Crash course on causal limits in the LOCAL model

Francesco d'Amore

Introduction

This document is a self-contained note on recent advances in the investigation of the role of quantum information in the LOCAL model of computation. I want to stress that no background in quantum information is needed to understand these notes: however, it is assumed that the reader is familiar with basic probability theory, and has some introductory knowledge of distributed algorithms and graph theory. These notes were written quickly, so I apologize for possible mistakes and confusion. If you spot one mistake, or you have any suggestion, please contact me.

Preliminaries

We consider the set $\mathbb{N}$ of natural numbers to start with 0. We also define $\mathbb{N}_+ = \mathbb{N} \setminus \{0\}$. Similarly, we denote the set of real numbers by $\mathbb{R}$ and its restriction to positive real numbers $\mathbb{R}_+$. For any positive integer $n \in \mathbb{N}_+$, we denote the set $\{1, \dots, n\}$ by $[n]$, and for any two integers $n < m \in \mathbb{N}$ the set $[n : m]$ denotes $\{n, n+1, \dots, m\}$. For a k-dimensional vector $\mathbf{x} \in [n]^k$, we denote by $\mathbf{x}[i]$ its i-th entry.

Graphs. All graphs in this paper are simple graphs without self-loops. Let $G = (V, E)$ be a simple graph with $n \in \mathbb{N}$ nodes. If the set of nodes and the set of edges are not specified, we refer to them by $V(G)$ and $E(G)$, respectively. For any edge $e = \{u, v\} \in E(G)$, we also write $e = uv = vu$.

For any graph H such that $V(H) \subseteq V$ and $E(H) \subseteq E$, we say that H is a subgraph of G, and we write $H \subseteq G$. For any subset of nodes $A \subseteq V$, we denote by $G[A]$ the subgraph induced by the nodes in A, i.e. the graph H such that $V(H) = A$ and $E(H) = \{\{u, v\} \in E(G) \mid u, v \in A\}$. For any nodes $u, v \in V$, $\mathrm{dist}_G(u, v)$ denotes the distance between u and v in G (i.e., the number of edges of any shortest path between u and v in G); if u and v are disconnected, then $\mathrm{dist}_G(u, v) = +\infty$. If G is clear from the context, we may also simply write $\mathrm{dist}(u, v) = \mathrm{dist}_G(u, v)$. The diameter $\mathrm{diam}(G)$ of a graph G is the maximum distance between any two nodes, i.e. $\mathrm{diam}(G) = \max_{u,v \in V(G)} \mathrm{dist}_G(u, v)$. If the graph is disconnected, $\mathrm{diam}(G) = +\infty$. For $T \in [n]$, the radius-T neighborhood of a node $u \in V$ is the set $\mathcal{N}_T(u) = \{v \in V \mid \mathrm{dist}(u, v) \leq T\}$. The radius-$T$ neighborhood of a subset $A \subseteq V$ is the set $\mathcal{N}_T(A) = \{v \in V \mid \exists u \in A : \mathrm{dist}(u, v) \leq T\}$. Similarly, the radius-$T$ neighborhood of a subgraph $H \subseteq G$ is the set $\mathcal{N}_T(H) = \{v \in V(G) \mid \exists u \in V(H) : \mathrm{dist}_G(u, v) \leq T\}$.

We also define the notion of *isomorphism* between graphs. Two graphs G and H are isomorphic if there exists a function $\varphi : V(G) \to V(H)$ such that the following properties are satisfied:

1. The function φ is bijective.
2. For any edge $\{u, v\} \in E(G)$ it holds that $\{\varphi u, \varphi(v)\} \in E(H)$.

In such case, we say that φ is an isomorphism between G and H.

Probability. For an event E, we denote its complementary by $\bar{E}$. When an event E_n depending on a parameter n holds *with high probability* (or w.h.p., in short) in n, we usually mean that the probability of E_n occurring is at least $1 - 1/n^c$ for some $c > 0$. We also write $f(n) = \mathrm{poly}(n)$ when there exist three constants $a, b, c \in \mathbb{R}_+$ such that $an^c \leq f(n) \leq bn^c$.

1 Lecture 1

1.1 The LOCAL model

The LOCAL model is a model of distributed computation that solves graph problems. In order to define the model, we first have to specify what *graph labeling problems* are.

Definition 1.1 (Graph labeling problem)**.** Let Σ_{in} and Σ_{out} two sets of input and output labels, respectively. A *labeling problem* Π is a mapping $(G, x) \mapsto \{y_i\}_{i \in I}$, with I being a discrete set of indices, that assigns to every graph G with any input labeling $x : V(G) \to \Sigma_{\text{in}}$ a set of permissible output functions $y_i : V(G) \to \Sigma_{\text{out}}$ that might depend on (G, x). The mapping must be closed under graph isomorphism, i.e., if $\varphi : V(G) \to V(G')$ is an isomorphism between G and G', and $y_i \in \Pi((G', \Sigma_{\text{in}}))$, then $y_i \circ \varphi \in \Pi((G, x \circ \varphi))$.

Note that we assume the mapping defining the problem to be defined for all possible graphs: the set of permissible output functions might be empty.

Example 1.1. Any coloring problem is a notorious graph labeling problem. E.g., in the 3-coloring problem, one wishes to color the nodes of a graph such that no pair of adjacent nodes share the same color. This problem can be formalized as follows: $\Sigma_{\text{in}} = \{0\}$, $\Sigma_{\text{out}} = \{1, 2, 3\}$. For any 3-colorable graph $G = (V, E)$, the set of permissible output functions is

$$\{y : V \to \Sigma_{\text{out}} \mid y(u) \neq y(v) \text{ for all } u, v \in V \text{ such that } \{u, v\} \in E\}.$$

Example 1.2. The consensus problem is also a graph labeling problem. In the consensus problem, every node is given in input an opinion out of a label set, and must output any opinion so that it agrees with all other nodes. Furthermore, the outputted opinion must be among the ones that have initially been assigned to the graph as inputs. In its binary formulation, $\Sigma_{\text{in}} = \{0, 1\}$, $\Sigma_{\text{out}} = \{0, 1\}$. Furthermore, for any input graph $G = (V, E)$ with input labeling $x : V \to \Sigma_{\text{in}}$, the set of permissible outputs is

$$\{y : V \to \Sigma_{\text{out}} \mid \exists u \in V : y(v) = x(u) \text{ for all } v \in V\}.$$

The class of graph labeling problems is quite broad and encompasses many natural and classical problems that are extensively studied also in centralized models of computing. We now describe our model of interest: the LOCAL model of computation.

The LOCAL model. The LOCAL model of distributed computation was introduced by Linial in 1987 [Lin87]. The LOCAL model allows solving graph labeling problems in a distributed fashion. The distributed system is represented by the graph $G = (V, E)$ given in input to the problem. Throughout the document, we will interchangeably use the terms *nodes* and *processors*.

The processors operate in a sequence of discrete, synchronous rounds. In each round the processors may perform unbounded computations on their respective local state variables and subsequently exchange messages of arbitrary size along the edges (or *links*) given by the underlying input graph.

Nodes identify their neighbors by using integer labels assigned successively to communication ports. (This assignment may be done adversarially.) Barring their degree, all nodes are identical and operate according to the same local computation procedures. Initially all local state variables have the same value for all processors; the sole exceptions consist in the variable $x(v)$ encoding the (possible) input for each processor v, an integer-valued identifier $\text{Id}(v)$ that belongs to the set $[n^c]$ for some constant $c > 0$ that is independent of the size of the graph and is unique for each processor, and a variable that encodes the number of nodes n in the system.[1]

The output of any algorithm is an assignment of output labels $y : V \to \Sigma_{\text{out}}$ to nodes, and the algorithm is assumed to terminate once all labels $y(v)$ are definitely fixed. In the LOCAL model all nodes and their links are fault-free. The local computation procedures may be randomized by giving each processor access to its own infinite random bit string of infinite length; in this case, we are in the *randomized* LOCAL (rand-LOCAL) model as opposed to *deterministic* LOCAL (det-LOCAL). In the randomized LOCAL model, we

[1] In other works, n is *unknown* to the nodes.

further assume that the identifiers are sampled by the nodes (which have knowledge of the value of n). All identifiers will unique with probability $1 - 1/\text{poly}(n)$.

The running time of an algorithm is the number of synchronous rounds required by all nodes to produce output labels. If an algorithm running time is T, we also say that the algorithm has locality T. Notice that T can be a function of the size of the input graph. We define the complexity class $\mathcal{L}[T]$ to be the family of pairs $(\Pi, \mathcal{F})$ where $\mathcal{P}$ is a graph labeling problem and $\mathcal{F}$ is the input graph family that can be solved by a LOCAL algorithm in T rounds.

Remark 1.1. Notice that the amount of information that a node $v \in V$ of a graph $G = (V, E)$ gathers in r rounds of communication in any algorithm is tantamount to the information that is contained in the graph $G[\mathcal{N}_r(v)]$.

Exercise 1.1. Show that any T-round algorithm in the LOCAL model can be turned into a 0-round algorithm where every node has access to the information contained in $G[\mathcal{N}_r(v)]$, and vice versa.

The LOCAL model is the *strongest* possible model of synchronous distributed computation, provided that local computation and communication are *classical*, in the sense that they involve classical bits. A lower bound in this model is a lower bound for any other synchronous distributed model. Such property makes the LOCAL model a fundamental model of computing to investigate the limits and the capabilities of synchronous distributed computation. Also, the LOCAL model captures all those distributed systems in which the cost of sending messages is incredibly more time-consuming than performing local computation.

Remark 1.2. Based on Remark 1.1 and Exercise 1.1, it is easy to see that any graph labeling problem can be solved by a T-round LOCAL algorithm with $T = \text{diam}(G) + 1$. We can simply design an algorithm that gathers all the topology of the graph and all the information contained in the graph locally at each node and brute-force a solution. We call this algorithm the *gathering algorithm*.

Exercise 1.2. Think about the following graph problem: every node must orient its adjacent edges so that the orientation agrees with that of the neighboring nodes. What is the complexity of this problem in det-LOCAL? What happens if nodes *don't have* identifiers?

We now want to introduce a lower bound technique that was already used in the seminal work by Linial [Lin87]. This technique is applicable to a specific subclass of graph labeling problems, i.e. *Locally Checkable Labeling problems* (LCLs). The notion of LCL problems was first formalized by Naor and Stockmeyer [NS95]: it captures labeling problems where, for any input graph, an output y is permissible if and only if the restrictions of the problem on any local neighborhoods can be solved and there exist compatible local permissible output functions whose combination provides y.

Let us introduce some notation and definitions. For any function $f : A \to B$ and any subset $A' \subseteq A$, let us denote the restriction of f to A' by $f \restriction_{A'}$. Furthermore, we define a centered graph to be a pair (H, v_H) where H is a graph and $v_H \in V(H)$ is a vertex of H that we name the *center* of H. The *radius* of a centered graph is the maximum distance from v_H to any other node in H.

Definition 1.2 (Locally verifiable labeling problem)**.** Let $r, \Delta \in \mathbb{N}$. Let Σ_{in} and Σ_{out} two finite sets of input and output labels, respectively, and Π a labeling problem. Π is *locally checkable* with checking radius r if there exists a finite family $\mathcal{S} = \{((H, v_H), \bar{x}, \bar{y})_i\}_{i \in I}$ of tuples, where (H, v_H) is a centered graph of radius at most r and maximum degree at most Δ, $\bar{x} : V(H) \to \Sigma_{\text{in}}$ is an input labeling for H, $\bar{y} : V(H) \to \Sigma_{\text{out}}$ is an output labeling for H (which can depend on $\bar{x}$) with the following property:

- For any input (G, x) to Π, an output function $y : V(G) \to \Sigma_{\text{out}}$ is permissible (i.e., $y \in \Pi((G, x))$) if and only if, for each node $v \in V(G)$, the tuple $((G[\mathcal{N}_r(v)]), x \restriction_{\mathcal{N}_r(v)}, y \restriction_{\mathcal{N}_r(v)})$ belongs to $\mathcal{S}$.

To put it differently, in an LCL a solution to one problem can be verified *locally* by every node. If all nodes are happy about the local solution, then the problem is *globally* solved. If at least one node is not happy about the local solution, then the global solution is not valid.

Example 1.3. The c-coloring problem is an LCL for any $c \geq 0$. Indeed, a node can check with its neighbors that the coloring is consistent: If all nodes are happy about their local solution, then the global solution is valid. On the other hand, if one node is not happy about the local solution, then the coloring is not proper. In order to be a real LCL, we don't require the problem to be solved on graphs of maximum degree greater than some fixed Δ.

Exercise 1.3. Prove that the maximal independent set (MIS) is also an LCL. An independent set is a set of nodes of a graph that do not share any common edge. MIS asks to find an independent set S so that adding any other node to S breaks the independent set property.

Exercise 1.4. Prove that the maximal matching problem (MM) is also an LCL. A matching is a collection of edges such that no two edges share a common endpoint. A matching M is maximal if adding any other edge to M breaks the matching property.

Exercise 1.5. Prove that the problem stated in Exercise 1.2 is also an LCL.

1.2 Indistinguishability argument

The indistinguishability argument is a lower bound technique that works for some LCL problems in the LOCAL model. The idea is that a T-round LOCAL algorithm cannot distinguish between good input instances and bad input instances that locally look good (up to neighborhoods of radius T). Let us explore this argument with an example.

1.2.1 Complexity of 2-coloring cycles

Deterministic LOCAL. Consider the problem of 2-coloring restricted to even cycles. By Remark 1.2 there exists a trivial $O(n)$-round LOCAL algorithm that can 2-color an even cycle of length n. Now we claim that the complexity of 2-coloring an even cycle of length n is also $\Omega(n)$. Let $G = (V, E)$ be a cycle of n nodes, with $V = [n]$ and $E = \{\{i, i+1\} \mid i \in [n-1]\} \cup \{\{1, n\}\}$.

Assume there is a det-LOCAL $\mathcal{A}$ algorithm that 2-colors even cycles in $T = n/2 - 1$. Now take an odd cycle C_{n+1} of length $n+1$ and run $\mathcal{A}$ in C_{n+1} providing to $\mathcal{A}$ the number n as the number of total nodes of the graph: we are lying to the algorithm. Of course $\mathcal{A}$ fails to 2-color C_{n+1}, meaning that two adjacent nodes u, v will share the same color at the end. Now consider the subgraph of C_{n+1} induced by the radius-T neighborhood of $\{u, v\}$: it is a path of n nodes. Add an edge between the two nodes at distance T from of $\{u, v\}$. Now we have an even cycle C_n in which $\mathcal{A}$ produces the same local failure in $\{u, v\}$ because the input for nodes u, v is the same in C_n and C_{n+1}, hence contradicting the hypothesis that $\mathcal{A}$ was 2-coloring even cycles and implying a lower bound of $\Omega(n)$.

Randomized LOCAL. In the case of the rand-LOCAL model, proceeding like we did before does not imply a significant lower bound: we know that in C_{n-1} there is a failure with probability 1, but every specific local failure might occur with some small probability. Hence, we need to be a bit more clever.

Let us consider two subsets V_1, V_2 of the nodes of C_{n+1} as follows: $V_1 = [\lceil n/2 \rceil] \cup \{n+1\}$ and $V_2 = [n+1] \setminus [\lceil n/2 \rceil - 1]$. Trivially, $V_1 \cup V_2 = V$. Now assume there exists a T-round LOCAL algorithm $\mathcal{A}$ that 2-colors cycles with locality $T = \lfloor n/8 \rfloor$.[2] Notice that the radius-T neighborhood of V_i is strictly included in C_{n+1} for $i = 1, 2$, hence it is 2-colorable. We apply $\mathcal{A}$ to C_{n+1}: it holds that there is at least an $i^\star \in \{1, 2\}$ such that $\mathcal{A}$ fails in $C_{n+1}[V_{i^\star}]$ with probability no less than $1/2$. We could already be satisfied that in $C_{n+1}[V_{i^\star}]$ there is such high failure probability, but we can do more. Consider now a new graph G consisting in M disjoint copies of C_{m+1}, with M, m being even numbers: G has $n = M \cdot (m+1)$ nodes. Assume there is a randomized LOCAL algorithm, $\mathcal{A}$ that 2-colors even cycles with locality T no greater than $\lfloor m/8 \rfloor = \lfloor n/(8M) \rfloor$. From what we said before, we know that for each cycle there is a subgraph over which $\mathcal{A}$ fails with probability no less than $1/2$. We take such subgraphs and their radius-T neighborhood induced subgraphs H_i, for $i = 1, \ldots, M$, and we can connect them, together with the remaining, unused nodes of G, so that they form an even cycle C_n with n nodes. Let us compute the probability that $\mathcal{A}$ fails over this new cycle. Let E_i be the event that $\mathcal{A}$ fails over H_i. Then, the following holds.

$$\begin{aligned} \Pr\left[\mathcal{A} \text{ fails on } C_n\right] &\geq \Pr\left[\cup_{i \in [M]} E_i\right] \\ &= 1 - \Pr\left[\cap_{i \in [M]} \bar{E}_i\right] \\ &\geq 1 - (1 - 1/2)^M \\ &= 1 - 1/2^M, \end{aligned} \tag{1}$$

[2] The constants are not optimized: we just care about the asymptotic behavior.

where Eq. (1) holds by independence. Hence, to get lower failure probability an algorithm must have locality at least $\Omega(n/M)$.

1.2.2 General statement

The argument can be stated for general LCLs provided that some conditions are met.

First, we define the notion of subgraph cover.

Definition 1.3. A *subgraph cover* of a graph G is a family of subgraphs $\{G_i\}_{i\in I}$ such that $G_i \subseteq G$ for all $i \in I$ and $\cup_{i\in I} G_i = G$.

Suppose we have an LCL problem Π, with checking radius $r \geq 0$, that is solvable over some graph family $\mathcal{F}$, and assume we have a rand-LOCAL algorithm $\mathcal{A}$ that solves Π over $\mathcal{F}$ and has running time $T(n) \geq r > 0$, n being the size of the input graph. Suppose there exists a graph $G_n \notin \mathcal{F}$ such that Π is not solvable over G_n, in which we want to run $\mathcal{A}$ for $T(n)$ rounds over G_n (G_n does not have necessarily size n—we force outputs after time $T(n)$ if the protocol did not produce any or if, at any time, at any node the computational procedure is not well-defined). Assume G_n admits a subgraph cover $\{G_n^{(i)}\}_{i\in I}$ with the following properties:

(1) For each $v \in G_n$, there exists $i_v \in I$ such that $\mathcal{N}_r(v) \in G_n^{(i_v)}$.

(2) For each $i \in I$, there exists a graph $H_n^{(i)} \in \mathcal{F}$ of size n which contains a subgraph $\tilde{H}_n^{(i)}$ such that $\tilde{H}_n^{(i)}$ is isomorphic to $G_n^{(i)}$, and $H_n^{(i)}[\mathcal{N}_{T(n)}(\tilde{H}_n^{(i)})]$ is isomorphic to $G_n[\mathcal{N}_{T(n)}(G_n^{(i)})]$.

We run $\mathcal{A}$ on G_n: G_n will contain at least one radius-r neighborhood that has a non-admissible output assignation. As $\{G_n^{(i)}\}_{I\in I}$ is a subgraph cover, there exists $i^\star \in I$ such that the probability of $\mathcal{A}$ failing over $G_n^{(i^\star)}$ is at least $1/|I|$. We denominate $G_n^{(i^\star)}$ the *bad subgraph*. Hence, assuming $H_n^{(i)}[\mathcal{N}_{T(n)}(\tilde{H}_n^{(i)})]$ and $G_n[\mathcal{N}_{T(n)}(G_n^{(i)})]$ are given the same identifiers and input labels, $\mathcal{A}$ fails on $H_n^{(i^\star)}$ with probability at least $1/|I|$. We observe that property (2) is sufficient but not necessary to the technique: it is sufficient to ensure the existence of the graph $H_n^{(i)}$ for $i = i^\star$. Nevertheless, in many practical scenarios actually determining $i^\star$ is hard, while it is easier to ensure (2) for many graph families.

This result is useful when $|I|$ is not too large: However, in many cases, it is not possible to find subgraph covers with few elements. Furthermore, one may want a failure probability that is higher than a constant value. Luckily, other properties of the rand-LOCAL model come to our aid and sometimes allow to boost the failing probability. Let $M \in \mathbb{N}$. The idea is to replicate M times the radius-$T(n)$ neighborhood of the bad subgraph (making sure we still obtain a graph that belongs to the graph family under consideration) and exploit the independence of the outcome generated by $\mathcal{A}$ over subsets of the nodes that are "far enough". Let us redefine the properties that the graph G_n needs to meet.

Definition 1.4 (Cheating graph)**.** Let Π be any LCL problem with checking radius $r \in \mathbb{N}$ that is solvable over some graph family $\mathcal{F}$. Suppose that, for infinitely many integer $n \in \mathbb{N}$, there exists a triple $(k, M, T) \in \mathbb{N}^3$ (that can depend on n), with $T \geq r$, and a graph G_n of size at most $\lfloor n/M \rfloor$, such that the following properties are met:

(i) Π is not solvable on G_n.

(ii) G_n has a subgraph cover $\{G_n^{(1)}, \dots G_n^{(k)}\}$ such that the following holds:

  (a) For each $v \in V(G_n)$, there exists $j \in [k]$ such that $\mathcal{N}_r(v) \subseteq V(G_n^{(j)})$;

  (b) For each choice of indices $\mathbf{x}_M = (x_1, \dots, x_M) \in [k]^M$, there exists a graph $H_{\mathbf{x}_M} \in \mathcal{F}$ of size n which contains a subgraph $\tilde{H}_{\mathbf{x}_M}$ such that $\tilde{H}_{\mathbf{x}_M}$ is isomorphic to the disjoint union $\bigsqcup_{j=1}^{M} G_n^{(x_j)}$, and $H_{\mathbf{x}_M}[\mathcal{N}_T(\tilde{H}_{\mathbf{x}_M})]$ is isomorphic to the disjoint union $\bigsqcup_{j=1}^{M} G_n[\mathcal{N}_T(G_n^{(x_j)})]$.

Then we say that the family $\{G_n\}_{n\in I}$ is a family of (k, M, T)-*cheating graphs* for the pair $(\Pi, \mathcal{F})$ or, more generally, that $\mathcal{F}$ admits a family of (k, M, T)-*cheating graphs* for Π.

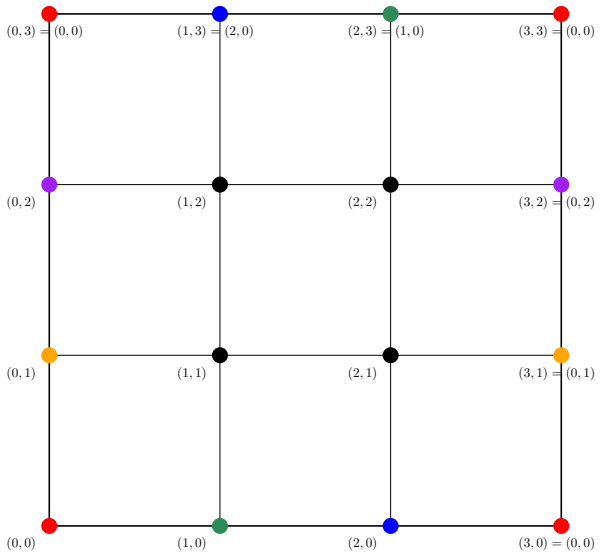


Figure 1: Representations of graphs G and H with $n = 3$. Nodes on the borders are identified in H as indicated by the same colors. Each face of the graph is oriented the same as the oriented circle inside it.

Remark 1.3. Being Π an LCL problem, Definition 1.4.(ii).(b) implies that Π is solvable on $G_n[\mathcal{N}_T(G_n^{(i)})]$ for $i = 1, \dots, k$.

In both arguments for the det-LOCAL and the rand-LOCAL model, we call the graph G_n the cheating graph because it allows us to "trick" the distributed algorithm, since nodes cannot distinguish between different inputs if they have the same local view. Then, we have the following theorem.

Theorem 1.1. *Consider any LCL problem Π with checking radius $r \geq 0$ and fix any family $\mathcal{F}$ of input graphs of arbitrarily large size. Suppose that for some integers $k, M \geq 1$ and non-decreasing function $T(n)$ there exists $(k, M, T(n))$-cheating graphs for $(\mathcal{P}, \mathcal{F})$. Then, any rand-LOCAL algorithm $\mathcal{A}$ solving $\mathcal{P}$ over $\mathcal{F}$ that runs with locality at most $T(n/M)$ has failure probability $1 - (1 - 1/k)^M$.*

Proof. Consider now a cheating graph G_n of n nodes for $(\Pi, \mathcal{F})$. Now, construct a new graph G' that consist in M copies of G_n. By contradiction, suppose that there is an algorithm that has locality $T(n)$ on graphs of $n \cdot M$ nodes and solves Π over $\mathcal{F}$. Let us run $\mathcal{A}$ on G': it will fail with probability 1 in each of the M copies of G_n. More specifically, there exist an index $\mathbf{x}_M \in [|I|^M]$ such that the probability of $\mathcal{A}$ failing on $G_n^{(\mathbf{x}_M[i])}$ on the i-th copy of G_n is at least $1/2$.

By independence, the probability that $\mathcal{A}$ solves Π over $H_{\mathbf{x}_M}$ is now $(1 - 1/|I|)^M$. Hence, we get a lower bound $T(n)$ for graphs of $n \cdot M$ nodes, which means that we get a lower bound of $T(n/M)$ for graphs of n nodes. □

Exercise 1.6. There are graphs that are known to be locally bipartite but have chromatic number 4 (Mohar, Simonyi, and Tardos [MST13]). The description of the graph follows: Let $n \in \mathbb{N}$ be any odd number. The graph $G = (V(G), E(G))$ has $V(G) = \{0, \dots, n\} \times \{0, \dots, n\}$, where two nodes (i, j) and (i', j') are connected by an edge if $|i - i'| + |j - j'| = 1$, i.e., their Manhattan distance is 1. Now, define the following equivalence relation for nodes: $(i, 0) \sim_V (n - i, n)$ for $i = 0, \dots, n$, and $(0, j) \sim_V (n, j)$ for $j = 0, \dots, n$. Now define a new graph $H = (V(H), E(H))$ where $V(H) = V(G)/\sim_V$ and $E(H)$ is characterized as follows: Take any two nodes $u, v \in V(H)$. Such nodes are equivalence classes for $\sim_V$. If there exists $u' \in u \subseteq V(G), v' \in u \subseteq V(G)$ such that $u'v' \in E(G)$, then $uv \in E(H)$. See Fig. 1 for a representation of G and H. The chromatic number of G is 4 as long as n is odd. Can we derive a lower bound for the problem of 3-coloring bipartite graphs?

2 Lecture 2

In Section 1 we have defined the LOCAL model of computation, the class of LCL problems, and we have gone through a classical lower bound technique: the indistinguishability argument In this lecture we will talk about quantum and superquantum models and how the indistinguishability argument relates to all this.

As the LOCAL model is the strongest possible model of synchronous distributed computation, it is natural to consider a quantum generalization of the model, where the nodes of the system are capable of quantum computation and quantum communication. More precisely, the quantum computers manipulate local states consisting of an unbounded number of qubits with arbitrary unitary transformation. The communication links are quantum communication channels (adjacent nodes can exchange any number of qubits), and the local output can be the result of any quantum measurement.

Now, the overarching question: can the quantum-LOCAL model *beat* the classical LOCAL model for natural problems? There has been a long line of research addressing quantum advantage in distributed models [Elk+14; LM18; MN22; WY22; WWY22; IL19; CH+22; ILM20; AV22], but almost nothing regarding the quantum-LOCAL model.

When we ask for quantum advantage, we usually mean *asymptotic advantage*: we do not care about constant factors in running times. We do know that the quantum-LOCAL model admits quantum advantage for a very specific, artificially crafted problem that is inherently *global*, i.e. in order to verify a solution, a node must gather the entire output of the graph [LNR19]. The very same authors of [LNR19] declare that this problem is of no interest for the distributed community: rather, it serves as a proof of concept that the quantum-LOCAL model might be arbitrarily stronger than the classical LOCAL model. Hence, we might ask a natural question: is there any LCL for which the quantum-LOCAL model admits asymptotic advantage with respect to the classical LOCAL model?

Interestingly, few results on the quantum-LOCAL model have been derived by analyzing superquantum models and providing suitable lower bounds. The first idea of analyzing superquantum models based on the pure causality principle dates back to 2009, where Gavoille, Kosowski, and Markiewicz [GKM09] introduced the φ-LOCAL model, also called the non-signaling LOCAL model in later works [AF14; Coi+24] (and here we adopt this nomenclature).

2.1 The non-signaling LOCAL model

We now introduce the non-signaling LOCAL model and start giving some results. The non-signaling LOCAL model produces *outcomes*: hence, we first define what an outcome is.

Definition 2.1 (Outcome)**.** Let Σ_{in} and Σ_{out} be two sets of input and output labels, respectively. An *outcome* $\mathcal{O}$ is a mapping $(G, x) \mapsto \{(y_i, p_i)\}_{i \in I}$, with I being a discrete set of indices, assigning to every input graph G with any input data $x : V(G) \to \Sigma_{\text{in}}$, a discrete probability distribution $\{p_i\}_{i \in I}$ over (not necessarily permissible) output functions $y_i : V(G) \to \Sigma_{\text{out}}$ such that:

1. for all $i \in I$, $p_i > 0$;
2. $\sum_{i \in I} p_i = 1$;
3. p_i represents the probability of obtaining y_i as the output assignment of the distributed system.

Notice that we are not taking into account the assignation of port numbers and unique identifiers: this is on purpose, as the output of an outcome can depend on ports and identifiers produced uniformly at random. An algorithm can be thought of producing an output distribution on every input: Whenever the computations of a node in a given round are defined, the algorithm proceeds normally. If at some round some computation is undefined for a node, the node outputs some "garbage label", say $\perp$: we remark that we can assume Σ_{out} always contains such a garbage label without loss of generalization. Hence, an outcome can be always thought of as being defined on the family of all graphs and all valid inputs: for this reason, we will omit specifying the graph family on which the outcome is defined.

We say that an outcome $\mathcal{O}$ over some graph family $\mathcal{F}$ *solves* problem Π over $\mathcal{F}$ *with probability* p if, for every $G \in \mathcal{F}$ and any input data (G, x), it holds that

$$\sum_{\substack{(y_i,p_i)\in\mathcal{O}((G,x)) \,:\\ y_i\in\Pi((G,x))}} p_i \geq p.$$

When $p = 1$, we will just say that $\mathcal{O}$ *solves* problem Π over the graph family $\mathcal{F}$.

In order to proceed, we first define the *non-signaling* property of an outcome. Let $T \geq 0$ be an integer, and I a set of indices. For any set of nodes V, subset $S \subseteq V$, and for any input $(G = (V, E), x)$, we define its T-*local view* $\mathcal{V}_T(G, x, S)$ as the radius-T neighborhood $G[\mathcal{N}_T(S)]$ of S in G together with all input data stored in the nodes of $\mathcal{N}_T(S)$. For future use, we extend the definition of graph isomorphisms to local views. Two local views $\mathcal{V}_T(G, x_G, S_G)$ and $\mathcal{V}_T(H, x_H, S_H)$ are isomorphic if the following properties are met:

1. $G[\mathcal{N}_T(S_G)]$ and $H[\mathcal{N}_T(S_H)]$ are isomorphic and the isomorphism restricted to $G[S_G]$ brings to $H[S_H]$.
2. Input data is invariant under the above isomorphism, i.e. if the isomorphism brings $u \in V(G)$ into $v \in V(H)$, then the respective input data are the same.

For any subset of nodes $S \subseteq V$ and any output distribution $\{(y_i, p_i)\}_{i\in I}$, we define the *marginal distribution* of $\{(y_i, p_i)\}_{i\in I}$ on set S as the unique output distribution $\{(\bar{y}_i, \bar{p}_i)\}_{i\in I}$ acting on S which satisfies the condition

$$\bar{p}_j = \sum_{i \,:\, \bar{y}_j = y_i[S]} p_i,$$

where $y_i[S]$ is the restriction of output y_i to the processes in S.

Definition 2.2 (Non-signaling outcome)**.** An outcome $\mathcal{O} : (G, x) \mapsto \{(y_i, p_i)\}_{i\in I}$ is *non-signaling* beyond distance T if for any two inputs $(G, x_G), (H, x_H)$ such that $|V(G)| = |V(H)|$, for any two subsets of nodes $S_G \subseteq V(G), S_H \subseteq V(H)$ such that $\mathcal{V}_T(G, x, S_G)$ is isomorphic to $\mathcal{V}_T(H, x_H S_H)$, the output distributions restricted to S_G and S_H are identical (under the isomorphism). Notice that T can depend on the input labeled graph.

Definition 2.2 is also the more general definition for the locality of an outcome: an outcome $\mathcal{O}$ has locality T if it T is the minimum parameter such that $\mathcal{O}$ is non-signaling beyond distance T.

The non-signaling LOCAL model. The non-signaling LOCAL model is a computational model that produces non-signaling outcomes. The complexity class $\mathcal{NS}[T]$ is defined by all pairs $(\Pi, \mathcal{F})$ where Π is a problem and $\mathcal{F}$ is a graph family such that there exists an outcome $\mathcal{O}$ that is non-signaling beyond distance T which solves Π over $\mathcal{F}$ with probability at least $1 - 1/\text{poly}(n)$ for graphs of n nodes.

As every (deterministic or randomized) algorithm running in time at most T in the LOCAL model produces an outcome which has locality T, we can provide lower bounds for the LOCAL model by proving them in the non-signaling LOCAL model. For the sake of readability, we assume that every outcome $\mathcal{O}$ that has locality T can be produced by a hypothetical non-signaling LOCAL algorithm $\mathcal{A}$ with running time T. This is just an artifact of the text and does not affect in any way the validity of our proofs.

Let us give an example of a lower bound in non-signaling LOCAL.

Lemma 2.1. *The consensus problem does not belong to* $\mathcal{NS}[0]$.

Proof. Consider an input graph that is a path of two nodes u_1, u_2. Assume also that the input opinions belong to the set $\{0, 1\}$. Assume that an outcome $\mathcal{O}$ solves the consensus problem with locality 0. Then we can describe this outcome just as a mapping from the (ordered) opinions of the two nodes two the output distribution. Hence, the outcome must work as follows:

1. $(0, 0) \mapsto \{((0, 0), 1)\}$.
2. $(1, 1) \mapsto \{((1, 1), 1)\}$.
3. $(0, 1) \mapsto \{((0, 0), p), ((1, 1), 1 - p)\}$.

4. $(1,0) \mapsto \{((0,0),q),((1,1),1-q)\}$.

Now, consider S to be the subset of only u_1. By the non-signaling property, the restriction of the outcome to S with inputs $(0,0)$ and $(0,1)$ implies that $p=1$. Now consider the subset $\bar{S}$ that contains only u_2. By the non-signaling property, the restriction of the outcome to $\bar{S}$ with inputs $(1,1)$ and $(0,1)$ implies that $p=0$, reaching a contradiction. □

We now want to argue that the indistinguishability argument can be extended all the way up to the non-signaling LOCAL model.

2.2 Indistinguishability argument in the non-signaling LOCAL model

The very first work introducing the non-signaling LOCAL model showed some applications of the indistinguishability to the non-signaling LOCAL model [GKM09].

Lemma 2.2 (2-coloring even cycles [GKM09])**.** *Let $2 \leq n \in \mathbb{N}$ be an even number. The problem of 2-coloring C_n belongs to $\mathcal{NS}[\lceil (n-2)/4 \rceil]$ but not to $\mathcal{NS}[\lceil (n-2)/4 \rceil - 1]$.*

Proof. Let $C_n = ([n], E_n)$, with $E_n = \{\{i, (i \mod n)+1\} \mid i \in [n]\}$ We first prove the lower bound. Consider the two nodes 1 and $j = 2\lceil (n-2)/4 \rceil$. Let $T = \lceil (n-2)/4 \rceil - 1$. Notice that the radius-T neighborhoods of 1 and j do not intersect. Now we change the topology of the graph so that the distance between i and j becomes $T+1$ (hence either i or j – but noth borh nodes – must have a different color in the colorings with respect to before). The outcome restricted to $\{i,j\}$ changes completely the probability of the colorings of $\{i,j\}$.

We now prove the upper bound. Consider the following outcome. To each cycle of even nodes $C_n = ([n], E_n)$, we assign a 2-coloring chosen uniformly at random. Let $T = \lceil (n-2)/4 \rceil$. Notice that for any two nodes $i, j \in [n]$, the radius-T neighborhood of i intersects the radius-T neighborhood of j. Hence, if we change the topology of the graphs outside the radius-T neighborhood of any subset of nodes $S \subseteq [n]$, the restriction of the outcome to S is still the uniform distribution over 2-colorings. □

The proof of the lower bound in Lemma 2.2 exploits a different kind of indistinguishability argument: there, the local views of two nodes that are distant enough cannot discriminate if the two nodes have odd or even distance, implying two different 2-colorings. For any choice of local coloring, we can make the 2-coloring *propagate badly*. This is an example of *propagation argument*. Propagation arguments can work in non-signaling, but it is less clear what happens for graph-existential indistinguishability argument. However, with some care, we can prove that everything can still work properly.

The same definition of cheating graph holds. We now present our general lower bound theorem.

Theorem 2.1 (Graph-existential indistinguishability argument in non-signaling LOCAL [Coi+24])**.** *Let Π be an LCL problem with checking radius r, and $\mathcal{F}$ be a graph family that admits a family of (k,M,T)-cheating graphs for Π. Suppose $\mathcal{O}$ is an outcome over $\mathcal{F}$ in non-signaling LOCAL with locality $T \geq r$. Then, for infinitely many n, there exists a graph $H \in \mathcal{F}$ on n vertices such that the probability of $\mathcal{O}$ solving Π on H is at most $(1-1/k)^M$. Furthermore, H can be chosen among the graphs in the family $\{H_{\mathbf{x}_M}\}_{\mathbf{x}_M \in [k]^M}$ given by Definition 1.4.(ii).(b).*

Proof. Let G_n be a (k,M,T)-cheating graph for $(\Pi, \mathcal{F})$. We know that G_n has size at most $\lfloor n/M \rfloor$ and satisfies the properties listed in Definition 1.4. Now, consider a new graph that consists of M disjoint copies $G_{n,1}, \ldots, G_{n,M}$ of G_n, and $n - |V(G_n)|$ isolated nodes.

For each $i = 1, \ldots, M$, consider the subgraph cover $\{G_{n,i}^{(j)}\}_{j \in [k]}$ for $G_{n,i}$ given by Definition 1.4. Let $\mathcal{O}$ be any outcome having locality T and solving problem Π over $\mathcal{F}$.

As Π is not solvable on $G_{n,i}$, then the failing probability of $\mathcal{O}$ over $G_{n,i}$ is 1, for each $i = 1, \ldots, M$. Consider one of the $G_{n,i}$ and notice that, if $\mathcal{O}$ produces a permissible output function on $G_{n,i}[\mathcal{N}_T(G_{n,i}^{(j)})]$ for each $j \in [k]$ at the same time, then, by definition of LCL problem, we have a global permissible output function on $G_{n,i}$ (which does not exist). Hence, by Definition 1.4.(ii).(a), there must exist $j \in [k]$ and $v \in V(G_{n,i}^{(j)})$ such that $\mathcal{N}_r(v) \subseteq V(G_{n,i}^{(j)})$ and the output function on $\mathcal{N}_r(v)$ is not permissible: in such a case, we say that $G_{n,i}^{(j)}$ contains a *bad node*.

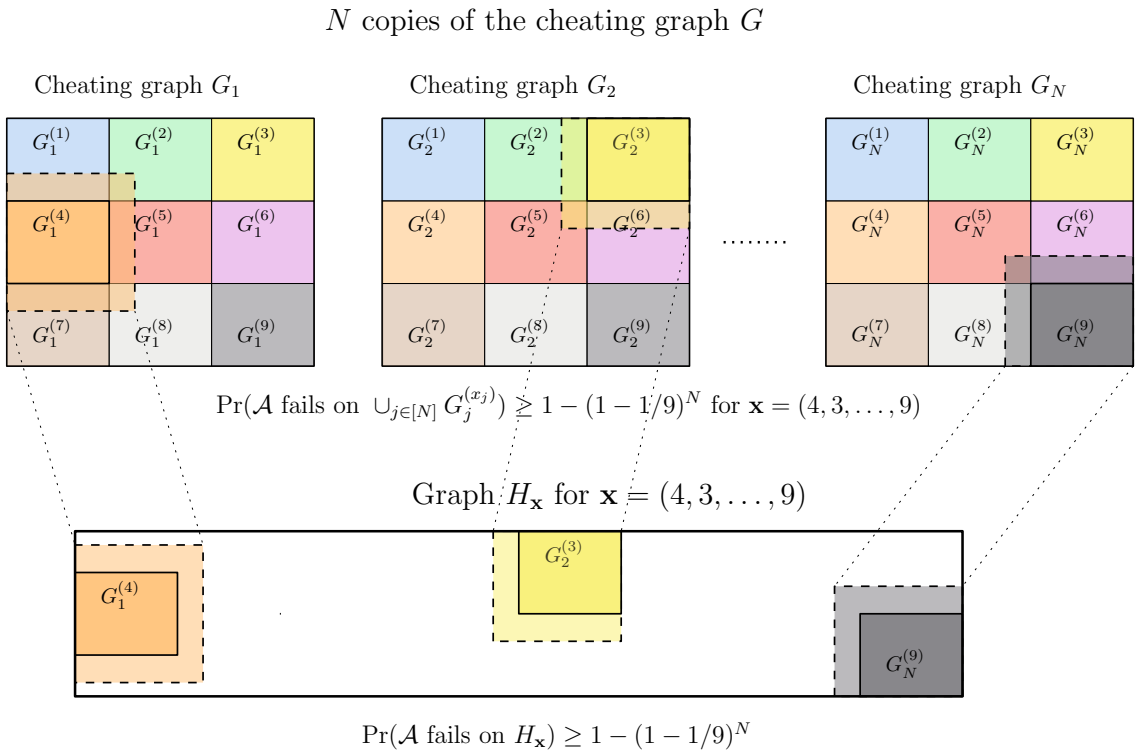


Figure 2: Representation of the indistinguishability argument technique in non-signaling LOCAL.

We now prove that there exists a sequence of indices $\mathbf{x}_M = (x_1, \ldots, x_M) \in [k]^M$ such that $\mathcal{O}$ produces a bad node in $\bigsqcup_{j=1}^{M} G_{n,j}^{(x_j)}$ with probability at least $1-(1-1/k)^M$. If we had independence between "far away" parts of the graphs (as in the randomized LOCAL model), this thesis would be trivial. However, in the non-signaling LOCAL model non-trivial dependencies are possible (e.g., pre-shared quantum state).

We here present a short proof by induction on M. Assume $M=1$: as $\mathcal{O}$ fails on $G_{n,1}$ with probability 1, and the latter is covered by $\{G_{n,1}^{(j)}\}_{j=1}^{k}$, then there exists an index $x_1 \in [k]$ such that the probability that $G_{n,1}^{(x_1)}$ contains a bad node is at least $1/k$. Mow, assume $M>1$ and the claim to be true for $M-1$. Let $\mathcal{E}_i^{(j)}$ be the event that $\mathcal{O}$ produces a bad node in $G_{n,i}^{(j)}$ (we remark that Π is solvable on $G_{n,i}^{(j)}$ by Remark 1.3): the inductive hypothesis can be rewritten as $\Pr\left[\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)}\right] = 1-(1-1/k)^{M-1}+y$ for some $y \geq 0$. Assume $\Pr\left[\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)}\right] < 1$ otherwise the thesis is trivial. The key argument is to show that there exists $x_M \in [k]$ with

$$\Pr\left[\mathcal{E}_M^{(x_M)} \;\middle|\; \cap_{i=1}^{M-1}\bar{\mathcal{E}}_i^{(x_i)}\right] \geq 1/k. \tag{2}$$

If so, then, by the inclusion-exclusion principle and the law of total probability,

$$\begin{aligned}
\Pr\left[\cup_{i=1}^{M}\mathcal{E}_i^{(x_i)}\right] &= \Pr\left[\mathcal{E}_M^{(x_M)}\right] + \Pr\left[\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)}\right] - \Pr\left[\mathcal{E}_M^{(x_M)} \bigcap (\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)})\right] \\
&= \Pr\left[\mathcal{E}_M^{(x_M)} \bigcap (\cap_{i=1}^{M-1}\bar{\mathcal{E}}_i^{(x_i)})\right] + \Pr\left[\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)}\right] \\
&= \Pr\left[\cup_{j=1}^{k}\mathcal{E}_M^{(j)} \;\middle|\; \cap_{i=1}^{M-1}\bar{\mathcal{E}}_i^{(x_i)}\right] \Pr\left[\cap_{i=1}^{M-1}\bar{\mathcal{E}}_i^{(x_i)}\right] + \Pr\left[\cup_{i=1}^{M-1}\mathcal{E}_i^{(x_i)}\right] \\
&\geq \frac{1}{k}\cdot\left[\left(1-\frac{1}{k}\right)^{M-1} - y\right] + 1 - \left(1-\frac{1}{k}\right)^{M-1} + y \\
&= 1-\left(1-\frac{1}{k}\right)^{M} + y\left(1-\frac{1}{k}\right)
\end{aligned}$$

$$\geq 1 - \left(1 - \frac{1}{k}\right)^M.$$

By Definition 1.4.(ii).(b), there exists a graph $H_{\mathbf{x}_M} \in \mathcal{F}$ over the same set of n nodes that contains a subgraph $\tilde{H}_{\mathbf{x}_M}$ with $H_{\mathbf{x}_M}[\mathcal{M}_T(\tilde{H}_{\mathbf{x}_M})]$ being the same graph as $\bigsqcup_{i=1}^{M} G_{n,i}[\mathcal{N}_T(G_{n,i}^{(x_i)})]$. Consider the same identifiers and input labels over $H_{\mathbf{x}_M}[\mathcal{N}_T(\tilde{H}_{\mathbf{x}_M})]$ and $\bigsqcup_{i=1}^{M} G_{n,i}[\mathcal{N}_T(G_{n,i}^{(x_i)})]$: by the definition of non-signaling LOCAL, the probability that $\mathcal{O}$ fails on $\tilde{H}_{\mathbf{x}_M} \subseteq H_{\mathbf{x}_M}$ is the same as that on $\bigsqcup_{j=1}^{M} G_{n,j}^{(x_j)}$, yielding the thesis.

Let us now prove Eq. (2). As $\mathcal{O}$ fails on $G_{n,M}$ with probability 1, we know that

$$\begin{aligned} 1 &= \Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)}\right] \\ &= \Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)} \bigcup (\cup_{i=1}^{M-1} \mathcal{E}_i^{(x_i)})\right]. \end{aligned}$$

By the law of total probability, we get that

$$\begin{aligned} 1 &= \Pr\left[(\cup_{j=1}^{k} \mathcal{E}_M^{(j)}) \bigcup (\cup_{i=1}^{M-1} \mathcal{E}_i^{(x_i)})\right] \\ &= \Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)} \;\middle|\; \cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right] \Pr\left[\cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right] + \Pr\left[\cup_{i=1}^{M-1} \mathcal{E}_i^{(x_i)}\right]. \end{aligned}$$

Hence,

$$\begin{aligned} \Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)} \;\middle|\; \cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right] &= \frac{1 - \Pr\left[\cup_{i=1}^{M-1} \mathcal{E}_i^{(x_i)}\right]}{\Pr\left[\cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right]} \\ &= 1. \end{aligned}$$

We know that $\Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)} \;\middle|\; \cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right] = 1$ as $\Pr\left[\cup_{j=1}^{k} \mathcal{E}_M^{(j)}\right] = 1$ and the conditioning event has non-zero probability; by the union bound, it follows there exists $x_M \in [k]$ with

$$\Pr\left[\mathcal{E}_M^{(x_M)} \;\middle|\; \cap_{i=1}^{M-1} \bar{\mathcal{E}}_i^{(x_i)}\right] \geq 1/k,$$

proving Eq. (2). □

As long as one can find a cheating graph for a pair $(\Pi, \mathcal{F})$, where Π is an LCL problem and $\mathcal{F}$ an input graph family, the lower bound technique can be applied. In Fig. 2 we depict how we use the cheating graph to create a bad, admissible input instance that yields the lower bound.

3 Lecture 3

In this section, we first exhibit an application of the indistinguishability argument to an important coloring problem, and then we show how to bound quantum advantage for a specific complexity class in rooted trees.

3.1 Lower bound for approximate graph coloring

The problem of interest is approximate graph coloring: given two integers $2 \leq \chi \leq c$ and a χ-chromatic graph G in input, we are required to output a c-coloring of G. In this subsection we prove the following theorem.

Theorem 3.1. *Let $\chi \geq 2$, $c \geq \chi$ be integers, and let $\alpha = \left\lfloor \frac{c-1}{\chi-1} \right\rfloor$. Let $\varepsilon \in (0, \frac{\alpha-1}{\alpha})$ be a real value, and let $n \in \mathbb{N}$ with*

$$n \geq \left\lceil \frac{\log \varepsilon^{-1}}{\log\left(1 + \frac{1}{\alpha}\right)} \right\rceil \cdot \frac{(6\chi+1)^{\alpha+1} - 1}{6}.$$

Suppose $\mathcal{A}$ is a non-signaling LOCAL algorithm for c-coloring graphs in the family $\mathcal{F}$ of χ-chromatic graphs of n nodes with success probability $q > \varepsilon$. Then the running time of $\mathcal{A}$ is at least

$$T = \Omega\left(\frac{1}{\chi^{1+\frac{1}{\alpha}}} \cdot \left(\frac{n}{\log \varepsilon^{-1}}\right)^{\frac{1}{\alpha}}\right).$$

The results on [Bog13, Theorem 1.2], a result that has gone relatively unnoticed and lies at the intersection between graph theory, combinatorics, and topology, which ensures the existence of a graph with high chromatic number which, locally, is χ-chromatic. The first half of the sections aims at constructing such graph, and the second half is devoted to the proof that this graph admits a (small enough) subgraph cover that satisfies the properties of Definition 1.4.

We first define some graph operations. For any two graphs G, H, we define the intersection graph $G \cap H$ as a graph whose vertex set is the set $V(G) \cap V(H)$, and whose edge set is the set $E(G) \cap E(H)$. Similarly, we define the union graph $G \cup H$ as a graph whose vertex set is the set $V(G) \cup V(H)$, and whose edge set is the set $E(G) \cup E(H)$. We define the difference graph $G \setminus H$ as the subgraph of G induced by $V(G) \setminus V(H)$. The *T-local chromatic number* of a graph G, denoted by $\mathcal{LX}_T(G)$, is the minimum $c \in \mathbb{N}$ such that the graph induced by the T-neighborhood of any node is c-colorable. More formally

$$\mathcal{LX}_T(G) = \min\left\{c \in \mathbb{N} \mid \forall u \in V,\, G[\mathcal{N}_t(u)] \text{ is } c\text{-colorable}\right\}.$$

Given two graphs G and H, a function $f : V(G) \to V(H)$ is a homomorphism from G to H if, for any $\{u, v\} \in E(G)$, $\{f(u), f(v)\} \in E(H)$. A homomorphism from G to K_c, the c-clique, is equivalent to saying that G is c-colorable. Notice that the composition of homomorphisms is a homomorphism: hence, if G is homomorphic to H, then $\mathcal{X}(H) \geq \mathcal{X}(G)$. Furthermore, we define the *tensor product* of graphs G and H as a graph $G \times H$ whose vertex set is $V(G) \times V(H)$, and whose edge set is determined by the following: for any $(g, h), (g', h') \in V(G \times H)$, $\{(g, h), (g', h')\} \in E(G \times H)$ iff $gg' \in E(G)$ and $hh' \in E(H)$ (see Fig. 3 for an example).

We hereby state [Bog13, Theorem 1.2].

Theorem 3.2 ([Bog13])**.** *Let $\chi \geq 2$, $r \geq 2$, and $k \geq 1$ be integers. There exists a graph $G_k = (V, E)$ such that $\mathcal{LX}_r(G_k) = \chi$ and $\mathcal{X}(G_k) \geq k(\chi - 1) + 1$ with*

$$|V| = \frac{(2r\chi + 1)^k - 1}{2r}.$$

Remark 3.1. [Bog13, Theorem 1.2] has been stated for $\chi \geq 3$ since the result for $\chi = 2$ was already known from a different construction provided by [Sti85] (a proof translated in English was reproduced by [GJS04]). Nevertheless, the proof of [Bog13, Theorem 1.2] also holds for the case $\chi = 2$.

Remark 3.2. Theorem 3.2 holds even for $r \geq 1$, as explicitly mentioned at the end of [Bog13, Section 4], slightly changing the proof.

We proceed proving Theorem 3.1. The idea of the whole proof is to show that the graph from Theorem 3.2 provides a cheating graph for the c-coloring graphs problem and the family of χ-chromatic graphs.

We now construct the graph from Theorem 3.2.

The r-join of graphs. Given two graphs G and H, we aim to define the r-join operation $G \star_r H$, for any $r \geq 0$. The vertex set $V(G \star_r H)$ is defined by $V(G \star_r H) = V(G) \cup V(G) \times V(H) \times [r] \cup V(H)$. Let $E_{1,r}(G \star_r H)$ be equal to

$$\left\{\{(g, h, i), (g', h', j)\} \in \binom{V(G) \times V(H) \times [r]}{2} \;\middle|\; gg' \in E(G), hh' \in E(H), |i - j| \leq 1\right\}.$$

Furthermore, let

$$E_0(G \star_r H) = E(G) \cup \{\{g, (g', h', 1)\} \mid g, g' \in V(G), h' \in V(H), gg' \in E(G)\}$$

and

$$E_{r+1}(G \star_r H) = E(H) \cup \{\{(g, h, r + 1), h'\} \mid g \in V(G), h, h' \in V(H), hh' \in E(H)\}.$$

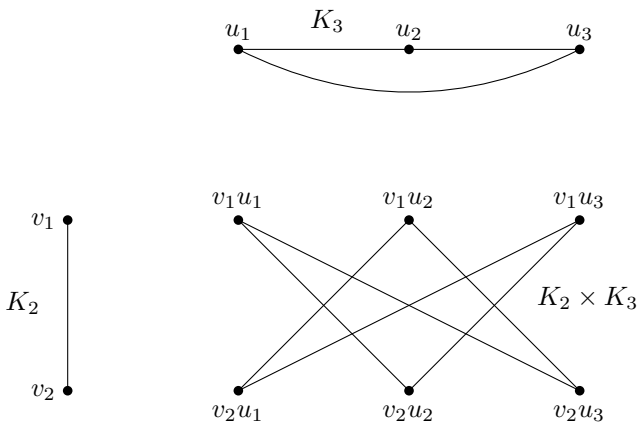


Figure 3: Tensor product $K_2 \times K_3$.

Then, the edge set $E(G \star_r H)$ is defined by $E(G \star_r H) = E_0(G \star_r H) \cup E_{1,r}(G \star_r H) \cup E_{r+1}(G \star_r H)$.

An intuitive visualization of this graph follows: take a sequence of $r+2$ disjoint copies $(G \times H)_0, (G \times H)_1, \ldots, (G \times H)_{r+1}$ of the tensor product $G \times H$ (an example of a tensor product graph is given in Fig. 3). Clearly, for any $0 \leq i \leq r$, there is an isomorphism $f_i : (G \times H)_i \to (G \times H)_{i+1}$. Then, any two nodes $(g,h) \in (G \times H)_i$ and $(g',h') \in (G \times H)_{i+1}$ are connected if and only if $\{f_i((g,h)), (g',h')\} \in E((G \times H)_{i+1})$, or, equivalently, $\{(g,h), f_i^{-1}((g',h'))\} \in E((G \times H)_i)$. Finally, "collapse" $(G \times H)_0$ into G and $(G \times H)_{r+1}$ into H by merging nodes (merging multiple edges and deleting self-loops).

This join operation in graphs is some kind of "discrete" variant of the join operation between two topological spaces (see [Bog13] or [Mat08]).

We define two projection operators for the join of graphs.

Definition 3.1. Let $\mathrm{pr}_G : G \star_r H \setminus H \to G$ and $\mathrm{pr}_H : G \star_r H \setminus G \to H$ be defined as follows: $\mathrm{pr}_G((g,h,i)) = g$ and $\mathrm{pr}_H((g,h,i)) = h$ for $(g,h,i) \in V(G) \times V(H) \times \{1,\ldots,r\}$, while $\mathrm{pr}_G \restriction_G$ and $\mathrm{pr}_H \restriction_H$ are the identity maps on G and H, respectively.

Remark 3.3. The two projections are homomorphisms, implying that the chromatic number of $G \star_r H \setminus H$ is the same as that of G, and the chromatic number of $G \star_r H \setminus G$ is the same as that of H.

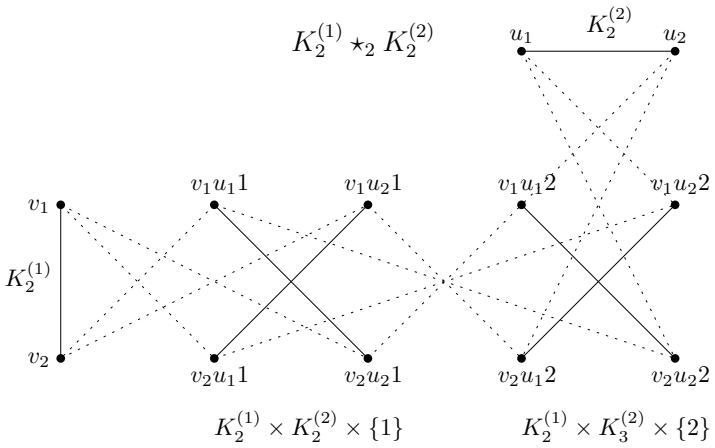
The join of two connected graphs results in a connected graph.

Lemma 3.1. *Let $r \geq 1$, and let G and H be two connected graphs with at least two nodes each. Then, $G \star_r H$ is connected.*

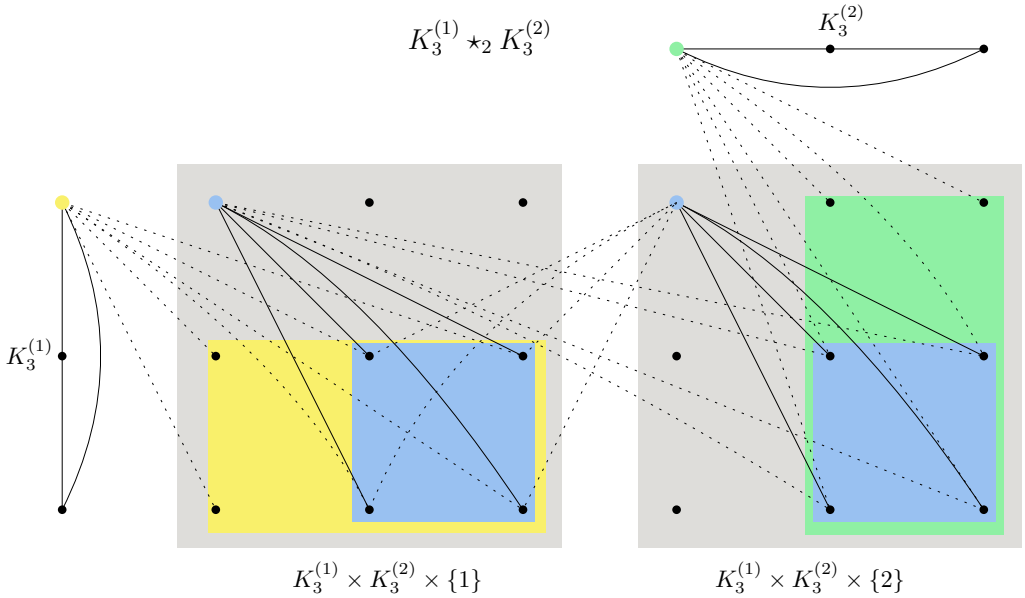
Proof. Consider any node u in $G \star_r H$ which does not belong to $G \cup H$. Then $u = (g,h,i)$ for $g \in V(G)$, $h \in V(H)$, and $i \in [r]$. There exist $g' \in V(G)$ and $h' \in V(H)$ such that $gg' \in E(G)$ and $hh' \in E(H)$. We now construct two paths $v_0 v_1 \ldots v_i$ and $w_{r+1} w_r \ldots w_i$ that connect G and H to $v_i = w_i = u$, respectively. Suppose i is odd. Then, set $v_0 = g'$, $v_j = (g,h,j)$ for any odd j, and $v_j = (g',h',j)$ for any even j. We have that v_j is connected to v_{j+1} for any $0 \leq j \leq i-1$, and $v_i = u$. Similarly, set $w_{r+1} = h'$, $w_j = (g,h,j)$ for any odd j, and $w_j = (g',h',j)$ for any even j. We have that w_j is connected to w_{j-1} for any $i+1 \leq j \leq r+1$, and $w_i = u$. Suppose i is even. Then, set $v_0 = g$, $v_j = (g',h',j)$ for any odd j, and $v_j = (g,h,j)$ for any even j. We have that v_j is connected to v_{j+1} for any $0 \leq j \leq i-1$, and $v_i = u$. Similarly, set $w_{r+1} = h$, $w_j = (g',h',j)$ for any odd j, and $w_j = (g,h,j)$ for any even j. We have that w_j is connected to w_{j-1} for any $i+1 \leq j \leq r+1$, and $w_i = u$. □

We are ready to define the graph from Theorem 3.2 which we will prove to be a cheating graph for the family of χ-chromatic graphs.

Definition 3.2 (The construction)**.** Let $\chi \geq 2$, $r \geq 2$, and $k \geq 1$. Consider a sequence $K_\chi^{(1)}, \ldots, K_\chi^{(k)}$ of k disjoint copies of K_χ. We construct the graph recursively. Let $G_1 = K_\chi^{(1)}$ be the clique with χ nodes. Then,



(a) The 2-join of $K_2^{(1)}$ and $K_3^{(2)}$, with all the connections. Full lines represent edges within the tensor product graphs plus the starting and ending graph. Dotted lines represent edges among these graphs. $K_2^{(1)} \times K_2^{(2)} \times \{1\}$ and $K_2^{(1)} \times K_2^{(2)} \times \{2\}$ are two copies of the tensor product $K_2^{(1)} \times K_2^{(2)}$.



(b) Representation of the 2-join of two copies $K_3^{(1)}$ and $K_3^{(2)}$ of K_3. Here, for the sake of visibility, only some edges are represented. The yellow node in $K_3^{(1)}$ is connected to all nodes in the yellow area of $K_3^{(1)} \times K_3^{(2)} \times \{1\}$; similarly, the green node in $K_3^{(2)}$ is connected to all nodes in the green area of $K_3^{(1)} \times K_3^{(1)} \times \{2\}$. The blue nodes of $K_3^{(1)} \times K_3^{(2)} \times \{1,2\}$ are connected to all nodes in the blue areas of $K_3^{(1)} \times K_3^{(2)} \times \{1,2\}$. Other connections can be deduced by symmetry.

Figure 4: Examples for the 2-join of graphs.

for $k \geq 2$, $G_k = G_{k-1} \star_{2r} K_\chi^{(k)}$. [Bog13] proved that $\mathcal{LX}_r(G_k) = \chi$, $\mathcal{X}(G_k) \geq k(\chi-1)+1$, and

$$|V(G_k)| = \frac{(2r\chi+1)^k - 1}{2r}.$$

See Fig. 4 for some examples (notice that $r = 1$ in the examples: as observed in Remark 3.2, the result still holds in this case).

In order to continue, we state the following lemma for an induced subgraph, whose proof is trivial.

Lemma 3.2. *Let $T \in \mathbb{N}$. Let G be a connected graph, and $H \subseteq G$ be a connected subgraph of G. Then, $G[\mathcal{N}_T(H)]$ is connected.*

For a graph G and any node $v \in V(G)$, we define by $\text{dist}_G(v, H) = \min_{u \in V(H)} \{\text{dist}_G(u, v)\}$ the distance between v and any subgraph $H \subseteq G$. We write $\text{dist}(v, H)$ when the underlying graph G is clear from the context. We now prove some key-properties of G_k which allow us to show that G_k is a cheating graph.

Lemma 3.3. *Let $\chi \geq 2$, $r \geq 3$, and $k \geq 2$ be integers. Define $T = \lfloor \frac{2r}{3} \rfloor$. Let G_k be the graph defined in Definition 3.2 built with copies of K_χ. There exists a subgraph cover $\{G_k^{(i)}\}_{i \in [k]}$ of G_k such that the following statements hold:*

(i) The chromatic number of $G_k[\mathcal{N}_T(G_k^{(i)})]$ is χ for all $i \in [k]$;

(ii) $G_k^{(i)}$ is connected for all $i \in [k]$;

(iii) for each $v \in V(G_k)$, there exists $i \in [k]$ such that $\mathcal{N}_1(v) \subseteq V(G_k^{(i)})$;

(iv) $G_k[\mathcal{N}_T(G_k^{(i)})]$ contains at least one node at distance T from $G_k^{(i)}$ for all $i \in [k]$.

Proof. We prove the thesis by induction on k. Remember that G_2 is obtained by the $2r$-join of two disjoint copies $K_\chi^{(1)}$, $K_\chi^{(2)}$ of K_χ. Assume $K_\chi^{(1)}$ is connected to the first copy of $K_\chi^{(1)} \times K_\chi^{(2)}$, and $K_\chi^{(2)}$ to the last. Let

$$V_2^{(1)} = V(K_\chi^{(1)}) \cup V(K_\chi^{(1)}) \times V(K_\chi^{(2)}) \times \{1, \dots, T+2\};$$
$$V_2^{(2)} = V(K_\chi^{(2)}) \cup V(K_\chi^{(1)}) \times V(K_\chi^{(2)}) \times \{T+1, \dots, 2r\}.$$

Consider the graphs $G_2^{(1)} = G_2[V_2^{(1)}]$ and $G_2^{(2)} = G_2[V_2^{(2)}]$. The cover property and properties (i)-(iii) are straightforward. Clearly, $G_2 = \cup_{i \in [2]} G_2^{(i)}$. Furthermore, $\mathcal{X}(G_2[\mathcal{N}_T(G_2^{(1)})]) = \mathcal{X}(G_2[\mathcal{N}_T(G_2^{(2)})]) = \chi$ as the projections $\text{pr}_{K_\chi^{(1)}}[\mathcal{N}_T(G_2^{(1)})], \text{pr}_{K_\chi^{(2)}}[\mathcal{N}_T(G_2^{(2)})]$ are homomorphisms. Moreover, it is easily verifiable that $G_2^{(1)}$ and $G_2^{(2)}$ are connected graphs: Observe that $G_2^{(1)} = (K_\chi^{(1)} \star_{2r} K_\chi^{(2)})[\mathcal{N}_{T+2}(K_\chi^{(1)})]$. As both $K_\chi^{(1)}$ (by the inductive hypothesis) and $K_\chi^{(2)}$ are connected, Lemma 3.1 implies their join is connected. Then, Lemma 3.2 implies $G_2^{(1)}$ is connected. The same applies for $G_2^{(2)}$ by observing that $G_k^{(2)} = (K_\chi^{(1)} \star_{2r} K_\chi^{(2)})[\mathcal{N}_{2r-T}(K_\chi^{(2)})]$. As for property (iv), consider any two nodes $u \in V(K_\chi^{(1)}) \times V(K_\chi^{(2)}) \times \{2T+2\}$ and $v \in V(K_\chi^{(1)}) \times V(K_\chi^{(2)}) \times \{1\}$. Clearly, $u \in G_2[\mathcal{N}_T(G_2^{(1)})]$ and has distance T from $G_2^{(1)}$, while $v \in G_2[\mathcal{N}_T(G_2^{(2)})]$ and has distance T from $G_2^{(2)}$.

Let $k \geq 3$ and assume the thesis is true for G_{k-1}. We now construct the subgraph cover of $G_k = G_{k-1} \star_{2r} K_\chi$. For $i = 1, \dots, k-1$, we define $G_k^{(i)}$ by

$$G_k^{(i)} = G_{k-1}^{(i)} \cup \left(G_k[\mathcal{N}_{T+2}(G_{k-1}^{(i)})] \cap G_k[V(G_{k-1}^{(i)}) \times V(K_\chi^{(k)}) \times \{1, \dots, T+2\}] \right)$$

Then, we define

$$G_k^{(k)} = G_k[V(G_{k-1}) \times V(K_\chi^{(k)}) \times \{T+1, \dots, 2r\}] \cup K_\chi^{(k)}.$$

We now prove that the family $\{G_k^{(i)}\}_{i \in [k]}$ respects properties (i)-(iv).

Subgraph covering. Any node $v \in (V(G_{k-1}) \cup V(K_\chi^{(k)}))$ belongs either to $(\cup_{i\in[k-1]} G_k^{(i)})$ (by observing that the latter contains $V(G_{k-1})$ and by using the inductive the hypothesis on the subgraph covering) or to $G_k^{(k)}$ (which contains $V(K_\chi^{(k)})$ by construction). Consider any node $v \in V(G_k) \setminus (V(G_{k-1}) \cup V(K_\chi^{(k)}))$. Then $v = (v_1, v_2, j)$ for $v_1 \in V(G_{k-1})$, $v_2 \in V(K_\chi^{(k)})$, and $j \in [2r]$. By the inductive hypothesis, there exists $i \in [k-1]$ such that $v_1 \in G_{k-1}^{(i)}$. Then, $(v_1, v_2, j) \in V(G_{k-1}^{(i)}) \times V(K_\chi^{(k)}) \times \{j\}$. As $G_{k-1}^{(i)}$ and $K_\chi^{(k)}$ contain no isolated nodes (inductive hypothesis (ii)), there exist $u_1 \in V(G_{k-1}^{(i)})$ adjacent to v_1 and $u_2 \in V(K_\chi^{(k)})$ adjacent to v_2. The path $w_0 w_1 \dots w_j$ defined by $w_0 = u_1$, $w_k = (v_1, v_2, k)$ for $1 \leq k \leq j$ odd, $w_k = (u_1, u_2, k)$ for $2 \leq k \leq j$ even connects u_1 to v if j is odd. The path $w_0 w_1 \dots w_j$ defined by $w_0 = v_1$, $w_k = (u_1, u_2, k)$ for $1 \leq k \leq j$ odd, $w_k = (v_1, v_2, k)$ for $2 \leq k \leq j$ even connects v_1 to v if j is even. Hence, $v \in \mathcal{N}_j(G_{k-1}^{(i)})$. If $1 \leq j \leq T+2$, then $u \in V(G_k^{(i)})$. If, instead, $T+2 \leq j \leq 2r$, then $u \in V(G_k^{(k)})$.

Now consider any two nodes u, v which are connected in G_k. If $u, v \in V(G_{k-1})$ or $u, v \in V(K_\chi^{(k)})$ we have that $\{u, v\} \in E(\cup_{i\in[k-1]} G_k^{(i)})$ or $\{u, v\} \in E(G_k^{(k)})$, respectively. Suppose $u \in V(G_{k-1})$ but $v \notin V(G_{k-1})$. Then, $v = (v_1, v_2, 1)$ for some $v_1 \in V(G_{k-1})$ and some $v_2 \in V(K_\chi^{(k)})$. As u and v are connected, it means that uv_1 is an edge in G_{k-1}. By the inductive hypothesis on the subgraph covering, there exists $i \in [k-1]$ such that $uv_1 \in G_{k-1}^{(i)}$. Hence, $uv \in E(G_k^{(i)})$. If $u \in V(K_\chi^{(k)})$ but $v \notin V(K_\chi^{(k)})$, $uv \in E(G_k^{(k)})$. Suppose now that $u, v \notin V(G_{k-1}) \cup V(K_\chi^{(k)})$. Then, there exist $u_1, v_1 \in V(G_{k-1})$, $u_2, v_2 \in V(K_\chi^{(k)})$, and $j_u, j_j \in [2r]$ with $|j_u - j_v| \leq 1$ such that $u = (u_1, u_2, j_u)$ and $v = (v_1, v_2, j_v)$. Furthermore, as uv is an edge of G_k, it holds that $u_1 u_2$ is an edge in G_{k-1}, and $v_1 v_2$ is an edge in $K_\chi^{(k)}$. From the inductive hypothesis on the subgraph covering, there exists $i \in [k-1]$ such that $G_{k-1}^{(i)}$ contains $u_1 u_2$. We have that $u, v \in V(G_k[V(G_{k-1}^{(i)}) \times V(K_\chi^{(k)}) \times \{j_u, j_v\}])$. Let $j = \max(j_u, j_v)$ Then, $uv \in E(G_k[\mathcal{N}_j(G_{k-1}^{(i)})])$. Hence, if $j \leq T+2$, $uv \in E(G_k^{(i)})$. If, instead, $T+1 \leq j \leq 2r$, $uv \in E(G_k^{(k)})$.

Property (i). Consider $G_k^{(i)}$ for $i < k$. The function $f_i = \mathrm{pr}_{G_{k-1}} \restriction_{\mathcal{N}_T(G_k^{(i)})}$ is a homomorphism from $G_k[\mathcal{N}_T(G_k^{(i)})]$ to $G_{k-1}[\mathcal{N}_T(G_{k-1}^{(i)})]$. As $G_{k-1}[\mathcal{N}_T(G_{k-1}^{(i)})]$ is χ-colorable by the inductive hypothesis (i), so it is $G_k[\mathcal{N}_T(G_k^{(i)})]$. Since $G_k[\mathcal{N}_T(G_k^{(i)})]$ contains $K_\chi^{(i)}$ as a subgraph, χ colors are also necessary.

Similarly, consider $G_k[\mathcal{N}_T(G_k^{(k)})]$. Then, $f_k = \mathrm{pr}_{K_\chi^{(k)}} \restriction_{[\mathcal{N}_T(G_k^{(k)})]}$ is a homomorphism from $G_k[\mathcal{N}_T(G_k^{(k)})]$ to $K_\chi^{(k)}$. Hence, $G_k[\mathcal{N}_T(G_k^{(k)})]$ is χ-colorable. As $G_k[\mathcal{N}_T(G_k^{(k)})]$ contains $K_\chi^{(k)}$, its chromatic number is χ.

Property (ii). Fix $i \in [k-1]$. Observe that $G_k^{(i)} = G_{k-1}^{(i)} \star_{2r} K_\chi^{(k)}[\mathcal{N}_{T+1}(G_{k-1}^{(i)})]$. As both $G_{k-1}^{(i)}$ (by the inductive hypothesis (ii)) and $K_\chi^{(k)}$ are connected, their join is connected. Then, Lemma 3.2 implies that $G_k^{(i)}$ is connected. The same applies for $G_k^{(k)}$ by observing that $G_k^{(i)} = G_{k-1}^{(i)} \star_{2r} K_\chi^{(k)}[\mathcal{N}_{2r-T}(K_\chi^{(k)})]$.

Property (iii). Consider any node $v \in V(G_{k-1}) \cup V(G_{k-1}) \times V(K_\chi^{(k)}) \times \{1, \dots, T+1\}$. Let $v' = \mathrm{pr}_{G_{k-1}}(v)$. By the inductive hypothesis (iii), the set of nodes $\mathcal{N}_1(v') \cap V(G_{k-1})$ is contained in some $G_{k-1}^{(i)}$ for $i \in [k-1]$. By definition of $G_k^{(i)}$, it follows that $\mathcal{N}_1(v) \subseteq V(G_k^{(i)})$. Now, consider any node $v \in V(K_\chi^{(k)}) \cup V(G_{k-1}) \times V(K_\chi^{(k)}) \times \{T+2, \cdots 2r\}$. By definition of $G_k^{(k)}$, we have that $\mathcal{N}_1(v) \subseteq G_k^{(k)}$.

Property (iv). Fix $1 \leq i \neq j \leq k-1$. By the inductive hypothesis (iv), there exists a node $u \in V(G_{k-1}[\mathcal{N}_T(G_{k-1}^{(i)})])$ from $G_{k-1}^{(i)}$. By definition of $G_k^{(i)}$, we have that u has distance T from $G_k^{(i)}$. Furthermore, observe that any node in $V(G_{k-1}) \times V(K_\chi^{(k)}) \times 1$ has distance T from G_k^k, concluding the proof. □

Remark 3.4. We highlight that the subgraph cover from Lemma 3.3 is not unique: other families of graphs could be used obtaining a shorter and simpler proof at the expense of a higher number of graphs in the family (e.g., a number of subgraphs that is exponential in k). The latter would result in a worse bound on T in the next corollary.

We are now ready to show that the family of χ-chromatic graphs admits a cheating graph for the c-coloring graphs problem.

Corollary 3.1. *Let $\chi \geq 2$, $c \geq \chi$ be integers, and $k = \left\lfloor \frac{c-1}{\chi-1} \right\rfloor$. Consider $\mathcal{F}$ to be the family of all connected χ-chromatic graphs, and $\mathcal{P}$ to be the problem of c-coloring graphs. For every $N \geq 1$ and $n \geq ((6\chi+1)^{k+1}-1)N/6$, there exists a value T with*

$$T = \Theta\left(\frac{1}{\chi^{1+\frac{1}{k}}}\left(\frac{n}{N}\right)^{\frac{1}{k}}\right)$$

such that $\mathcal{F}$ admits an $(n, k+1, N, T)$-cheating graph for $\mathcal{P}$.

Proof. There exists a unique integer $r \geq 3$ such that

$$\frac{(2r\chi+1)^{k+1}-1}{2r} \leq \frac{n}{N} < \frac{(2r\chi+2\chi+1)^{k+1}-1}{2r+2}.$$

We claim that the graph G_{k+1} defined in Definition 3.2 by iterating $2r$-join operations is an $(n, k+1, N, T)$-cheating graph for $(\mathcal{P}, \mathcal{F})$. Clearly, $\mathcal{P}$ is not solvable on G_{k+1}, while property Definition 1.4.(ii).(a) follows by Lemma 3.3.(iii) (since coloring is an LVL problem with checking radius $t = 1$).

We now prove that Definition 1.4.(ii).(b) holds as well. Consider N copies $G_{k+1,1}, \ldots, G_{k+1,N}$ of the graph G_{k+1}. For each $j \in [N]$, Lemma 3.3 gives us a subgraph cover $\{G^{(i)}_{k+1,j}\}_{i\in[k+1]}$ of $G_{k,j}$ with properties Lemma 3.3.(i) and (iv) verified for $T = \left\lfloor \frac{2r}{3} \right\rfloor$. Notice that

$$T = \Theta\left(\frac{1}{\chi^{1+\frac{1}{k}}}\left(\frac{n}{N}\right)^{\frac{1}{k}}\right).$$

For any choice of indices $\mathbf{x}_N = (x_1, \ldots, x_N) \in [k+1]^N$, we now show that there exists a connected graph $H_{\mathbf{x}_N} \in \mathcal{F}$ on n nodes that admits a subgraph $\tilde{H}_{\mathbf{x}_N}$ such that

(1) $H_{\mathbf{x}_N}[\mathcal{N}_T(\tilde{H}_{\mathbf{x}_N})]$ is isomorphic to $\bigsqcup_{j\in[N]} G_{k+1,j}[\mathcal{N}_T(G^{(x_j)}_{k+1,j})]$;

(2) $\mathcal{X}(H_{\mathbf{x}_N}) = \chi$.

The vertex set $V(H_{\mathbf{x}_N})$ is $V(\sqcup_{j\in[N]} G_{k+1,j})$ together with $n - N \cdot \frac{(2r\chi+1)^{k+1}-1}{2r} \leq \frac{n}{N}$ extra nodes. We take $H_{\mathbf{x}_N}$ to be the disjoint union of $G_{k+1,j}[\mathcal{N}_T(G^{(x_j)}_{k+1,j})]$ for all $j \in [N]$ where, for each $j \in [N-1]$ we add an edge between a node $v_j \in G_{k+1,j}[\mathcal{N}_T(G^{(x_j)}_{k+1,j})]$ and a node $v_{j+1} \in G_{k+1,j+1}[\mathcal{N}_T(G^{(x_{j+1})}_{k+1,j+1})]$ such that $\mathrm{dist}(v_j, G^{(x_j)}_{k+1,j}) = T$ and $\mathrm{dist}(v_{j+1}, G^{(x_{j+1})}_{k+1,j+1}) = T$ (such nodes exist because of Lemma 3.3.(iv)). All remaining nodes form a path of which one endpoint is connected to any node in $G_{k+1,N}[\mathcal{N}_T(G^{(x_N)}_{k+1,N})]$ at distance T from $G^{(x_N)}_{k+1,N}$. Property (1) follows by construction. As for property (2), we observe that the chromatic number of $H_{\mathbf{x}_N}$ is still χ as each component $\mathcal{N}_T(G^{(x_j)}_{k+1,j})$ is χ-chromatic by Lemma 3.3.(i), and $\chi \geq 2$. Furthermore, $H_{\mathbf{x}_N}$ is connected by Lemma 3.3.(ii) combined with Lemma 3.2 and observing that connected disjoint connected components through paths. □

The proof of our main lower bound now follows easily.

Proof of Theorem 3.1. Let $k = \alpha$ and

$$N = \left\lceil \frac{\log \frac{1}{\varepsilon}}{\log\left(1+\frac{1}{k}\right)} \right\rceil.$$

By Corollary 3.1, $\mathcal{F}$ admits an $(n, k+1, N, T)$-cheating graph for the c-coloring graphs problem for any $n \geq ((6\chi+1)^{k+1}-1)N/6$ and for some

$$T = \Theta\left(\frac{1}{\chi^{1+\frac{1}{k}}} \cdot \left(\frac{n\log\left(1+\frac{1}{k}\right)}{\log\frac{1}{\varepsilon}}\right)^{\frac{1}{k}}\right).$$

Theorem 2.1 implies that there is a connected graph $H \in \mathcal{F}$ on n nodes such that the probability that any outcome $\mathcal{O}$ with locality T is c-coloring H is at most $(1 - 1/(k+1))^N$. We remind the reader that the graph can be chosen as in Definition 1.4.(ii).(b), and Corollary 3.1 implies that such graph is connected. By definition of k and N, this probability is at most ε. By observing that $\log^{\frac{1}{k}}(1+\frac{1}{k}) = \Theta(1)$, we get the thesis. □

3.2 Bounding quantum advantage in rooted trees

Let us define a model that is slightly *weaker* than the non-signaling LOCAL model.

The bounded-dependence model. The bounded-dependence model produces non-signaling outcomes exactly like the non-signaling LOCAL model. However, for any produced outcome $\mathcal{O}$ that is non signaling beyond distance T, we further require the following property: For any input (G, x) and any two subsets $S_1, S_2 \subseteq V(G)$ such that $\text{dist}_G(S_1, S_2) > 2T$, the output distribution given by $\mathcal{O}$ restricted to S_1 is independent of that restricted to S_2.

The bounded-dependence model captures all physical variants of the LOCAL model that do not make use of share resources. In this section we prove that LCL problems over a special class of graphs that can be solved with locality $O(1)$ in the bounded-dependence model are then also solvable in the LOCAL model with locality $O(\log^\star n)$, limiting possible quantum advantage. We focus on *rooted trees*. A tree $G = (V, E)$ is a rooted tree if the following properties hold: There is a special node $v \in V$ that is called the *root* of the tree, and all leaf-to-root paths are oriented towards the root. The theorem we are going to prove follows and come from Akbari et al. [Akb+24].

Theorem 3.3. *Let Π be an LCL problem over rooted trees that has complexity $O(1)$ in the bounded-dependence model. Then Π has complexity $O(\log^\star n)$ in the classical LOCAL model.*

Notice that it has been shown also the reverse, but we are not going to prove it here.

Theorem 3.4. *Let Π be an LCL problem over rooted trees that has complexity $O(\log^\star n)$ has complexity $O(\log^\star n)$ in the classical LOCAL model. Then Π has complexity $O(1)$ in the bounded-dependence model. Furthermore, the output distribution solving Π has success probability 1.*

Notice that LCL complexity classes in the region $O(\log^\star n)$ coincide between the deterministic and randomized LOCAL model, i.e. randomness does not help for these problems Naor and Stockmeyer [NS95]. In order to prove Theorem 3.3, we consider a sequential version of the LOCAL model: the sequential LOCAL (SLOCAL) model.

The SLOCAL model. The SLOCAL model is a sequential formulation of the LOCAL model. For each input graph $G = (V, E)$ of n nodes, an adversary picks an ordering $v_1 v_2 \dots v_n$ of the nodes in V. Let $T = T(n)$ be a parameter depending on the size of the input graph. An algorithm $\mathcal{A}$ in the SLOCAL model processes nodes according to this order. When v_i is processed, $\mathcal{A}$ gets to gather the radius-T local view of the node v_i and must decide the final output of v_i. Notice that $\mathcal{A}$ might store in the output of v_i the whole memory of v_i. Hence, when processing a node v_j with $j > i$, if v_i belongs to the radius-T view of v_j, $\mathcal{A}$ has access to the whole local view of v_i as well. We call $T = T(n)$ the running time (or locality) of $\mathcal{A}$.

A noticeable result states that the complexity classes $O(\log^\star n)$ in LOCAL and $O(1)$ in SLOCAL coincide for LCL problems [CKP19; Akb+23].

Lemma 3.4. *Let Π be an LCL problem. Π has complexity $O(1)$ in SLOCAL if and only if it has complexity $O(\log^\star n)$ in the classical LOCAL model (both deterministic and randomized).*

We first show how the SLOCAL model can nicely "partition" a rooted forest. A rooted forest is just a collection of rooted trees. The idea is not new and comes from [CSS23, Section 7] where it is directly applied to the LOCAL model; we present here an adaptation to SLOCAL.

For a rooted forest $G = (V, E)$ and each node $v \in V$, we denote by G_v the subtree of G that is rooted at v. A root-to-leaf path $v_0 v_1 \dots v_k$, where $v_i \in V$ for all $i \in \{0, 1, \dots, k\}$, is a path of a rooted tree such that (v_i, v_{i-1}) is an edge for all $i \in [k]$: we say that the path *starts* at v_0 and *ends* at v_k.

Definition 3.3 ((α, β)-clustering of rooted trees)**.** Let $G = (V, E)$ be a rooted forest. An (α, β)-clustering of G is a subset $\mathcal{L} \subseteq V$ of *leader nodes* that contains the root and is such that, for each $v \in \mathcal{L}$, the following properties are met:

1. G_v does not contain elements of $\mathcal{L}$ at levels $1, \dots, \alpha - 1$.

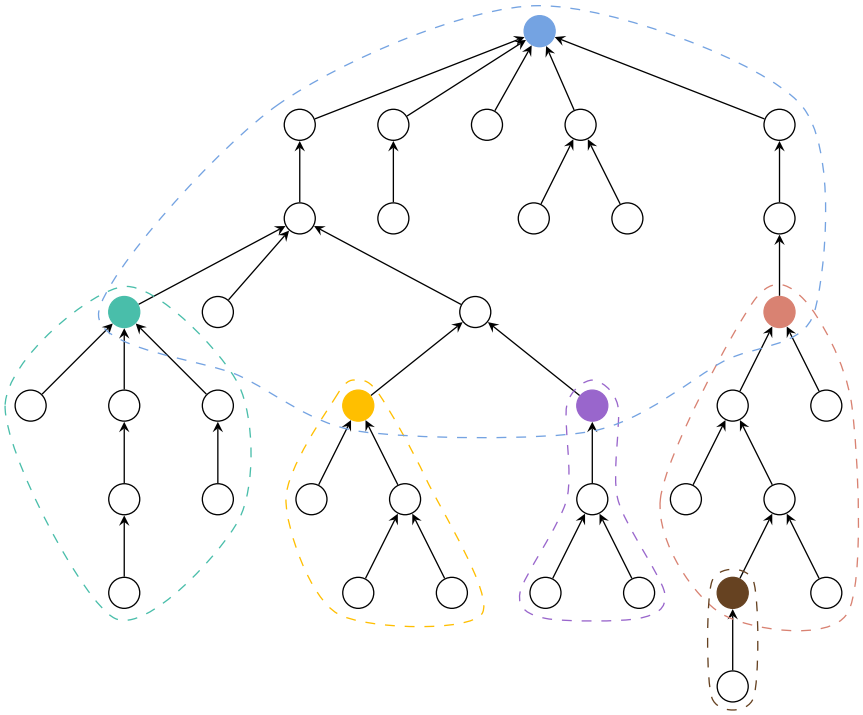


Figure 5: A $(3,4)$-clustering of a rooted tree. The leader nodes are colored and their clusters marked with their respective color.

2. Each maximal oriented path in G_v contains exactly one element $u \in \mathcal{L}$ such that $\text{dist}_{G_v}(u,v) \in [\alpha,\beta]$, unless the length of the maximal oriented path is at most $\beta-1$, in which case there is at most one element of $\mathcal{L}$ in the path (and possibly none).

A *cluster* is a maximal connected component of non-leader nodes. See Fig. 5 for an example.

We will combine many SLOCAL algorithms: it is easy to prove that the combination of two SLOCAL algorithms with localities T_1, T_2 gives an SLOCAL algorithm with locality $O(T_1+T_2)$ [GKM17, Lemma 2.3]. Notice that the same holds for LOCAL algorithms.

Lemma 3.5. *Let $\alpha \in \mathbb{N}_+$, and let $G=(V,E)$ be a rooted forest. There is an SLOCAL algorithm with locality $O(\alpha)$ that produces an $(\alpha-1, 2\alpha+1)$-clustering of G in time at most 4α.*

Proof. We combine some SLOCAL algorithms. Consider first the following algorithm $\mathcal{A}_1$: Suppose a node $v \in V$ is picked by the adversary and asked to commit to something. For each root-to-leaf path $v_0 v_1 \dots v_{2\alpha-1}$ (with $v_0 = v$) that v can distinguish using locality $2\alpha-1$, in parallel, v precommits that the node $v_{\alpha-1}$ is the *leader* of the path $v_{\alpha-1} v_\alpha \dots v_{2\alpha-1}$ unless there is another precommitment on the path within distance α from v.

Next we define an algorithm $\mathcal{A}_2$ that takes as input a rooted forest labelled by $\mathcal{A}_1$: Each node v looks at its radius-α neighborhood and checks if there is some other (unique) node that precommitted for v in the neighborhood. If so, it becomes a leader and stores the path that is under its leadership, otherwise it does nothing.

The final algorithm $\mathcal{A}_3$ takes as input a rooted forest labelled by $\mathcal{A}_2$ and is defined as follows: Each node v becomes a leader if and only if it belongs to an oriented root-to-leaf path $v_0 v_1 \dots v_{\alpha-1}$, with $v_{\alpha-1} = v$ lead by v_0. All other nodes do not output anything, except the root, that becomes a leader.

We now prove that the SLOCAL algorithm $\mathcal{A}$ providing the clustering is the composition of $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$, and has locality $O(\alpha)$.

Notice that the root is a leader thanks to $\mathcal{A}_3$. Property 1 in Definition 3.3 is satisfied due to $\mathcal{A}_3$ as well. Furthermore, for each maximal root-to-leaf path, consecutive leader nodes must be within distance at most

$2\alpha + 1$ between each other, unless the path ends with a leaf at distance at most 2α from the last leader. In fact, if the distance between consecutive leaders is at least $2\alpha + 2$, or there is only one leader and the path is longer than 2α from the last leader, there would be at least one non-leader node that does not see any leader within distance α in the path, which is impossible due to how precommitments are done in $\mathcal{A}_1$. □

We are now ready to prove Theorem 3.3. Notice that, by Theorem 3.4, we will assume that any outcome solving an LCL problem Π in constant locality has success probability 1.

Proof of Theorem 3.3. Let G denote the input graph, which is a rooted forest, and assume we are given an outcome $\mathcal{O}$ with locality $T = O(1)$ in the bounded-dependence model that solves the problem with probability 1. We now show how to get an randomized LOCAL algorithm $\mathcal{B}$ solving the problem with roughly the same locality. $\mathcal{B}$ is the composition of four different LOCAL algorithms $\mathcal{B}_1, \mathcal{B}_3, \mathcal{B}_3, \mathcal{B}_4$, each of them having locality $O(\log^\star n)$:

1. We first use Lemma 3.5 with $\alpha = 10T+4r$, where r is the checking radius of Π, and obtain an algorithm $\mathcal{B}_1$ that outputs an $(\alpha - 1, 2\alpha + 1)$-clustering of G in time $O(\log^\star n)$ in randomized LOCAL (since it has complexity $O(4\alpha) = O(1)$ in SLOCAL).

2. Now consider an algorithm $\mathcal{B}_2$ with locality $2\alpha + 4r$ that takes as input G as labelled by $\mathcal{B}_1$ and works as follows: When processing a leader node v, it collects the topology of the radius-$2r$ neighborhood of the set composed of the closest leaders v sees in each root-to-leaf path. Then, v samples from $\mathcal{O}$ in this neighborhood and precommits a solution to the LCL for the nodes in such neighborhood. If v is the root, it also presents its own radius-$2r$ neighborhood to $\mathcal{A}'$ and precommits a solution for the whole neighborhood. Observe that all such neighborhoods are disjoint by construction of the $(\alpha - 1, 2\alpha + 1)$-clustering.

3. $\mathcal{B}_3$ also has locality $2\alpha+4r$ and just makes all nodes whose label has been precommitted by some other node actually output the precommitted label. (The output of $\mathcal{B}_2$ guarantees there are no conflicts to be resolved, since $\mathcal{O}$ solves the problem with probability 1.)

4. Finally, $\mathcal{B}_4$ has again locality $2\alpha + 4r$ and just brute-forces a solution in each cluster. The solution is guaranteed to exist because we sampled from $\mathcal{O}$ in each disjoint neighborhood and solves the problem with probability 1. In fact, $\mathcal{B}_4$ can just continue sampling on each cluster it sees (restricting to the outputs of previous algorithms).

The combination of $\mathcal{B}_1$, $\mathcal{B}_2$, $\mathcal{B}_3$, and $\mathcal{B}_4$ yields a randomized LOCAL algorithm $\mathcal{B}$ that has locality less than $O(\log^\star n)$. □

References

[AF14] Heger Arfaoui and Pierre Fraigniaud. "What can be computed without communications?" In: *SIGACT News* 45.3 (2014), pp. 82–104. DOI: 10.1145/2670418.2670440.

[Akb+23] Amirreza Akbari et al. "Locality in Online, Dynamic, Sequential, and Distributed Graph Algorithms". In: *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*. Ed. by Kousha Etessami, Uriel Feige, and Gabriele Puppis. Vol. 261. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 10:1–10:20. DOI: 10.4230/LIPICS.ICALP.2023.10.

[Akb+24] Amirreza Akbari et al. "Online Locality Meets Distributed Quantum Computing". In: *CoRR* abs/2403.01903 (2024). DOI: 10.48550/ARXIV.2403.01903. arXiv: 2403.01903. URL: https://doi.org/10.48550/arXiv.2403.01903.

[AV22] Joran van Apeldoorn and Tijn de Vos. "A Framework for Distributed Quantum Queries in the CONGEST Model". In: *PODC '22: ACM Symposium on Principles of Distributed Computing, Salerno, Italy, July 25 - 29, 2022*. Ed. by Alessia Milani and Philipp Woelfel. ACM, 2022, pp. 109–119. DOI: 10.1145/3519270.3538413.

[Bog13] Ilya I. Bogdanov. "Examples of topologically highly chromatic graphs with locally small chromatic number". In: *CoRR* abs/1311.2844 (2013). DOI: 10.48550/arXiv.1311.2844. arXiv: 1311.2844. URL: https://arxiv.org/abs/1311.2844.

[CH+22] Keren Censor-Hillel et al. "Quantum Distributed Algorithms for Detection of Cliques". In: *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January 31 - February 3, 2022, Berkeley, CA, USA*. Ed. by Mark Braverman. Vol. 215. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 35:1–35:25. DOI: 10.4230/LIPICS.ITCS.2022.35.

[CKP19] Yi-Jun Chang, Tsvi Kopelowitz, and Seth Pettie. "An Exponential Separation between Randomized and Deterministic Complexity in the LOCAL Model". In: *SIAM Journal on Computing* 48.1 (2019), pp. 122–143. DOI: 10.1137/17M1117537. eprint: https://doi.org/10.1137/17M1117537. URL: https://doi.org/10.1137/17M1117537.

[Coi+24] Xavier Coiteux-Roy et al. "No Distributed Quantum Advantage for Approximate Graph Coloring". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*. Ed. by Bojan Mohar, Igor Shinkar, and Ryan O'Donnell. ACM, 2024, pp. 1901–1910. DOI: 10.1145/3618260.3649679. URL: https://doi.org/10.1145/3618260.3649679.

[CSS23] Yi-Jun Chang, Jan Studený, and Jukka Suomela. "Distributed graph problems through an automata-theoretic lens". In: *Theoretical Computer Science* 951 (2023), p. 113710. DOI: 10.1016/J.TCS.2023.113710.

[Elk+14] Michael Elkin et al. "Can quantum communication speed up distributed computation?" In: *ACM Symposium on Principles of Distributed Computing, PODC '14, Paris, France, July 15-18, 2014*. Ed. by Magnús M. Halldórsson and Shlomi Dolev. ACM, 2014, pp. 166–175. DOI: 10.1145/2611462.2611488.

[GJS04] András Gyárfás, Tommy Jensen, and Michael Stiebitz. "On graphs with strongly independent color-classes". In: *Journal of Graph Theory* 46.1 (2004), pp. 1–14. DOI: 10.1002/JGT.10165.

[GKM09] Cyril Gavoille, Adrian Kosowski, and Marcin Markiewicz. "What Can Be Observed Locally?" In: *Distributed Computing, 23rd International Symposium, DISC 2009, Elche, Spain, September 23-25, 2009. Proceedings*. Ed. by Idit Keidar. Vol. 5805. Lecture Notes in Computer Science. Springer, 2009, pp. 243–257. DOI: 10.1007/978-3-642-04355-0_26.

[GKM17] Mohsen Ghaffari, Fabian Kuhn, and Yannic Maus. "On the complexity of local distributed graph problems". In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*. Ed. by Hamed Hatami, Pierre McKenzie, and Valerie King. ACM, 2017, pp. 784–797. DOI: 10.1145/3055399.3055471.

[IL19] Taisuke Izumi and François Le Gall. "Quantum Distributed Algorithm for the All-Pairs Shortest Path Problem in the CONGEST-CLIQUE Model". In: *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing, PODC 2019, Toronto, ON, Canada, July 29 - August 2, 2019*. Ed. by Peter Robinson and Faith Ellen. ACM, 2019, pp. 84–93. DOI: 10.1145/3293611.3331628.

[ILM20] Taisuke Izumi, François Le Gall, and Frédéric Magniez. "Quantum Distributed Algorithm for Triangle Finding in the CONGEST Model". In: *37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020, March 10-13, 2020, Montpellier, France*. Ed. by Christophe Paul and Markus Bläser. Vol. 154. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 23:1–23:13. DOI: 10.4230/LIPICS.STACS.2020.23.

[Lin87] Nathan Linial. "Distributive Graph Algorithms-Global Solutions from Local Data". In: *28th Annual Symposium on Foundations of Computer Science, Los Angeles, California, USA, 27-29 October 1987*. IEEE Computer Society, 1987, pp. 331–335. DOI: 10.1109/SFCS.1987.20.

[LM18] 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*. ACM, 2018, pp. 337–346. DOI: 10.1145/3212734.3212744.

[LNR19] François Le Gall, Harumichi Nishimura, and Ansis Rosmanis. "Quantum Advantage for the LOCAL Model in Distributed Computing". In: *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019, March 13-16, 2019, Berlin, Germany*. Ed. by Rolf Niedermeier and Christophe Paul. Vol. 126. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, 49:1–49:14. DOI: 10.4230/LIPICS.STACS.2019.49.

[Mat08] Jiří Matoušek. *Using the Borsuk–Ulam Theorem*. Springer Berlin Heidelberg, 2008. DOI: 10.1007/978-3-540-76649-0.

[MN22] Frédéric Magniez and Ashwin Nayak. "Quantum Distributed Complexity of Set Disjointness on a Line". In: *ACM Transactions on Computation Theory* 14.1 (2022), 5:1–5:22. DOI: 10.1145/3512751.

[MST13] Bojan Mohar, Gábor Simonyi, and Gábor Tardos. "Local chromatic number of quadrangulations of surfaces". In: *Combinatorica* 33.4 (2013), pp. 467–495. DOI: 10.1007/S00493-013-2771-Y.

[NS95] Moni Naor and Larry J. Stockmeyer. "What Can be Computed Locally?" In: *SIAM Journal on Computing* 24.6 (1995), pp. 1259–1277. DOI: 10.1137/S0097539793254571.

[Sti85] Michael Stiebitz. "Beiträge zur Theorie der färbungskritischen Graphen". Habilitation. TH Ilmenau, 1985.

[WWY22] ChengSheng Wang, Xudong Wu, and Penghui Yao. "Complexity of Eccentricities and All-Pairs Shortest Paths in the Quantum CONGEST Model". In: *CoRR* abs/2206.02766 (2022). DOI: 10.48550/ARXIV.2206.02766. arXiv: 2206.02766.

[WY22] Xudong Wu and Penghui Yao. "Quantum Complexity of Weighted Diameter and Radius in CONGEST Networks". In: *PODC '22: ACM Symposium on Principles of Distributed Computing, Salerno, Italy, July 25 - 29, 2022*. Ed. by Alessia Milani and Philipp Woelfel. ACM, 2022, pp. 120–130. DOI: 10.1145/3519270.3538441.