $\Theta(n)$-node network N of diameter $\Theta(\log n)$ and weight aspect ratio W such that any ϵ-error α-approximation quantum algorithm with prior entanglement for computing the minimum spanning tree problem on N requires $\Omega(\frac{1}{\sqrt{B\log n}}\min(W/\alpha, \sqrt{n}))$ time.*

This result generalizes the bounds in [13] to the quantum setting. Moreover, this lower bound implies the same bound in the classical model, which improves [13] (see Fig. 3) and matches the deterministic upper bound of $O(\min(W/\alpha, \sqrt{n}))$ resulting from a combination of Elkin's α-approximation $O(W/\alpha)$-time deterministic algorithm [21] and Peleg and Rubinovich's $O(\sqrt{n})$-time exact deterministic algorithm [24, 38] in the classical communication model. Thus this bound is tight up to a $\Theta(\sqrt{B\log n})$ factor. It is the first bound that is *tight for all values of the aspect ratio W*. Fig. 3 compares our lower bounds with previous bounds. By using the same reduction as in [13], our bound also implies that all lower bounds in [13] hold even in the quantum setting.

COROLLARY 3.9. *The statement in Theorem 3.8 also holds for the following problems: minimum spanning tree, shallow-light tree, s-source distance, shortest path tree, minimum routing cost spanning tree, minimum cut, minimum s-t cut, shortest s-t path and generalized Steiner forest. (See [13] and the full version for definitions.)*

3.3 Additional Results: Lower Bounds on Communication Complexity

In proving the results in previous subsections, we prove several bounds on the Server model. Since the Server model is stronger than the standard communication complexity model (as discussed in Subsection 3.1), we obtain lower bounds in the communication complexity model as well. Some of these lower bounds are new even in the classical setting. In particular, our bounds in Theorem 3.4 lead to the following corollary. (Note that we use $Q^{*,cc}_{\epsilon_0,\epsilon_1}(\mathcal{P}_n)$ to denote the communication complexity of verifying property $\mathcal{P}$ of n-node graphs on the standard quantum communication complexity model with entanglement.)

COROLLARY 3.10. *For any n and some constants $\epsilon, \beta > 0$, $Q^{*,cc}_{\epsilon,\epsilon}(\mathsf{P}_n) = \Omega(n)$, and $Q^{*,cc}_{0,\epsilon}((\beta n) - \mathsf{P}_n) \geq Q^{*,sv}_{0,\epsilon}((\beta n) - \mathsf{P}_n) = \Omega(n)$, where P_n can be any of the following verification problems: Hamiltonian cycle, spanning tree, connectivity, s-t connectivity, and bipartiteness.*

To the best of our knowledge, the lower bounds for Hamiltonian cycle and spanning tree verification problems are the first two-sided error lower bounds for these problems, even in the classical two-party setting (only nondeterministic, thus one-sided error, lower bounds are previously known [56]). The bounds for Bipartiteness and s-t connectivity follow from a reduction from Inner Product given in [2], and a lower bound for Connectivity was recently shown in [30]. We note that we prove the gap versions via a reduction from recent lower bounds in [33] and observe new lower bounds for the gap versions of Set Disjointness and Equality.

4. OTHER RELATED WORK

While our work focuses on solving graph problems in quantum distributed networks, there are several prior works focusing on other distributed computing problems (including communication complexity in the two-party or multiparty communication model) using quantum effects. We note that fundamental distributed computing problems such as leader election and byzantine agreement have been shown to solved better using quantum phenomena (see e.g., [17, 63, 5]). Entanglement has been used to reduce the amount of communication of a specific function of input data distributed among 3 parties [12] (see also the work of [9, 16, 62] on multiparty quantum communication complexity).

There are several results showing that quantum communication complexity in the two-player model can be more efficient than classical randomized communication complexity (e.g. [7, 55]). These results also easily extend to the so-called number-in-hand multiparty model (in which players have separate inputs). As of now no separation between quantum and randomized communication complexity is known in the number-on-the-forehead multiparty model, in which players' inputs overlap. Other papers concerning quantum distributed computing include [8, 11, 34, 35, 52, 23].

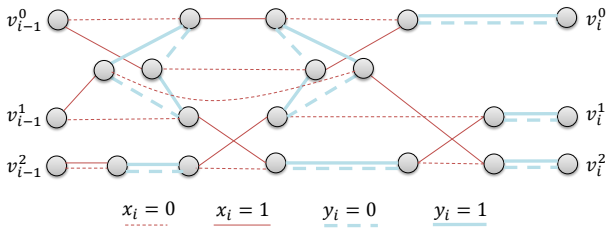


Figure 4: The construction of gadget G_i. If $x_i = 0$ then Alice adds dashed thin edges (in red); otherwise she adds solid thin edges (in red). If $y_i = 0$ then Bob adds dashed thick edges (in blue); otherwise he adds solid thick edges (in blue).

5. SERVER-MODEL LOWER BOUNDS FOR $\mathsf{Ham_n}$ (THEOREM 3.4)

In this section, we prove Theorem 3.4, which leads to new lower bounds for several graph problems as discussed in Section 3.3. The proof uses gadget-based reductions between problems on the Server model. This illustrates our point that no knowledge in quantum computing is needed after a lower bound in the Server model is obtained. We will reduce from the $\mathsf{IPmod3}_n$: two players, Carol and David, are given n-bit strings x and y, respectively, and they have to output 1 if $(\sum_{i=1}^{n} x_i y_i) \mod 3 = 0$ and 0 otherwise. We use the following lower bound, which is proved in the full version.

THEOREM 5.1. *For some $\beta, \epsilon > 0$ and any large n,*

$$Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{IPmod3}_n) = \Omega(n).$$

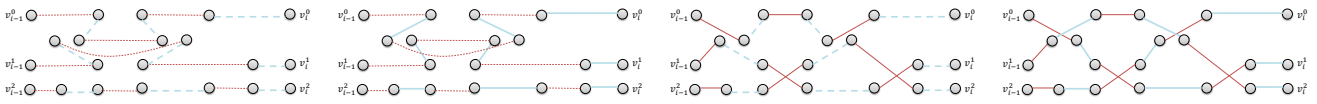
We first sketch the lower bound proof of $Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{Ham}_n)$ and show later how to extend to the gap version. We will show that for any $0 \leq \epsilon \leq 1$ and some constant c, $Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{IPmod3}_n) = O(Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{Ham}_{cn}))$. The theorem then immediately follows from the fact that $Q^{*,sv}_{\epsilon,\epsilon}(\mathsf{IPmod3}_n) = \Omega(n)$.

Let $x = x_1 \dots x_n$ and $y = y_1 \dots y_n$ be the input of $\mathsf{IPmod3}_n$. We construct a graph G which is an input of Ham_{cn} as follows. The graph G consists of n gadgets, denoted by $G_1, \dots, G_n$. For any $1 \leq i \leq n-1$, gadgets G_i and G_{i+1} share exactly three nodes denoted by v^0_i, v^1_i, v^2_i. Each gadget G_i is constructed based on the values of x_i and y_i as outlined in Fig. 4. The following observation can be checked by drawing G_i for all cases of x_i and y_i (as in Fig. 5).

OBSERVATION 5.2. *For any value of (x_i, y_i), G_i consists of three paths where v^j_{i-1} is connected by a path to $v^{(j+x_i \cdot y_i) \mod 3}_i$, for any $0 \leq j \leq 2$. Moreover, Alice's (respectively Bob's) edges, i.e. thin (red) lines (respectively thick (blue) lines) in Fig. 4, form a matching that covers all nodes except v^j_i (respectively v^j_{i-1}) for all $0 \leq j \leq 2$.*

Thus, when we put all gadgets together, graph G will consist of three paths connecting between nodes in $\{v^j_0\}_{0 \leq j \leq 2}$ on one side and nodes in $\{v^j_n\}_{0 \leq j \leq 2}$ on the other. How these paths look like depends on the structure of each gadget G_i which depends on the value of $x_i \cdot y_i$. The following lemma follows trivially from Observation 5.2.

LEMMA 5.3. *G consists of three paths P^0, P^1 and P^2 where for any $0 \leq j \leq 2$, P^j has v^j_0 as one end vertex and $v^{(j+\sum_{1 \leq i \leq n} x_i \cdot y_i) \mod 3}_n$ as the other.*



(a) $x_i y_i = 00$ (b) $x_i y_i = 01$ (c) $x_i y_i = 10$ (d) $x_i y_i = 11$

Figure 5: Gadget G_i for different values of x_i and y_i. The main observation is that if $x_i \cdot y_i = 0$ then G_i consists of paths from v_{i-1}^j to v_i^j for all $0 \leq j \leq 2$. Otherwise, it consists of paths from v_{i-1}^j to $v_i^{(j+1) \mod 3}$.

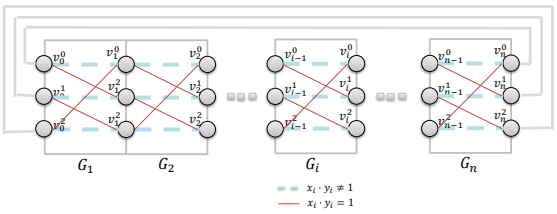


Figure 6: The graph G consists of gadgets $G_1, \ldots G_n$. The solid thick edges (in gray) linking between v_0^j and v_n^j, for $0 \leq j \leq 2$ represent the fact that $v_0^j = v_n^j$. Lines that appear in each gadget G_i depicts what we observe in Observation 5.2: solid thin lines (in red) represent paths that will appear in G_i if $x_i \cdot y_i = 0$, and dashed thick lines (in blue) represent paths that will appear in G_i if $x_i \cdot y_i = 1$.

Now, we complete the description of G by letting $v_0^j = v_n^j$ for all $0 \leq j \leq 2$. It then follows that G is a Hamiltonian cycle if and only if $\sum_{1 \leq i \leq n} x_i \cdot y_i \mod 3 \neq 0$ (see Fig. 6). Thus we can check that $\sum_{1 \leq i \leq n} x_i \cdot y_i \mod 3$ is zero or not by checking whether G is a Hamiltonian cycle or not. Theorem 3.4 now follows from Theorem 5.1.

6. CONCLUSION AND OPEN PROBLEMS

In this paper, we derive several lower bounds for important network problems in a quantum distributed network. We show that quantumness does not really help in obtaining faster distributed algorithms for fundamental problems such as minimum spanning tree, minimum cut, and shortest paths. Our approach gives a uniform way to prove lower bounds for various problems. Our technique closely follows the Simulation Theorem introduced by Das Sarma et al. [13], which shows how to use the two-party communication complexity to prove lower bounds for distributed algorithms. The main difference of our approach is the use of the Server model. We show that many problems that are hard in the quantum two-party communication setting (e.g. IPmod3) are also hard in the Server model, and show new reductions from these problems to graph verification problems of our interest. Some of these reductions give tighter lower bounds even in the classical setting.

Since the technique of Das Sarma et al. can be used to show lower bounds of many problems that are not covered in this paper (e.g. [22, 29, 48, 41, 14, 26, 10]), it is interesting to see if these lower bounds remain valid in the quantum setting. Since most of these problems rely on a reduction from the set disjointness problem, the main challenge is to obtain new reductions that start from problems that are proved hard on the Server model such as IPmod3. One problem that seems to be harder than others is the random walk problem [48, 15] since the previous lower bound in the classical setting requires a *bounded-round* communication complexity [48]. Proving lower bounds for the random walk problem thus requires proving a bounded-round communication complexity in the Server model as the first step. This requires different techniques since the nonlocal games used in this paper destroy the round structure of protocols.

It is also interesting to better understand the role of the Server model: Can we derive a quantum two-party version of the Simulation Theorem, thus eliminating the need of the Server model? Is the Server model *strictly* stronger than the two-party quantum communication complexity model? Also, it will be interesting to explore *upper* bounds in the quantum setting: Do quantum distributed algorithms help in solving other fundamental graph problems ?

References

[1] S. Aaronson and A. Ambainis. Quantum search of spatial regions. *Theory of Computing*, 1(1):47–79, 2005. Also in FOCS'03.

[2] L. Babai, P. Frankl, and J. Simon. Complexity classes in communication complexity theory. In *FOCS*, pages 337–347, 1986.

[3] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004. Also in FOCS'02.

[4] J. S. Bell. On the Einstein-Podolsky-Rosen paradox. *Physics*, 1:195–200, 1964.

[5] M. Ben-Or and A. Hassidim. Fast quantum byzantine agreement. In *STOC*, pages 481–485, 2005.

[6] A. Broadbent and A. Tapp. Can quantum mechanics help distributed computing? *SIGACT News*, 39(3):67–76, 2008.

[7] H. Buhrman, R. Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *STOC*, pages 63–68, 1998.

[8] H. Buhrman and H. Röhrig. Distributed quantum computing. In *MFCS*, pages 1–20, 2003.

[9] H. Buhrman, W. van Dam, P. Hoyer, and A. Tapp. Quantum multiparty communication complexity. *Physical Review A*, 60:2737–2741, 1999.

[10] K. Censor-Hillel, M. Ghaffari, and F. Kuhn. Distributed connectivity decomposition. In *PODC*, 2014.

[11] B. S. Chlebus, D. R. Kowalski, and M. Strojnowski. Scalable quantum consensus for crash failures. In *DISC*, pages 236–250, 2010.

[12] R. Cleve and H. Buhrman. Substituting quantum entanglement for communication. *Physical Review A*, 56(2):1201–1204, 1997.

[13] A. Das Sarma, S. Holzer, L. Kor, A. Korman, D. Nanongkai, G. Pandurangan, D. Peleg, and R. Wattenhofer. Distributed verification and hardness of distributed approximation. *SIAM J. Comput.*, 41(5):1235–1265, 2012.

[14] A. Das Sarma, A. R. Molla, and G. Pandurangan. Distributed computation of sparse cuts. *CoRR*, abs/1310.5407, 2013.

[15] A. Das Sarma, D. Nanongkai, G. Pandurangan, and P. Tetali. Distributed random walks. *J. ACM*, 60(1):2, 2013.

[16] R. de Wolf. Quantum communication and complexity. *Theoretical Computer Science*, 287(1):337–352, 2002.

[17] V. S. Denchev and G. Pandurangan. Distributed quantum computing: a new frontier in distributed systems or science fiction? *SIGACT News*, 39(3):77–95, 2008.

[18] D. P. Dubhashi, F. Grandioni, and A. Panconesi. Distributed Algorithms via LP Duality and Randomization. In *Handbook of Approximation Algorithms and Metaheuristics*. Chapman and Hall/CRC, 2007.

[19] A. Einstein, B. Podolsky, and N. Rosen. Can quantum-mechanical description of physical reality be considered complete? *Phys. Rev.*, 47(10):777–780, May 1935.

[20] M. Elkin. Distributed approximation: a survey. *SIGACT News*, 35(4):40–57, 2004.

[21] M. Elkin. An Unconditional Lower Bound on the Time-Approximation Trade-off for the Distributed Minimum Spanning Tree Problem. *SIAM J. Comput.*, 36(2):433–456, 2006. Also in STOC'04.

[22] S. Frischknecht, S. Holzer, and R. Wattenhofer. Networks cannot compute their diameter in sublinear time. In *SODA*, pages 1150–1162, 2012.

[23] S. Gaertner, M. Bourennane, C. Kurtsiefer, A. Cabello, and H. Weinfurter. Experimental demonstration of a quantum protocol for byzantine agreement and liar detection. *PHYS.REV.LETT.*, 100:070504, 2008.

[24] J. Garay, S. Kutten, and D. Peleg. A sublinear time distributed algorithm for minimum-weight spanning trees. *SIAM J. on Computing*, 27:302–316, 1998. Also in FOCS'93.

[25] C. Gavoille, A. Kosowski, and M. Markiewicz. What can be observed locally? In *DISC*, pages 243–257, 2009.

[26] M. Ghaffari. Near-optimal distributed approximation of minimum-weight connected dominating set. In *ICALP (2)*, 2014.

[27] M. Ghaffari and F. Kuhn. Distributed minimum cut approximation. In *DISC*, pages 1–15, 2013.

[28] A. S. Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973. English translation in *Problems of Information Transmission*, 9:177–183, 1973.

[29] S. Holzer and R. Wattenhofer. Optimal distributed all pairs shortest paths and applications. In *PODC*, pages 355–364, 2012.

[30] G. Ivanyos, H. Klauck, T. Lee, M. Santha, and R. de Wolf. New bounds on the classical and quantum communication complexity of some graph properties. In *FSTTCS*, pages 148–159, 2012.

[31] B. Kalyanasundaram and G. Schnitger. The Probabilistic Communication Complexity of Set Intersection. *SIAM J. Discrete Math.*, 5(4):545–557, 1992.

[32] M. Khan, F. Kuhn, D. Malkhi, G. Pandurangan, and K. Talwar. Efficient distributed approximation algorithms via probabilistic tree embeddings. In *PODC*, pages 263–272, 2008.

[33] H. Klauck and R. de Wolf. Fooling one-sided quantum protocols. *Manuscript*, 2012.

[34] H. Kobayashi, K. Matsumoto, and S. Tani. Ba: Exactly electing a unique leader is not harder than computing symmetric functions on anonymous quantum networks. In *PODC*, pages 334–335, 2009.

[35] H. Kobayashi, K. Matsumoto, and S. Tani. Computing on anonymous quantum network. *CoRR*, abs/1001.5307, 2010.

[36] L. Kor, A. Korman, and D. Peleg. Tight bounds for distributed mst verification. In *STACS*, pages 69–80, 2011.

[37] E. Kushilevitz and N. Nisan. *Communication complexity*. Cambridge University Press, New York, NY, USA, 1997.

[38] S. Kutten and D. Peleg. Fast Distributed Construction of Small k-Dominating Sets and Applications. *J. Algorithms*, 28(1):40–66, 1998. Also in PODC'95.

[39] T. Lee and A. Shraibman. Lower bounds in communication complexity. *Foundations and Trends in Theoretical Computer Science*, 3(4):263–398, 2009.

[40] T. Lee and S. Zhang. Composition theorems in communication complexity. In *ICALP (1)*, pages 475–489, 2010.

[41] C. Lenzen and B. Patt-Shamir. Fast routing table construction using small messages: extended abstract. In *STOC*, pages 381–390, 2013.

[42] N. Linial and A. Shraibman. Lower bounds in communication complexity based on factorization norms. *Random Struct. Algorithms*, 34(3):368–394, 2009. Also in STOC'07.

[43] Z. Lotker, B. Patt-Shamir, and D. Peleg. Distributed MST for constant diameter graphs. *Distributed Computing*, 18(6):453–460, 2006. Also in PODC'01.

[44] M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM J. Comput.*, 15(4):1036–1053, 1986.

[45] M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM J. Comput.*, 15(4):1036–1053, 1986. Also in STOC'85.

[46] D. Nanongkai. Brief announcement: Almost-tight approximation distributed algorithm for minimum cut. In *PODC*, 2014.

[47] D. Nanongkai. Distributed Approximation Algorithms for Weighted Shortest Paths. In *STOC*, 2014.

[48] D. Nanongkai, A. Das Sarma, and G. Pandurangan. A tight unconditional lower bound on distributed randomwalk computation. In *PODC*, pages 257–266, 2011.

[49] D. Nanongkai and H. Su. Almost-tight distributed minimum cut algorithms. Manuscript, 2014.

[50] A. Nayak. Optimal lower bounds for quantum automata and random access codes. In *FOCS*, pages 369–377, 1999.

[51] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information (Cambridge Series on Information and the Natural Sciences)*. Cambridge University Press, 1 edition, Jan. 2004.

[52] S. P. Pal, S. K. Singh, and S. Kumar. Multi-partite quantum entanglement versus randomization: Fair and unbiased leader election in networks, 2003.

[53] D. Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[54] D. Peleg and V. Rubinovich. A Near-Tight Lower Bound on the Time Complexity of Distributed Minimum-Weight Spanning Tree Construction. *SIAM J. Comput.*, 30(5):1427–1442, 2000. Also in FOCS'99.

[55] R. Raz. Exponential separation of quantum and classical communication complexity. In *STOC*, pages 358–369, 1999.

[56] R. Raz and B. Spieker. On the "log rank"-conjecture in communication complexity. *Combinatorica*, 15(4):567–588, 1995. Also in FOCS'93.

[57] A. A. Razborov. On the Distributional Complexity of Disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992. Also in ICALP'90.

[58] A. A. Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya: Mathematics*, 67(1):145, 2003.

[59] A. A. Sherstov. The pattern matrix method. *SIAM J. Comput.*, 40(6):1969–2000, 2011. Also in STOC'08.

[60] H. Su. Brief announcement: A distributed minimum cut approximation scheme. In *SPAA*, 2014.

[61] J. Suomela. Survey of local algorithms. *ACM Computing Surveys*, to appear.

[62] A. Ta-Shma. Classical versus quantum communication complexity. *SIGACT News*, 30(3):25–34, 1999.

[63] S. Tani, H. Kobayashi, and K. Matsumoto. Exact quantum algorithms for the leader election problem. In *STACS*, pages 581–592, 2005.