Computing Power of Hybrid Models in Synchronous Networks

Pierre Fraigniaud
IRIF, Université Paris Cité and CNRS, France

Pedro Montealegre
Faculty of Engineering and Science, Adolfo Ibáñez University, Santiago, Chile

Pablo Paredes
Department of Mathematical Engineering, University of Chile, Santiago, Chile

Ivan Rapaport
DIM-CMM (UMI 2807 CNRS), University of Chile, Santiago, Chile

Martín Ríos-Wilson
Faculty of Engineering and Science, Adolfo Ibáñez University, Santiago, Chile

Ioan Todinca
LIFO, Université d'Orléans and INSA Centre-Val de Loire, France

Abstract

During the last two decades, a small set of distributed computing models for networks have emerged, among which LOCAL, CONGEST, and Broadcast Congested Clique (BCC) play a prominent role. We consider *hybrid* models resulting from combining these three models. That is, we analyze the computing power of models allowing to, say, perform a constant number of rounds of CONGEST, then a constant number of rounds of LOCAL, then a constant number of rounds of BCC, possibly repeating this figure a constant number of times. We specifically focus on 2-round models, and we establish the complete picture of the relative powers of these models. That is, for every pair of such models, we determine whether one is (strictly) stronger than the other, or whether the two models are incomparable. The separation results are obtained by approaching communication complexity through an original angle, which may be of an independent interest. The two players are not bounded to compute the value of a binary function, but the *combined* outputs of the two players are constrained by this value. In particular, we introduce the `XOR-Index` problem, in which Alice is given a binary vector $x \in \{0,1\}^n$ together with an index $i \in [n]$, Bob is given a binary vector $y \in \{0,1\}^n$ together with an index $j \in [n]$, and, after a single round of 2-way communication, Alice must output a boolean out_A, and Bob must output a boolean out_B, such that $\text{out}_A \wedge \text{out}_B = x_j \oplus y_i$. We show that the communication complexity of `XOR-Index` is $\Omega(n)$ bits.

2012 ACM Subject Classification Theory of computation → Distributed algorithms

Keywords and phrases hybrid model, synchronous networks, LOCAL, CONGEST, Broadcast Congested Clique

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2022.20

Related Version *Full Version*: `https://arxiv.org/abs/2208.02640`

Funding *Pierre Fraigniaud*: Additional support for ANR projects QuData and DUCAT.
Pedro Montealegre: This work was supported by Centro de Modelamiento Matemático (CMM), FB210005, BASAL funds for centers of excellence from ANID-Chile, and ANID-PAI77170068.
Ivan Rapaport: This work was supported by Centro de Modelamiento Matemático (CMM), FB210005, BASAL funds for centers of excellence from ANID-Chile, and ANID-FONDECYT 1220142.
Martín Ríos-Wilson: Additional support from ANID-FONDECYT Postdoctorado 3220205.

1 Introduction

This paper analyzes the relative power of distributed computing models for networks, all resulting from the combination of standard synchronous models such as LOCAL and CONGEST [47], as well as Broadcast Congested Clique (BCC) [21]. Each of these three models has its strengths and limitations. In particular, CONGEST assumes the ability for each node to send a specific message to each of its neighbors at every round (even in a clique). However, the communication links have limited bandwidth. Specifically, at most $O(\log n)$ bits can be sent through any link during a round, in n-node networks. LOCAL assumes a link with unlimited bandwidth between any two neighboring nodes, but the information acquired by any node u after $t \geq 0$ rounds of communication is limited to the data available at nodes at distance at most t from u in the network. Finally, BCC supports all-to-all communications between the nodes, and thus does not suffer from the locality constraint of LOCAL and CONGEST. However, at each round, each node is bounded to send a *same* $O(\log n)$-bit message to all the other nodes. In this paper, we investigate the power of models resulting from combining these three models, in order to take advantage of their positive aspects without suffering from their negative ones.

For the sake of comparing models, we focus on the standard framework of distributed *decision* problems on labeled graphs (see [27]). Such problems are defined by a collection $\mathcal{L}$ of pairs (G, ℓ), where $G = (V, E)$ is a graph, and $\ell : V \to \{0,1\}^*$ is a function assigning a label $\ell(u) \in \{0,1\}^*$ to every $u \in V$. Such a set $\mathcal{L}$ is called a distributed *language*. For instance, deciding whether a certain set U of nodes in a graph G forms a vertex cover can be modeled by the language

$$\texttt{vertex-cover} = \big\{(G, \ell) : \forall \{u, v\} \in E(G),\ \ell(u) = 1 \lor \ell(v) = 1\big\},$$

by labeling 1 all the vertices in U, and 0 all the other vertices. Similarly, deciding C_4-freeness can be modeled by the language $C_4\texttt{-freeness} = \{(G, \ell) : C_4 \not\preceq G\}$, where $H \preceq G$ denotes that H is a subgraph of G, and deciding whether a graph is planar can be captured by the language $\texttt{planarity} = \{(G, \ell) : G \text{ is planar}\}$. A distributed algorithm A *decides* $\mathcal{L}$ if every node running A eventually accepts or rejects, and the following condition is satisfied: for every labeled graph (G, ℓ),

$$(G, \ell) \in \mathcal{L} \iff \text{all nodes accept.}$$

That is, every node should accept in a yes-instance (i.e., an instance $(G, \ell) \in \mathcal{L}$), and, in a no-instance (i.e., an instance $(G, \ell) \notin \mathcal{L}$), at least one node must reject.

For every $t \geq 0$, let us denote by $\mathbf{L}^t$ the set of distributed languages $\mathcal{L}$ for which there is a t-round algorithm in the LOCAL model deciding $\mathcal{L}$, with $\mathbf{L} = \mathbf{L}^1$. The sets $\mathbf{C}^t$ and $\mathbf{B}^t$ are defined similarly, for the CONGEST and BCC models, respectively. Note that while it is easy to show, using indistinguishability arguments, that, for every $t \geq 1$, $\mathbf{L}^t \smallsetminus \mathbf{L}^{t-1} \neq \varnothing$ and $\mathbf{C}^t \smallsetminus \mathbf{C}^{t-1} \neq \varnothing$, establishing that there is indeed a decision problem in $\mathbf{B}^t \smallsetminus \mathbf{B}^{t-1}$ requires significantly more work [46]. Also, we define $\mathbf{L}^* = \cup_{t\geq 0}\mathbf{L}^t$, $\mathbf{C}^* = \cup_{t\geq 0}\mathbf{C}^t$, and $\mathbf{B}^* = \cup_{t\geq 0}\mathbf{B}^t$. So, in particular, $\mathbf{L}^*$ is the class of distributed languages that can be decided in a constant number of rounds in the LOCAL model.

The three models under consideration, i.e., LOCAL, CONGEST, and BCC exhibit very different behaviors with respect to decision problems. For instance, it is known [22] that

$$C_4\texttt{-freeness} \in \mathbf{L} \smallsetminus (\mathbf{B}^* \cup \mathbf{C}^*),$$

whenever one assumes, as we do in this paper, that, for all models under consideration, every node is initially aware of the identifiers[1] of its neighbors. On the other hand, it is also known [12] that

$$\texttt{planarity} \in \mathbf{B} \setminus \mathbf{L}^*.$$

This means that while no LOCAL algorithms can decide planarity in a constant number of rounds, there is a 1-round BCC algorithm deciding planarity, and while no BCC algorithms can decide C_4-freeness in a constant number of rounds, there is a 1-round LOCAL algorithm deciding C_4-freeness. So, if one allows LOCAL algorithms to do just a single round of all-to-all communication, as in BCC, then both C_4-freeness and planarity can be solved in a constant number of rounds, hence increasing the computational power of LOCAL dramatically.

This observation led us to investigate scenarios such as the case in which the CONGEST model is enhanced by allowing nodes to perform few rounds in either LOCAL, or BCC. What would be the computing power of such a *hybrid* model? For answering this question, for a collection of non-negative integers $\alpha_1, \dots, \alpha_k$, $\beta_1, \dots, \beta_k$, and $\gamma_1, \dots, \gamma_k$, we define the set

$$\prod_{i=1}^{k} \mathbf{L}^{\alpha_i} \mathbf{B}^{\beta_i} \mathbf{C}^{\gamma_i}$$

as the class of decision languages $\mathcal{L}$ which can be decided by an algorithm performing $\alpha_1 \geq 0$ rounds of LOCAL, followed by $\beta_1 \geq 0$ rounds of BCC, followed by $\gamma_1 \geq 0$ rounds of CONGEST, followed by $\alpha_2 \geq 0$ rounds of LOCAL, etc., up to $\gamma_k \geq 0$ rounds of CONGEST. For instance, we have

$$\{\texttt{planarity}, C_4\texttt{-freeness}\} \subseteq \mathbf{LB} \cap \mathbf{BL}.$$

However, how do **LB** and **BL** compare? And what about **CB** vs. **BC**, and **LC** vs. **CL**? These are the kinds of questions that we are studying in this paper. In the long-term perspective, this line of research is motivated by the following question. Let $\mathcal{L}$ be a fixed distributed language, and let us assume that a round of LOCAL costs a (say, for acquiring high-throughput channels), that a round of BCC costs b (say, for benefiting of facilities supporting all-to-all communications), and that a round of CONGEST costs c. The goal is to minimize the total cost of an algorithm deciding $\mathcal{L}$ in a constant number of rounds, that is, to solve the following minimization problem:

$$\min_{\prod_{i=1}^{k} \mathbf{L}^{\alpha_i} \mathbf{B}^{\beta_i} \mathbf{C}^{\gamma_i} \ni \mathcal{L}} \left(a \sum_{i=1}^{k} \alpha_i + b \sum_{i=1}^{k} \beta_i + c \sum_{i=1}^{k} \gamma_i \right). \tag{1}$$

Note that, for $a = b = c = 1$, Eq. (1) corresponds to minimizing the number of rounds for deciding $\mathcal{L}$ when using a combination of the communication facilities provided by LOCAL, CONGEST, and BCC. For instance, deciding whether a graph is C_k-free can be achieved in $\lfloor \frac{k}{2} \rfloor$ rounds in LOCAL, that is, $C_k\texttt{-freeness} \in \mathbf{L}^{\lfloor k/2 \rfloor}$. Eq. (1) is asking whether deciding C_k-**freeness** could be achieved at a lower cost by combining LOCAL, CONGEST, and BCC. For tackling Eq. (1), we need a better understanding of the fundamental effects resulting from combining these models.

[1] In each of the models, every node u of a n-node network $G = (V, E)$ is supposed to be provided with an identifier $\mathsf{id}(u)$, where $\mathsf{id} : V \to [1, N]$ is one-to-one, and $N(n) = \mathrm{poly}(n)$, i.e., all identifiers can be stored on $O(\log n)$ bits in n-node networks. We also assume that all nodes are initially aware of the size n of the network, merely because this is the case in model BCC.

1.1 Our Results

On the negative side, we provide a series of separation results between 2-round hybrid models. In particular, we show that **BC** and **CB** are incomparable. That is, there are languages in $\mathbf{BC} \setminus \mathbf{CB}$, and languages in $\mathbf{CB} \setminus \mathbf{BC}$. In fact, we show stronger separation results, by establishing that $\mathbf{BC} \setminus \mathbf{C}^*\mathbf{B} \neq \varnothing$, and $\mathbf{CB} \setminus \mathbf{BL}^* \neq \varnothing$. That is, in particular, there are languages that can be decided by a 2-round algorithm performing a single BCC round followed by one CONGEST round, which cannot be decided by any algorithm performing k CONGEST rounds followed by a single BCC round, for any $k \geq 1$.

On the positive side, we show that, for any non-negative integers $\alpha_1, \dots, \alpha_k, \beta_1, \dots, \beta_k$,

$$\prod_{i=1}^{k} \mathbf{L}^{\alpha_i}\mathbf{B}^{\beta_i} \subseteq \mathbf{L}^{\sum_{i=1}^{k}\alpha_i}\mathbf{B}^{\sum_{i=1}^{k}\beta_i}. \tag{2}$$

That is, if a language $\mathcal{L}$ can be decided by a t-round algorithm alternating LOCAL and BCC rounds, then $\mathcal{L}$ can be decided by a t-round algorithm performing all its LOCAL rounds first, and then all its BCC rounds – with the notations of Eq. (2), $t = \sum_{i=1}^{k}(\alpha_i + \beta_i)$. So, in particular $\mathbf{BL} \subseteq \mathbf{LB}$. This inclusion is strict, since, as said before, $\mathbf{CB} \setminus \mathbf{BL}^* \neq \varnothing$. In fact, this separation holds even if the number of LOCAL rounds depends on the number of nodes n in the network, as long as the algorithm performs $o(n)$ LOCAL rounds after its BCC round. Another consequence of Eq. (2) is that the largest class of languages among all the ones considered in this paper is $\mathbf{L}^*\mathbf{B}^*$, that is, languages that can be decided by algorithms performing k LOCAL rounds followed by k' BCC rounds, for some $k \geq 0$ and $k' \geq 0$. Thus, Eq. (1) should be studied for languages $\mathcal{L} \in \mathbf{L}^*\mathbf{B}^*$.

Interestingly, our separation results hold even for randomized protocols, which can err with probability at most $\epsilon \leq 1/5$. That is, in particular, there is a language $\mathcal{L} \in \mathbf{CB}$ (i.e., that can be decided by a deterministic 2-round algorithm) which cannot be decided with error probability at most $1/5$ by any randomized algorithm performing one BCC round first, followed by k LOCAL rounds, for any $k \geq 1$. All our results about 2-rounds hybrid models are summarized on Figure 1.

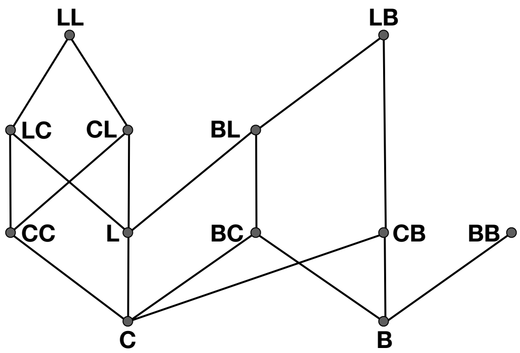

Figure 1 *The poset of 2-round hybrid models. An edge between a set of languages* $\mathbf{S}_1$ *and a set* $\mathbf{S}_2$*, where* $\mathbf{S}_1$ *is at a level lower than* $\mathbf{S}_2$*, indicates that* $\mathbf{S}_1 \subseteq \mathbf{S}_2$*. In fact, all inclusions are strict. Transitive edges are not displayed. Two sets that are not connected by a monotone path are incomparable. For instance,* **CB** *and* **BL** *are incomparable, while* $\mathbf{BC} \subseteq \mathbf{LB}$*.*

Our Techniques. All our separation results are obtained by reductions from communication complexity lower bounds. However, we had to revisit several known communication complexity results for adapting them to the setting of distributed decision, in which no-instances may be rejected by a single node, and not necessarily by all the nodes. In particular, we revisit the classical `Index` problem. Recall that, in this problem, Alice is given a binary vector $x \in \{0,1\}^n$, Bob is given an index $i \in [n]$, and Bob must output x_i based on a single message received from Alice (1-way communication). We define the `XOR-Index` problem, in which Alice is given a binary vector $x \in \{0,1\}^n$ together with an index $i \in [n]$, Bob is given a binary vector $y \in \{0,1\}^n$ together with an index $j \in [n]$, and, after a single round of 2-way communication, Alice must output a boolean out_A and Bob must output a boolean out_B, such that

$$\mathrm{out}_A \wedge \mathrm{out}_B = x_j \oplus y_i.$$

That is, if $x_j \neq y_i$ then Alice and Bob must both accept (i.e., output *true*), and if $x_j = y_i$ then *at least* one of these two players must reject (i.e., output *false*). We show that the sum of the sizes of the message sent by Alice to Bob and the message sent by Bob to Alice is $\Omega(n)$ bits. This bound holds even if the communication protocol is randomized and may err with probability at most $1/5$, and even if the two players have access to shared random coins.

The fact that only one of the two players may reject a no-instance (i.e., an instance where $x_j \oplus y_i = 0$), and not necessarily both, while a yes-instance must be accepted by both players, yields an asymmetry which complicates the analysis. We use information theoretic tools for establishing our lower bound. Specifically, we identify a way to decorrelate the behaviors of Alice and Bob, so that to analyze separately the distribution of decisions taken by each player, and then to recombine them for lower bounding the probability of error in case the messages exchanged between the players are small, contradicting the fact that this error probability is supposed to be small. Roughly, given messages m_A and m_B exchanged by the two players, and given two indices i and j, we compute the value y_i maximizing the error probability for Alice, and the value x_j maximizing the error probability for Bob, conditioned to m_A, m_B, i, j. We then show that the combined pair (x_j, y_i) provide a sufficiently good lower bound on the probability of error for the whole protocol, which contradicts the fact that the error must be at most ϵ.

1.2 Related Work

The LOCAL model was introduced in [42] at the beginning of the 1990s, when the celebrated $\Omega(\log^* n)$ lower bound on the number of rounds for computing a 3-coloring or a maximal independent set (MIS) in the n-node cycle was proved. A few years later, the class of *locally checkable labeling* (LCL) problems was introduced and studied in [45]. This class essentially corresponds to the class $\mathbf{L}^*$, but restricted to graphs with constant maximum degrees. Given $\mathcal{L} \in \mathbf{L}^*$, and the family $\mathcal{G}_\Delta$ of graphs with maximum degree at most Δ, solving the LCL problem induced by $\mathcal{L}$ and $\mathcal{G}_\Delta$ consists of designing a distributed algorithm which, given a graph $G \in \mathcal{G}_\Delta$, computes a labeling ℓ of the nodes such that $(G, \ell) \in \mathcal{L}$. It is known that many LCL problems can be solved in constant number of rounds in LOCAL. This is for instance the case of certain types of weak colorings problems [45]. Also, there is an $O(\sqrt{k}\Delta^{1/\sqrt{k}} \log \Delta)$-approximation algorithm for minimum dominating set running in $O(k)$ rounds [38] (where $k \geq 1$ is a parameter), and there is an $O(n^\varepsilon)$-approximation algorithm for the minimum coloring problem running in $\exp(O(1/\varepsilon))$ rounds [11]. In fact, it is undecidable, in general, whether a given LCL problem has a (construction) algorithm running in a constant number of rounds [45]. A plethora of papers have addressed graph problems in the LOCAL model, and we refer to the survey [52], but several significant results have been obtained since then, among which it is worth mentioning two fields in close connection to the topic of this paper, which emerged in the early 2010s. One is the systematic study of distributed decision problems in various settings, including non-determinism [28, 31, 37] and interactive protocols [36, 44]. The other is a systematic study of the round-complexity of LCL problems (see, e.g., [9, 53], and the references therein).

The CONGEST model is a weaker variant of the LOCAL model in which the size of the messages exchanged at each round between neighbors is bounded to $O(\log n)$ bits, or B bits in the parametrized version of the model. This bound on the message size creates bottlenecks limiting the power of algorithms under this model. A fruitful line of research has established several non-trivial lower bounds on the round-complexity of CONGEST algorithms, by reduction from communication complexity problems (see for instance [1, 5, 24, 48, 50]). Nevertheless, several problems can still be solved in a constant number of rounds in CONGEST. This is for instance the case of computing a $(2+\varepsilon)$-approximation of minimum vertex cover which can be done in $O(\log\Delta/\log\log\Delta)$ rounds [10] in graphs with maximum degree Δ. Also, *testing* (a weaker variant of decision, a la property-testing) the presence of specific subgraphs like small cliques or short cycles can be done in a constant number of rounds in CONGEST(see, e.g., [14, 25, 29, 30, 41]).

The congested clique model [21, 43] has first been introduced in its *unicast* version (UCC), where every node is allowed to send potentially different $O(\log n)$-bit messages to each of the other $n-1$ nodes at every round. In the UCC model, many natural problems can be solved in a constant number of rounds [17, 35, 40]. The UCC model is very powerful, and it has actually been proved [21] that it can simulate powerful bounded-depth circuits classes, from which it follows that exhibiting non-trivial lower bounds for the UCC model is quite difficult. The *broadcast* variant of the congested clique, namely the BCC model, is significantly weaker than the unicast variant, and lower bounds on the round-complexity of problems in the BCC model have been established, again by reduction to communication complexity problems. This is the case of problems such as detecting the presence of particular subgraphs [21], detecting planted cliques [18], or approximating the diameter of the network [33]. Obviously, many fast, non-trivial BCC-algorithms have also been devised. As examples, we can mention the sub-logarithmic deterministic algorithm that finds a maximal spanning forest in $O(\log n/\log\log n)$ rounds [34], and algorithms for deciding and reconstructing several graph families (including bounded degeneracy graphs) performing in a constant number of rounds [13]. It is worth noticing that, for single round algorithms, the BCC model is also referred to using other terminologies, such as *simultaneous-messages* [8], or *sketches* [2, 54]. In these latter models though, the measure of complexity is the size of the messages, and therefore the restriction to $O(\log n)$-bits messages is not enforced.

Hybrid distributed computing models have been investigated in the literature only recently, motivated by the various forms of modern communication technologies, from high-throughput optical links to global wireless communication facilities, to peer-to-peer long-distance logical connections. In particular, a hybrid model allowing nodes to perform in a *local* mode, and in a *global* mode at each round has been recently considered [7]. The local mode corresponds to perform a LOCAL round [47], while the global mode corresponds to perform a *node-capacitated clique* (NCC) round [6], which allows each node to exchange $O(\log n)$-bit messages with $O(\log n)$ arbitrary nodes in the network. It is shown that, in the LOCAL+NCC hybrid model, SSSP can be approximated in $\widetilde{O}(n^{1/3})$ rounds, and APSP can be approximated in $\widetilde{O}(\sqrt{n})$ rounds. Several lower bounds are also presented in [7], including an $\widetilde{\Omega}(\sqrt{n})$-round lower bound for computing APSP, and an $\Omega(n^{1/3})$-round lower bound for computing the diameter. In a subsequent work [39], it was shown that APSP can actually be solved exactly in $\widetilde{O}(\sqrt{n})$ rounds in the LOCAL+NCC model. Some of these results were further improved in [15, 16] where it is shown how to solve multiple SSSP problems exactly in

$\widetilde{O}(n^{1/3})$ rounds, and how to approximate SSSP in $\widetilde{O}(n^{5/17})$ rounds. Other graph problems, such as spanning tree, maximal independent set (MIS) construction, and routing were also considered in the LOCAL+NCC model (see [20, 32]). In fact, it was very recently shown [3] that any problem on sparse graphs can be solved in $\widetilde{O}(\sqrt{n})$ rounds in the LOCAL+NCC model. Efficient distributed algorithms for general graphs in this model can then be obtained using sparsification techniques. Finally, it is worth pointing out that the weaker hybrid model CONGEST+NCC was considered in [26] for restricted families of graphs.

As a final remark, it is interesting to notice that the `XOR-Index` problem is related to the EPR paradox [23], and especially the so-called CHSH game [19] whose objective is to demonstrate the existence of quantum (non-classical) correlations in physics (see [4]).

2 Hybrid Models Based on LOCAL and BCC

In this section, we consider the combination of LOCAL and BCC, and, in particular, we compare the two classed **LB** and **BL**. The section can be considered as a warmup section before stating more complex separation results further in the text.

First, we establish a general result concerning the hybridation of LOCAL and BCC. Recall that $\prod_{i=1}^{k} \mathbf{L}^{\alpha_i}\mathbf{B}^{\beta_i}$ is the class of distributed decision problems that can be decided by an algorithm performing α_1 rounds of LOCAL, then β_1 rounds of BCC, then α_2 rounds of LOCAL, etc., ending with β_k rounds of BCC. We show that every language in this class can be computed in the same number of rounds by performing first all LOCAL rounds, and then all BCC rounds.

▶ **Theorem 1.** *Let $k \geq 1$ be an integer, and let $\alpha_1, \dots, \alpha_k$ and $\beta_1, \dots, \beta_k$ be non-negative integers. We have $\prod_{i=1}^{k} \mathbf{L}^{\alpha_i}\mathbf{B}^{\beta_i} \subseteq \mathbf{L}^{\sum_{i=1}^{k}\alpha_i}\mathbf{B}^{\sum_{i=1}^{k}\beta_i}$.*

Proof. Let $\mathcal{L} \in \prod_{i=1}^{k} \mathbf{L}^{\alpha_i}\mathbf{B}^{\beta_i}$, and let A be a distributed algorithm deciding $\mathcal{L}$ in the corresponding hybrid model combining LOCAL and BCC. Let us consider the maximum integer $t < \sum_{i=1}^{k}(\alpha_i + \beta_i)$ such that A performs BCC at round t, and LOCAL at round $t+1$. (If no such t exist, then A is already in the desired form.) We transform A into A' performing the same as A, excepted that rounds t and $t+1$ are switched. Specifically, let us consider a run of A for an instance (G, ℓ). Let B_u be the message broadcasted by u at round t of A, and, for every neighbor v of u, let $L_{u,v}$ be the message sent by u to v at round $t+1$ of A. To define A', let S_u be the state of every node u at the beginning of round t of A, and let $N_G(u)$ be the set of neighbors of u in G. In A', every node u sends its state S_u to all its neighbors at round t, using LOCAL. At round $t+1$ of A', every node u broadcasts B_u to all nodes, using BCC (this is doable, as u was able to produce B_u based on S_u at round t). Finally, before completing round $t+1$, every node u uses the collection $\{S_v : v \in N_G(u)\}$ and the collection $\{B_w : w \in V(G)\}$ to compute the messages $L_{v,u}$ for all $v \in N_G(u)$, by simulating what every such neighbor would have done v at round t of A. Indeed, $L_{v,u}$ depend solely on S_v and $\{B_w : w \in V(G)\}$. (We make the standard assumption that all nodes are running the same algorithm, but even if that was not the case, every node could also send the code of its algorithm to all its neighbors together with its state at round t.) It follows that, at the end of round $t+1$ of A', every node u can compute its state after $t+1$ rounds of A. By repeating the same switch operation until no LOCAL rounds occur after a BCC round, we eventually obtain an algorithm deciding $\mathcal{L}$ and establishing that $\mathcal{L} \in \mathbf{L}^{\sum_{i=1}^{k}\alpha_i}\mathbf{B}^{\sum_{i=1}^{k}\beta_i}$. ◀

▶ **Corollary 2.** $\mathbf{BL} \subsetneq \mathbf{LB}$.

Proof. The fact that $\mathbf{BL} \subseteq \mathbf{LB}$ is a direct consequence of Theorem 1. On the other hand, there is a distributed language in $\mathbf{LB} \smallsetminus \mathbf{BL}$ since, as shown by Theorem 5, $\mathbf{CB} \smallsetminus \mathbf{BL}^* \neq \varnothing$. ◀

We now show a separation between the class $\mathbf{BL}$ and the class $\mathbf{B}^* \cup \mathbf{L}^*$ of languages that can be decided in a constant number of rounds either in BCC or LOCAL. The proof does not use communication complexity reduction, but a mere reduction to `triangle-freeness`.

▶ **Theorem 3.** $\mathbf{BL} \smallsetminus (\mathbf{B}^* \cup \mathbf{L}^*) \neq \varnothing$

Proof. Let us consider the language `triangle-on-max-degree-freeness (TOMDF)` defined by the set of graphs G such that, for every triangle T in G, all nodes in T have a degree smaller than the maximum degree of G. Note that `TOMDF` $\in \mathbf{BL}$. Indeed, during the BCC round, every node can broadcast its degree. Thus, during the LOCAL round, each node can learn all triangles it belongs to. Every node rejects if it is of maximum degree, and it is contained in a triangle. Otherwise, it accepts. Moreover, `TOMDF` $\notin \mathbf{L}^*$ because, for every $k \geq 0$, in k LOCAL rounds a node cannot distinguish an instance G in which it has maximum degree from an instance G' in which there is a node with a larger degree. It remains to show that `TOMDF` $\notin \mathbf{B}^*$.

Let us assume, for the purpose of contradiction, that there exists $k \geq 0$, such that `TOMDF` can be decided by an algorithm $\mathcal{A}$ performing k BCC rounds, i.e., `TOMDF` $\in \mathbf{B}^k$. We can use $\mathcal{A}$ to decide `triangle-freeness` in $k+1$ BCC rounds. Let G be a graph. In the first BCC round, every node v broadcasts its identifier $\mathsf{id}(v)$ and its degree $d(v)$, and hence learns the maximum degree Δ of G. Then every node simulates $\mathcal{A}$ on the virtual graph G' on $\frac{n\Delta}{2}$ nodes obtained from G by adding a set S_v of $\Delta - d(v)$ pending vertices to each vertex v of G. Every node v simulates $\mathcal{A}$ in G' by simulating its execution on v and on all the nodes in S_v. Specifically, after the first BCC round, v knows the set of IDs used in G, and thus the rank of its ID in this set. Therefore, it can compute the set I composed of the smallest $\frac{n\Delta}{2} - n$ positive integers that are not used as IDs in G. Furthermore, it can assign IDs to its $\Delta - d(v)$ pending virtual neighbors in G', using its rank and the degrees of all the nodes with lower rank in G, so that (1) the ID of each virtual node is unique in G', and (2) every node of G knows the IDs assigned to the pending virtual neighbors of every other node in G. It follows that each node v does not need to simulate the messages broadcasted in $\mathcal{A}$ by the nodes in S_v. In fact, every node v can simulate the behavior of all the virtual nodes in $S = \cup_{u \in V(G)} S_u$ at each round of $\mathcal{A}$. As a consequence, the simulation of $\mathcal{A}$ in G' does not yield any overhead on the number of bits to be broadcasted by each (real) node v running $\mathcal{A}$. After the k BCC rounds of $\mathcal{A}$ in G' have been simulated, every node v accepts (on G) if itself and all the nodes in S_v accept in $\mathcal{A}$ on G'. Now, by construction, $G' \in$ `TOMDF` if and only if G is triangle-free. Since $\mathcal{A}$ decides `TOMDF`, we get that `triangle-freeness` $\in \mathbf{B}^{k+1}$, a contradiction. ◀

3 Hybrid Models Based on BCC and CONGEST

In this section, we consider the combination of CONGEST and BCC, and, in particular, we compare the two classes $\mathbf{CB}$ and $\mathbf{BC}$. The separation of these two classes uses the communication complexity problem `XOR-Index`. In the next section, we will establish that the 2-ways 1-round communication complexity of `XOR-Index` is $\Omega(n)$ bits. We use this lower bounds in the proofs of this section.

We first show that not only $\mathbf{CB} \smallsetminus \mathbf{BC} \neq \varnothing$ but also $\mathbf{CB} \smallsetminus \mathbf{BL}^* \neq \varnothing$.

▶ **Theorem 4.** $\mathbf{CB} \setminus \mathbf{BL}^* \neq \varnothing$. *This result holds even for randomized algorithms performing one* BCC *round followed by a constant number of* LOCAL *rounds, which may err with probability* ϵ*, for every* $\epsilon < 1/5$.

Proof. Let us consider the distributed language denoted `one-marked-edge` defined as

$$\texttt{one-marked-edge} = \Big\{(G,\ell) : \big(\ell : V(G) \to \{0,1\}\big) \\ \wedge \big(\big|\big\{\{u,v\} \in E(G) : \ell(u) = \ell(v) = 1\big\}\big| = 1\big)\Big\}.$$

In words, the language corresponds to the graphs G with a potential mark on each node, satisfying that exactly one edge of G has its two endpoints marked. We have `one-marked-edge` $\in \mathbf{CB}$. Indeed, a simple algorithm consists, for each node, to learn which of its neighbors are marked, in one CONGEST round, and to broadcast its number of marked incident edges, in one BCC round. The nodes reject if the total sum of marked edges is different from 2 (i.e., exactly two nodes are incident to a unique marked edge). They accept otherwise.

We now prove that `one-marked-edge` $\notin \mathbf{BL}^*$. We show that this result holds even for a randomized algorithm which may err with probability $\epsilon < 1/5$. For the purpose of contradiction, let us assume that, for some $k \geq 0$, there exists an ϵ-error algorithm $\mathcal{A}$ solving `one-marked-edge` using one BCC round followed by k consecutive LOCAL rounds. We show how to use $\mathcal{A}$ for designing an ϵ-error 1-round protocol Π solving `XOR-index` by communicating only $\mathcal{O}(\sqrt{m})$ bits on m-bit instances, contradicting the fact that `XOR-index` has communication complexity $\Omega(m)$.

Let $(x,i) \in \{0,1\}^m \times [m]$ and $(y,j) \in \{0,1\}^m \times [m]$ be an instance of `XOR-index`. Without loss of generality, we assume that $m = \binom{n}{2}$ for some $n \in \mathbb{N}$. Let us consider a graph G on $2n+4k$ nodes, composed of two disjoint copies of a clique of size n, plus a path P of $4k$ nodes. Let us denote by G^A and G^B the two cliques. The IDs assigned to the nodes of G^A are picked in $[1,n]$, while the IDs assigned to the nodes of G^B are picked in $[n+1,2n]$. One extremity of P is connected to all nodes in G^A, and the other extremity of P is connected to all nodes in G^B. Let us denote by P^A the $2k$ nodes of P closest to G^A, and by P^B the $2k$ nodes of P closest to G^B. These nodes are assigned IDs $2n+1,\dots,2n+4k$, consecutively, starting from the extremity of P connected to G^A.

We enumerate the $m = \binom{n}{2}$ edges in G^A and G^B from 1 to m. Then, in Π, the players interpret their input vectors x and y as indicators of the edges of G^A and G^B respectively. We denote by G_{xy} the subgraph of G such that, for every $r \in [m]$, the r-th edge e of G^A (resp., G^B) is in G_{xy} if and only if $x_r = 1$ (resp., $y_r = 1$). Also, all edges incident to nodes of P are in G_{xy}. Let $\{u_A^i, v_A^i\}$ be the endpoints of the i-th edge of G_A, and let $\{u_B^j, v_B^j\}$ represent the endpoints of the j-th edge of G_B. (These edges may or may not be in G_{xy} depending on the values of x_j and y_i.) We define $\ell_{ij} : V(G) \to \{0,1\}$ as the marking function such that $\ell_{ij}(w) = 1$ if and only if $w \in \{u_A^i, v_A^i, u_B^j, v_B^j\}$. By construction, we have that $(G_{xy}, \ell_{ij}) \in$ `one-marked-edge` if and only if $((x,i),(y,j))$ is a yes-instance of `XOR-index`, i.e., $x_j \neq y_i$. We say that Alice owns all nodes in $V(G^A) \cup V(P^A)$, and Bob owns all nodes in $V(G^B) \cup V(P^B)$. Observe that the edges of G_{xy} incident to nodes owned by Alice depend only on x, while the edges of G_{xy} incident to nodes owned by Bob only depend on y.

We are now ready to describe Π. First, Alice and Bob simulate the BCC round of algorithm $\mathcal{A}$ on all the nodes of G_{xy} owned by them, respectively, considering that *no vertices are marked*. This simulation results in each player constructing a set of $n+2k$ messages, one for each node of the clique owned by the player, plus one message for each of the $2k$ nodes in the sub-path owned by the player. We denote by M_0^A and M_0^B the set of messages produced by Alice and Bob, respectively. Next, the players repeat the same procedure, but considering now that *all vertices are marked*, from which it results sets of messages denoted by M_1^A and M_1^B, respectively. Finally, Alice sends the pair (M_0^A, M_1^A) to Bob, as well as her input index i. Similarly, Bob sends the pair (M_0^B, M_1^B) to Alice, as well as his input index j. Observe that the size of these messages is $\mathcal{O}((n+k)\log n)$ bits.

After the communication, Alice and Bob decide their outputs as follows. First, each player extracts from M_1^A the messages produced by u_A^j and v_A^j, and extract from M_1^B the messages produced by u_B^i and v_B^i. Then, they extract from M_0^A and M_0^B the messages of every other node. Let us call M the resulting set of messages. Observe that M corresponds exactly to the set of messages communicated during the BCC round of $\mathcal{A}$ on input (G_{xy}, ℓ_{ij}). Then, Alice and Bob simulate the k LOCAL rounds of $\mathcal{A}$ on all the vertices they own. This is possible as the nodes of P are not marked, for every instance of `XOR-index`. Each player accepts if all the nodes owned by this player accept. Since $(G_{xy}, \ell_{ij}) \in$ `one-marked-edge` if and only if $((x,i),(y,j))$ is a yes-instance of `XOR-index`, we get that Π is an ϵ-error protocol solving `XOR-index` on inputs of size m by communicating only $\mathcal{O}((n+k)\log n) = \mathcal{O}(\sqrt{m})$ bits, which is a contradiction with Theorem 7. ◀

We now show that $\mathbf{BC} \setminus \mathbf{CB} \neq \varnothing$.

▶ **Theorem 5.** $\mathbf{BC} \setminus \mathbf{CB} \neq \varnothing$. *This result holds even for randomized algorithms performing one* CONGEST *round followed by one* BCC *round, which may err with probability ϵ, for every $\epsilon < 1/5$.*

Proof. For every $n \geq 2$, let us consider the path P_{2n+1}, i.e., the path with $2n+1$ nodes, denoted consecutively $a_1, \ldots, a_n, c, b_n, \ldots, b_1$. Let $x \in \{0,1\}^n$, $y \in \{0,1\}^n$, $i \in [n]$, and $j \in [n]$. We define the labeling $\ell_{x,y,i,j}$ of the nodes of P_n as follows:

$$\ell_{x,y,i,j}(a_1) = i, \;\; \ell_{x,y,i,j}(a_n) = x, \;\; \ell_{x,y,i,j}(b_n) = y, \;\; \ell_{x,y,i,j}(b_1) = j,$$

and, for every $v \notin \{a_1, a_n, b_1, b_n\}$, $\ell_{x,y,i,j}(v) = \bot$. We define the distributed language

$$\texttt{XOR-index-path} = \{(P_{2n+1}, \ell_{x,y,i,j}) : (n \geq 2) \wedge (x, y \in \{0,1\}^n) \wedge (i,j \in [n]) \wedge (x_j \neq y_i)\}.$$

First, we show that `XOR-index-path` $\in \mathbf{BC}$. During the BCC round, every node broadcasts its ID, and the IDs of its neighbors (a node with more than two neighbors simply rejects). Also, degree-1 nodes broadcasts their labels. Note that the $2n+1$ nodes can then check whether they are vertices of the path P_{2n+1}, and, if this is not the case, they reject. Let i and j be the labels broadcasted by the two extremities of the path. Based on the information broadcasted by all the nodes, each of the two nodes a_n and b_n adjacent to the middle node c of the path knows which of the two labels i or j correspond to the index broadcasted by its farthest extremity in the path, b_1 and a_1, respectively. Thus, during the CONGEST round, a_n and b_n can send the bits x_j and y_i to the center c of the path, which checks whether $x_j \neq y_i$, and accepts or rejects accordingly.

Now, we show that `XOR-index-path` $\notin \mathbf{CB}$. Let us assume for the purpose of contradiction that there exists a 2-round algorithm $\mathcal{A}$ deciding `XOR-index-path` by performing one CONGEST round followed by one BCC round. To solve an instance $((x,i),(y,j))$ of `XOR-Index`, Alice and Bob simulate $\mathcal{A}$ on the path P_{2n+1} with consecutive IDs $1, \ldots, 2n+1$. Specifically, Alice simulates the $n+1$ nodes $a_1, \ldots, a_n, c$, while Bob simulates the $n+1$ nodes $b_1, \ldots, b_n, c$, with the nodes labeled with $\ell_{x,y,i,j}$. For simulating the CONGEST round, Alice sends to Bob the message m_{a_n} sent from a_n to c during that round, and Bob sends to Alice the message m_{b_n} sent from b_n to c during that round. The BCC round is actually simulated simultaneously. More precisely, Alice and Bob can both construct the messages broadcasted by all nodes $a_3, \dots, a_{n-2}$ and $b_3, \dots, b_{n-2}$, merely because they know their IDs and their labels (equal to $\perp$), and they can therefore infer the messages these nodes receive during the CONGEST round. So, these messages do not need to be communicated between the players. Moreover, Alice knows a priori what messages m'_{a_1}, m'_{a_2}, and m'_{a_n} are to be broadcasted by a_1, a_2 and a_n during the BCC round, and can send them to Bob. Symmetrically, Bob knows a priori what messages m'_{b_1}, m'_{b_2}, and m'_{b_n} are to be broadcasted by b_1, b_2 and b_n during the BCC round, and can send them to Alice. As for node c, thanks to the messages m_{a_n} and m_{b_n} sent by Alice to Bob, and by Bob to Alice, respectively, both players can construct the message to be sent by c during the BCC round. So, in total, for simulating $\mathcal{A}$, Alice (resp., Bob) just needs to send the messages $m_{a_n}, m'_{a_1}, m'_{a_2}, m'_{a_n}$ to Bob (resp., the messages $m_{b_n}, m'_{b_1}, m'_{b_2}, m'_{b_n}$ to Alice), which consumes $O(\log n)$ bits of communication in total. Each player accepts if all the nodes he or she simulates accept, and rejects otherwise. Alice and Bob are thus able to solve `XOR-index` by exchanging $\mathcal{O}(\log n)$ bits only, which contradicts Theorem 7. ◀

As a direct consequence of the previous two theorems, we get:

▶ **Corollary 6.** *The sets* **CB** *and* **BC** *are incomparable.*

4 The Communication Complexity of `XOR-index`

This section is dedicated to the analysis of the following communication problem.

> `XOR-index`:
> **Input:** Alice receives $x \in \{0,1\}^n$ and $i \in [n]$; Bob receives $y \in \{0,1\}^n$ and $j \in [n]$.
> **Task:** Alice outputs a boolean out_A and Bob outputs a boolean out_B such that $\text{out}_A \wedge \text{out}_B = x_i \oplus y_i$.

We focus on 2-way 1-round protocols, that is, each player sends only one message to the other player, both players send their messages simultaneously, and each player must decide his or her output upon reception of the message sent by the other player. For every 2-player communication problem P, and for every $\epsilon > 0$, let us denote by $CC^1(P, \epsilon)$ the communication complexity of the best 2-way 1-round randomized protocol solving P with error probability at most ϵ.

▶ **Theorem 7.** *For every non-negative $\epsilon < 1/5$, $CC^1(\texttt{XOR-index}, \epsilon) = \Omega(n)$ bits.*

The rest of the section is entirely dedicated to the proof of Theorem 7. Let $0 \le \epsilon < 1/5$, and let Π randomized protocol solving `XOR-index` with error probability at most ϵ, where Alice communicates k_A bits to Bob, and Bob communicates k_B bits to Alice. Without loss of generality, we can assume that, in Π, Alice (resp., Bob) sends explicitly the value of i (resp., j) to Bob (resp., Alice). Indeed, this merely increases the communication complexity of Π by an additive factor $O(\log n)$, which has no consequence, as we shall show that $k_A + k_B = \Omega(n)$.

Let us consider the probabilistic distribution over the inputs of Alice and Bob, where x and y are drawn uniformly at random from $\{0,1\}^n$, and i and j are drawn uniformly at random from $[n]$. Let us denote X and I the random variables equal to the inputs of Alice, and Y and J the random variables equal to the inputs of Bob. Let M_A (resp., M_B) be the random variable equal to the message sent by Alice (resp., Bob) in Π on input (X, I) (resp., (Y, J)). Note that M_A and M_B have values in $\Omega_A = \{0,1\}^{k_A}$ and $\Omega_B = \{0,1\}^{k_B}$, respectively, of respective size 2^{k_A} and 2^{k_B}.

Let us fix $i, j \in [n]$, $m_A \in \Omega_A$, and $m_B \in \Omega_B$. Let $\mathcal{E}^A_{m_A,j}$ be the event corresponding to Bob receiving $J = j$ as input, and Alice sending $M_A = m_A$ to Bob in the communication round. Similarly, let $\mathcal{E}^B_{m_B,i}$ be the event corresponding to Alice receiving $I = i$ as input, and Bob sending $M_B = m_B$ to Alice in the communication round. For $a, b \in \{0,1\}$, we set:

$$p(a, m_A, j) = \Pr[X_J = a \mid \mathcal{E}^A_{m_A,j}], \text{ and } q(b, m_B, i) = \Pr[Y_I = b \mid \mathcal{E}^B_{m_B,i}],$$

and

$$p(a, j) = \Pr[X_J = a \mid J = j], \text{ and } q(b, i) = \Pr[Y_I = b \mid I = i].$$

Observe that $p(a, j) = q(b, j) = 1/2$. Let a^* and b^* be the most probable values of X_j given (m_A, j), and of Y_i given (m_B, i), respectively. Formally,

$$a^* = \text{argmax}_{a\in\{0,1\}} p(a, m_A, j), \text{ and } b^* = \text{argmax}_{b\in\{0,1\}} q(b, m_B, i).$$

Observe that $p(a^*, m_A, j) \geq 1/2$ and $q(b^*, m_A, j) \geq 1/2$. We first establish the following technical lemma.

▶ **Lemma 8.** *Let $\mathcal{F}$ the the event that Π fails. We have*

$$\Pr[\mathcal{F} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \geq \Pr[a^* \neq X_J \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \cdot \Pr[b^* \neq Y_I \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}].$$

Proof. Without loss of generality, we assume that, in Π, after having communicated the pair (m_A, i), Alice computes b^*, and decides her output as follows. If $b^* \neq x_j$, then Alice accepts with some fixed probability p_A, and if $b^* = x_j$ then Alice accepts with some fixed probability q_A. The probabilities p_A and q_A determines the actions of Alice. Similarly, we can assume that, after having communicated (m_B, j), Bob computes a^*, and decides as follows. If $a^* \neq y_i$ then he accepts with some fixed probability p_B, and if $a^* = y_i$ then he accepts with some fixed probability q_B. Note that, in the case where the players do not take in account the value of a^* and b^*, then one can simply choose $p_A = q_A$ and $p_B = q_B$. Let us denote

$$R_A = \Pr[a^* = X_J \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}], \text{ and } R_B = \Pr[b^* = Y_I \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}].$$

Observe that

$$\Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] = {}^1\!/_2 \Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] + {}^1\!/_2 \Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I].$$

Now, conditioned on $X_J \neq Y_I$, the event $\overline{\mathcal{F}}$ corresponds to the event when Alice accepts and Bob accepts. Observe that, conditioned on $\mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}$, these two latter events are independent. Moreover, conditioned on $X_J \neq Y_I$, the event $a^* \neq Y_I$ is equal to the event $a^* = X_J$. Similarly, conditioned on $X_J \neq Y_I$, the event $b^* \neq X_J$ is equal to the event $b^* = X_I$. It follows that

$$\begin{cases} \Pr[\text{Alice accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] = R_B\, p_A + (1 - R_B)\, q_A; \\ \Pr[\text{Bob accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] = R_A\, p_B + (1 - R_A)\, q_B. \end{cases}$$

This implies that

$$\begin{aligned}
&\Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] \\
&\quad = \Pr[\text{Alice accepts and Bob accepts } \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] \\
&\quad = \Pr[\text{Alice accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] \cdot \Pr[\text{Bob accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J \neq Y_I] \\
&\quad = \big(R_B\, p_A + (1 - R_B)\, q_A\big) \cdot \big(R_A\, p_B + (1 - R_A)\, q_B\big).
\end{aligned}$$

let us now consider the case when conditioning on $X_J = Y_I$. In this case, the event $\overline{\mathcal{F}}$ corresponds to the complement of the event when Alice accepts and Bob accepts. Observe that, conditioned on $X_J = Y_I$, the event $a^* \neq Y_I$ is equal to the event $a^* \neq X_J$, and the event $b^* \neq X_J$ is equal to the event $b^* \neq Y_I$. It follows that

$$\begin{cases} \Pr[\text{Alice accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I] = (1 - R_B)\, p_A + R_B\, q_A; \\ \Pr[\text{Bob accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I] = (1 - R_A)\, p_B + R_A\, q_B/ \end{cases}$$

This implies that

$$\begin{aligned} &\Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I)] \\ &=1 - \Pr[\text{Alice accepts and Bob accepts } \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I] \\ &=1 - \Pr[\text{Alice accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I] \\ &\quad \cdot \Pr[\text{Bob accepts} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}, X_J = Y_I] \\ &=1 - \big((1 - R_B)\, p_A + R_B\, q_A\big) \cdot \big((1 - R_A)\, p_B + R_A\, q_B\big). \end{aligned}$$

Therefore, by combining the two cases, we get that

$$\begin{aligned} &\Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \\ &\quad = \frac{1}{2}\big(R_A\,(p_A + q_A)\,(p_B - q_B) + R_B\,(p_A - q_A)\,(p_B + q_B) + 1 - p_A\, p_B + q_A\, q_B\big). \end{aligned}$$

Conditioned to the events $\mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}$, the best protocol Π corresponds to the one that picks the values of p_A, q_A, p_B, q_B that maximize the previous quantity, restricted to the fact that p_A, q_A, p_B, q_B, R_A and R_B must be values in $[0, 1]$, and that R_A and R_B must be at least $1/2$. The maximum can be found using the Karush-Kuhn-Tucker (KKT) conditions [51]. In fact, as the restrictions are affine linear functions, the optimal value is one solution of the following system of equations:

$$\begin{array}{rcl} (R_A + R_B - 1)p_B - (R_A - R_B)q_B - 2\mu_1 + 2\mu_5 & = & 0 \\ (R_A + R_B - 1)p_A + (R_A - R_B)q_A - 2\mu_2 + 2\mu_6 & = & 0 \\ (R_A - R_B)p_B - (R_A + R_B - 1)q_B - 2\mu_3 + 2\mu_7 & = & 0 \\ -(R_A - R_B)p_A - (R_A + R_B - 1)q_A - 2\mu_4 + 2\mu_8 & = & 0 \\ \mu_1(p_A - 1) & = & 0 \\ \mu_2(p_B - 1) & = & 0 \\ \mu_3(q_A - 1) & = & 0 \\ \mu_4(q_B - 1) & = & 0 \\ -\mu_5 p_A & = & 0 \\ -\mu_6 p_B & = & 0 \\ -\mu_7 q_A & = & 0 \\ -\mu_8 q_B & = & 0 \end{array}$$

From the set of solutions to this system, we obtain that $\Pr[\overline{\mathcal{F}} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}]$ is upper bounded by $R_A + R_B - R_A R_B$. Finally, observe that

$$(1 - R_A)(1 - R_B) = \Pr[a^* \neq X_J \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \cdot \Pr[b^* \neq Y_I \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}],$$

from which we get that

$$\Pr[\mathcal{F} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \geq \Pr[a^* \neq X_J | \mathcal{E}^A_{m_A,j}] \cdot \Pr[b^* \neq Y_I \mathcal{E}^B_{m_B,i}],$$

as claimed. ◀

We now show that, whenever the messages sent by Alice and Bob are too small, the distributions of a^* and of b^* is not far from the uniform. We make use of some basic definitions and tools on information complexity, and we refer to [49] for more details. Let (Ω,μ) be a discrete probability space. Given a random variable X we denote by $p_X:\Omega\mapsto\mathbb{R}$ the discrete density function of X, i.e., $p_X(\omega)=\Pr[X=\omega]$. We denote by $\mathbb{H}:\Omega\mapsto\mathbb{R}^+$ the entropy function, defined as $\mathbb{H}(X)=\sum_{\omega\in\Omega}p_X(\omega)\frac{1}{\log p_X(\omega)}$. Recall that, given two random variables X,Y on Ω, the entropy of X conditioned to Y is

$$\mathbb{H}(X \mid Y)=\mathbb{E}_{p_Y(y)}(H(X \mid Y=y)).$$

Moreover, let μ and ν be two probability measures on Ω. The total variation distance between μ and ν is defined as $|u-v|_{\mathrm{TV}}=\sup_{E\subseteq\Omega}|\mu(E)-\nu(E)|$. It is known that $|u-v|_{\mathrm{TV}}=\frac{1}{2}\sum_{\omega\in\Omega}|\mu(\omega)-\nu(\omega)|$. In addition, the Kullback-Liebler divergence between μ and ν is defined as $\mathbb{D}(\mu||\nu)=\sum_{\omega\in\Omega}\mu(\omega)\log\frac{\mu(\omega)}{\nu(\omega)}$. Given two random variables X and Y, their mutual information is defined as $\mathbb{I}(X;Y)=\mathbb{D}(p_{X,Y}||p_Xp_Y)$. It is known that

$$\mathbb{I}(X;Y)=\mathbb{H}(X)-\mathbb{H}(X \mid Y)=\mathbb{H}(Y)-\mathbb{H}(Y \mid X)=\mathbb{I}(Y;X).$$

Finally, the mutual information of X,Y conditioned on a random variable Z is defined as the function $\mathbb{I}(X;Y \mid Z)=\mathbb{E}_{p_Z(z)}[\mathbb{I}(X;Y \mid Z=z)]$. Having all these notions at hand, we shall use the following technical lemmas:

▶ **Lemma 9** (Theorem 6.12 in [49]). *Let $A_1,\ldots,A_n$ be independent random variables, and let B be jointly distributed. We have $\sum_{i=1}^n\mathbb{I}(A_i;B)\le\mathbb{I}(A_1,\ldots,A_n;B)$.*

▶ **Lemma 10** (Pinsker's Inequality, Lemma 6.13 in [49]). *Let μ,ν be two probability measures over Ω. We have $|u-v|^2_{TV}\ \le\frac{2}{\ln 2}\mathbb{D}(\mu||\nu)$.*

Back into our problem, we observe that:

$$\begin{cases}\mathbb{I}(X_J;M_A \mid J) &= \frac{1}{n}\sum_{j\in[n]}\mathbb{I}(X_j;M_A) &\le \frac{\mathbb{I}(X;M_A)}{n} &\le \frac{\mathbb{H}(M_A)}{n} &\le \frac{k_A}{n}\\ \mathbb{I}(Y_I;M_B \mid I) &= \frac{1}{n}\sum_{i\in[n]}\mathbb{I}(Y_i;M_B) &\le \frac{\mathbb{I}(Y;M_B)}{n} &\le \frac{\mathbb{H}(M_B)}{n} &\le \frac{k_B}{n}.\end{cases}$$

By Pinsker's inequality, it follows that:

$$\begin{cases}\mathbb{E}_{(m_A,j)}\left(|p(\cdot,m_A,j)-p(\cdot,j)|_{\mathrm{TV}}\right) &\le \sqrt{\frac{k_A}{n}}\\ \mathbb{E}_{(m_B,i)}\left(|q(\cdot,m_B,i)-q(\cdot,i)|_{\mathrm{TV}}\right) &\le \sqrt{\frac{k_B}{n}}\end{cases}$$

These latter bounds imply that

$$\begin{cases}\mathbb{E}_{(m_A,j)}\left(p(a^*,m_A,j)\right) &\le \frac{1}{2}+\sqrt{\frac{k_A}{n}}\\ \mathbb{E}_{(m_B,i)}\left(q(b^*,m_B,i)\right) &\le \frac{1}{2}+\sqrt{\frac{k_B}{n}}\end{cases}$$

Now, from Lemma 8, we have that

$$\begin{aligned}\Pr[\mathcal{F} \mid \mathcal{E}^A_{m_A,j},\mathcal{E}^B_{m_B,i}] &\ge p(1-a^*,m_A,j)\cdot q(1-b^*,m_B,i)\\ &\ge \big(1-p(a^*,m_A,j)\big)\cdot\big(1-q(b^*,m_B,i)\big).\end{aligned}$$

As a consequence, we have

$$\begin{aligned}
\Pr[\mathcal{F}] &= \mathbb{E}_{m_A,m_B,i,j}(\Pr[\mathcal{F} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}]) \\
&= \sum_{m_A,m_B,i,j} \Pr[\mathcal{F} \mid \mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \cdot \Pr[\mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \\
&\geq \sum_{m_A,m_B,i,j} \left(1 - p(a^*_{(m_A,j)}, m_A, j)\right) \left(1 - q(b^*_{(m_B,i)}, m_B, i)\right) \Pr[\mathcal{E}^A_{m_A,j}, \mathcal{E}^B_{m_B,i}] \\
&= \sum_{m_A,m_B,i,j} \left(1 - p(a^*_{(m_A,j)}, m_A, j)\right) \left(1 - q(b^*_{(m_B,i)}, m_B, i)\right) \Pr[\mathcal{E}^A_{m_A,j}] \cdot \Pr[\mathcal{E}^B_{m_B,i}] \\
&= \sum_{m_A,j} \left(1 - p(a^*_{(m_A,j)}, m_A, j)\right) \Pr[\mathcal{E}^A_{m_A,j}] \cdot \sum_{m_B,i} \left(1 - q(b^*_{(m_B,i)}, m_B, i)\right) \Pr[\mathcal{E}^B_{m_B,i}] \\
&= \left(1 - \mathbb{E}_{(m_A,j)}\left(p(a^*_{(m_a,j)}, m_A, j)\right)\right) \cdot \left(1 - \mathbb{E}_{(m_B,i)}\left(q(b^*_{(m_B,i)}, m_B, i)\right)\right) \\
&\geq \left(\frac{1}{2} - \sqrt{\frac{k_A}{n}}\right) \cdot \left(\frac{1}{2} - \sqrt{\frac{k_B}{n}}\right).
\end{aligned}$$

Since $\Pr[\mathcal{F}] \leq \epsilon$, we must have $(\frac{1}{2} - \sqrt{\frac{k_A}{n}}) \cdot (\frac{1}{2} - \sqrt{\frac{k_B}{n}}) \leq \varepsilon \leq 1/5$, implying that $k_A = \Omega(n)$ or $k_B = \Omega(n)$.

5 Conclusion

In this paper, we have performed an extensive study of 2-round hybrid models resulting from mixing LOCAL, CONGEST, and BCC, and we obtained a complete picture of the relative power of these models (see Figure 1). This is a first step toward approaching the minimization problem expressed in Eq. (1), which asks for identifying the best combination of these three models for which there is an algorithm that solves a given distributed decision problem $\mathcal{L} \in \mathbf{L}^*\mathbf{B}^*$ with a minimum number of rounds, or at minimum cost. Solving this minimization problem appears to be currently out of reach, but this paper provides some knowledge about the computational power of hybrid models. Concretely, a step forward in the direction of solving the problem of Eq. (1) would be to determine whether most hybrid models remain incomparable when allowing t rounds for $t > 2$. In particular, in the case of hybrid models mixing LOCAL and BCC, we have shown that one can systematically assume that all LOCAL rounds are performed before all the BCC rounds. This does not holds for CONGEST and BCC, for 2-round algorithms. However, we do not know whether the classes $\prod_{i=1}^{k} \mathbf{B}^{\beta_i}\mathbf{C}^{\gamma_i}$ and $\prod_{i=1}^{k} \mathbf{B}^{\beta'_i}\mathbf{C}^{\gamma'_i}$ are systematically incomparable for all distinct sequences $((\beta_i, \gamma_i) : i = 1, \ldots, k)$ and $((\beta'_i, \gamma'_i) : i = 1, \ldots, k)$ such that $\sum_{i=1}^{k}(\beta_i + \gamma_i) = \sum_{i=1}^{k}(\beta'_i + \gamma'_i)$.

The line of research investigated in this paper could obviously be carried out by considering other models as well, in particular other congested clique models like UCC and NCC. It is easy to see that $\mathbf{U}$, the class of distributed languages that can be decided in one round in the unicast congested clique, is incomparable with the largest class of models considered in this paper. Namely, $\mathbf{U} \setminus \mathbf{L}^*\mathbf{B}^* \neq \varnothing$ and $\mathbf{L}^*\mathbf{B}^* \setminus \mathbf{U} \neq \varnothing$. Also, previous work on the hybrid model combining LOCAL and NCC reveals that computing the diameter of the network cannot be done in a constant number of rounds in this model. Taking this under consideration, it could be interesting to study the class $(\mathbf{LN})^*$ of distributed languages that can be decided in a constant number of rounds in the hybrid model combining LOCAL and NCC, where $(\mathbf{LN})^* = \cup_{t\geq 0}(\mathbf{LN})^t$, and $\mathbf{N}$ denotes the class of languages decidable in one round in the node-capacitated clique.

References

1 Amir Abboud, Keren Censor-Hillel, Seri Khoury, and Ami Paz. Smaller cuts, higher lower bounds. *ACM Transactions on Algorithms (TALG)*, 17(4):1–40, 2021.

2 Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Analyzing graph structure via linear measurements. In *23rd ACM-SIAM symposium on Discrete Algorithms*, pages 459–467, 2012.

3 Ioannis Anagnostides and Themis Gouleakis. Deterministic distributed algorithms and lower bounds in the hybrid model. In *35th International Symposium on Distributed Computing (DISC)*, volume 209 of *LIPIcs*, pages 5:1–5:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

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

5 Czumaj Artur and Christian Konrad. Detecting cliques in congest networks. *Distributed Computing*, 33(6):533–543, 2020.

6 John Augustine, Mohsen Ghaffari, Robert Gmyr, Kristian Hinnenthal, Christian Scheideler, Fabian Kuhn, and Jason Li. Distributed computation in node-capacitated networks. In *31st ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 69–79, 2019.

7 John Augustine, Kristian Hinnenthal, Fabian Kuhn, Christian Scheideler, and Philipp Schneider. Shortest paths in a hybrid network model. In *31st ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1280–1299, 2020.

8 László Babai, Anna Gál, Peter G Kimmel, and Satyanarayana V Lokam. Communication complexity of simultaneous messages. *SIAM Journal on Computing*, 33(1):137–166, 2003.

9 Alkida Balliu, Sebastian Brandt, Dennis Olivetti, Jan Studený, Jukka Suomela, and Aleksandr Tereshchenko. Locally checkable problems in rooted trees. In *40th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 263–272, 2021.

10 Reuven Bar-Yehuda, Keren Censor-Hillel, and Gregory Schwartzman. A distributed $(2+\varepsilon)$-approximation for vertex cover in $O(\log \Delta/\varepsilon \log\log \Delta)$ rounds. *Journal of the ACM*, 64(3):1–11, 2017.

11 Leonid Barenboim, Michael Elkin, and Cyril Gavoille. A fast network-decomposition algorithm and its applications to constant-time distributed computation. *Theoretical Computer Science*, 751:2–23, 2018.

12 Florent Becker, Adrian Kosowski, Martín Matamala, Nicolas Nisse, Ivan Rapaport, Karol Suchan, and Ioan Todinca. Allowing each node to communicate only once in a distributed system: shared whiteboard models. *Distributed Comput.*, 28(3):189–200, 2015.

13 Florent Becker, Martin Matamala, Nicolas Nisse, Ivan Rapaport, Karol Suchan, and Ioan Todinca. Adding a referee to an interconnection network: What can (not) be computed in one round. In *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 508–514, 2011.

14 Keren Censor-Hillel, Eldar Fischer, Gregory Schwartzman, and Yadu Vasudev. Fast distributed algorithms for testing graph properties. *Distributed Comput.*, 32(1):41–57, 2019.

15 Keren Censor-Hillel, Dean Leitersdorf, and Volodymyr Polosukhin. Distance computations in the hybrid network model via oracle simulations. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 187 of *LIPIcs*, pages 21:1–21:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

16 Keren Censor-Hillel, Dean Leitersdorf, and Volodymyr Polosukhin. On sparsity awareness in distributed computations. In *33rd ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 151–161, 2021.

17 Yi-Jun Chang, Manuela Fischer, Mohsen Ghaffari, Jara Uitto, and Yufan Zheng. The complexity of $(\Delta+1)$ coloring in congested clique, massively parallel computation, and centralized local computation. In *ACM Symposium on Principles of Distributed Computing (PODC)*, pages 471–480, 2019.

18 Lijie Chen and Ofer Grossman. Broadcast congested clique: Planted cliques and pseudorandom generators. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*, pages 248–255, 2019.

19 John F. Clauser, Michael A. Horne, Abner Shimony, and Richard A. Holt. Proposed experiment to test local hidden-variable theories. *Phys. Rev. Lett.*, 23(15):880–884, 1969.

20 Sam Coy, Artur Czumaj, Michael Feldmann, Kristian Hinnenthal, Fabian Kuhn, Christian Scheideler, Philipp Schneider, and Martijn Struijs. Near-shortest path routing in hybrid communication networks. In *25th International Conference on Principles of Distributed Systems (OPODIS)*, volume 217 of *LIPIcs*, pages 11:1–11:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

21 Andrew Drucker, Fabian Kuhn, and Rotem Oshman. On the power of the congested clique model. In *Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing*, pages 367–376, 2014.

22 Andrew Drucker, Fabian Kuhn, and Rotem Oshman. On the power of the congested clique model. In Magnús M. Halldórsson and Shlomi Dolev, editors, *ACM Symposium on Principles of Distributed Computing, PODC '14, Paris, France, July 15-18, 2014*, pages 367–376. ACM, 2014. doi:10.1145/2611462.2611493.

23 Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47(10):777–780, 1935.

24 Michael Elkin. An unconditional lower bound on the time-approximation trade-off for the distributed minimum spanning tree problem. *SIAM Journal on Computing*, 36(2):433–456, 2006.

25 Guy Even, Orr Fischer, Pierre Fraigniaud, Tzlil Gonen, Reut Levi, Moti Medina, Pedro Montealegre, Dennis Olivetti, Rotem Oshman, Ivan Rapaport, and Ioan Todinca. Three notes on distributed property testing. In *31st International Symposium on Distributed Computing (DISC)*, volume 91 of *LIPIcs*, pages 15:1–15:30. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017.

26 Michael Feldmann, Kristian Hinnenthal, and Christian Scheideler. Fast hybrid network algorithms for shortest paths in sparse graphs. In *24th International Conference on Principles of Distributed Systems (OPODIS)*, volume 184 of *LIPIcs*, pages 31:1–31:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020.

27 Laurent Feuilloley and Pierre Fraigniaud. Survey of distributed decision. *Bull. EATCS*, 119, 2016.

28 Pierre Fraigniaud, Amos Korman, and David Peleg. Towards a complexity theory for local distributed computing. *J. ACM*, 60(5):35:1–35:26, 2013.

29 Pierre Fraigniaud and Dennis Olivetti. Distributed detection of cycles. *ACM Trans. Parallel Comput.*, 6(3):12:1–12:20, 2019.

30 Pierre Fraigniaud, Ivan Rapaport, Ville Salo, and Ioan Todinca. Distributed testing of excluded subgraphs. In *30th International Symposium on Distributed Computing (DISC)*, volume 9888 of *LNCS*, pages 342–356. Springer, 2016.

31 Mika Göös and Jukka Suomela. Locally checkable proofs in distributed computing. *Theory Comput.*, 12(1):1–33, 2016.

32 Thorsten Götte, Kristian Hinnenthal, Christian Scheideler, and Julian Werthmann. Time-optimal construction of overlay networks. In *40th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 457–468. ACM, 2021.

33 Stephan Holzer and Nathan Pinsker. Approximation of distances and shortest paths in the broadcast congest clique. In *19th International Conference On Principles Of Distributed Systems (OPODIS)*, 2016.

34 Tomasz Jurdziński and Krzysztof Nowicki. Connectivity and minimum cut approximation in the broadcast congested clique. In *International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, pages 331–344. Springer, 2018.

35 Tomasz Jurdziński and Krzysztof Nowicki. Mst in $O(1)$ rounds of congested clique. In *29th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2620–2632, 2018.

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

37 Amos Korman, Shay Kutten, and David Peleg. Proof labeling schemes. *Distributed Comput.*, 22(4):215–233, 2010.

38 Fabian Kuhn, Thomas Moscibroda, and Rogert Wattenhofer. What cannot be computed locally! In *23rd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 300–309, 2004.

39 Fabian Kuhn and Philipp Schneider. Computing shortest paths and diameter in the hybrid network model. In *39th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 109–118, 2020.

40 Christoph Lenzen. Optimal deterministic routing and sorting on the congested clique. In *ACM Symposium on Principles of Distributed Computing (PODC)*, pages 42–50, 2013.

41 Reut Levi, Moti Medina, and Dana Ron. Property testing of planarity in the CONGEST model. *Distributed Comput.*, 34(1):15–32, 2021.

42 Nathan Linial. Locality in distributed graph algorithms. *SIAM J. Comput.*, 21(1):193–201, 1992.

43 Zvi Lotker, Elan Pavlov, Boaz Patt-Shamir, and David Peleg. Mst construction in o (log log n) communication rounds. In *15th ACM sSmposium on Parallel Algorithms and Architectures (SPAA)*, pages 94–100, 2003.

44 Moni Naor, Merav Parter, and Eylon Yogev. The power of distributed verifiers in interactive proofs. In *31st ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1096–115, 2020.

45 Moni Naor and Larry Stockmeyer. What can be computed locally? *SIAM Journal on Computing*, 24(6):1259–1277, 1995.

46 Noam Nisan and Avi Widgerson. Rounds in communication complexity revisited. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 419–429, 1991.

47 David Peleg. *Distributed Computing: A Locality-Sensitive Approach*. SIAM, 2000.

48 David Peleg and Vitaly Rubinovich. A near-tight lower bound on the time complexity of distributed minimum-weight spanning tree construction. *SIAM Journal on Computing*, 30(5):1427–1442, 2000.

49 Anup Rao and Amir Yehudayoff. *Communication Complexity: and Applications*. Cambridge University Press, 2020.

50 Atish Das Sarma, Stephan Holzer, Liah Kor, Amos Korman, Danupon Nanongkai, Gopal Pandurangan, David Peleg, and Roger Wattenhofer. Distributed verification and hardness of distributed approximation. *SIAM Journal on Computing*, 41(5):1235–1265, 2012.

51 Rangarajan K Sundaram et al. *A first course in optimization theory*. Cambridge university press, 1996.

52 Jukka Suomela. Survey of local algorithms. *ACM Comput. Surv.*, 45(2):24:1–24:40, 2013.

53 Jukka Suomela. Landscape of locality. In *17th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT)*, volume 162 of *LIPIcs*, pages 2:1–2:1, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

54 Huacheng Yu. Tight distributed sketching lower bound for connectivity. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1856–1873, 2021.