On the Communication Complexity of Finding a King in a Tournament

Nikhil S. Mande
University of Liverpool, UK

Manaswi Paraashar
University of Copenhagen, Denmark

Swagato Sanyal
Indian Institute of Technology Kharagpur, India

Nitin Saurabh
Indian Institute of Technology Hyderabad, India

Abstract

A tournament is a complete directed graph. A source in a tournament is a vertex that has no in-neighbours (every other vertex is reachable from it via a path of length 1), and a king in a tournament is a vertex v such that every other vertex is reachable from v via a path of length at most 2. It is well known that every tournament has at least one king. In particular, a maximum out-degree vertex is a king. The tasks of finding a king and a maximum out-degree vertex in a tournament has been relatively well studied in the context of query complexity. We study the *communication complexity* of finding a king, of finding a maximum out-degree vertex, and of finding a source (if it exists) in a tournament, where the edges are partitioned between two players. The following are our main results for n-vertex tournaments:

- We show that the communication task of finding a source in a tournament is *equivalent* to the well-studied Clique vs. Independent Set (CIS) problem on undirected graphs. As a result, known bounds on the communication complexity of CIS [Yannakakis, JCSS'91, Göös, Pitassi, Watson, SICOMP'18] imply a bound of $\widetilde{\Theta}(\log^2 n)$ for finding a source (if it exists, or outputting that there is no source) in a tournament.
- The deterministic and randomized communication complexities of finding a king are $\Theta(n)$. The quantum communication complexity of finding a king is $\widetilde{\Theta}(\sqrt{n})$.
- The deterministic, randomized, and quantum communication complexities of finding a maximum out-degree vertex are $\Theta(n \log n), \widetilde{\Theta}(n)$ and $\widetilde{\Theta}(\sqrt{n})$, respectively.

Our upper bounds above hold for all partitions of edges, and the lower bounds for a specific partition of the edges.

One of our lower bounds uses a fooling-set based argument, and all our other lower bounds follow from carefully-constructed reductions from Set-Disjointness. An interesting point to note here is that while the deterministic query complexity of finding a king has been open for over two decades [Shen, Sheng, Wu, SICOMP'03], we are able to essentially resolve the complexity of this problem in a model (communication complexity) that is usually harder to analyze than query complexity.

2012 ACM Subject Classification Theory of computation → Communication complexity; Theory of computation → Quantum complexity theory; Theory of computation → Graph algorithms analysis

Keywords and phrases Communication complexity, tournaments, query complexity

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2024.64

Category RANDOM

Related Version *Full Version URL*: https://arxiv.org/abs/2402.14751 [36]

Funding *Manaswi Paraashar*: M.P. is supported by ERC grant (QInteract, Grant Agreement No 101078107).

Nitin Saurabh: N.S. was supported by the seed grant (SG/IITH/F285/2022-23/SG-123) from IIT Hyderabad.

Acknowledgements We thank an anonymous reviewer for pointing out that Theorem 5 implies a better quantum communication upper bound for finding a king than the bound given in an earlier version of our paper.

1 Introduction

Graph problems have been very widely studied through the lens of query and communication complexity. In the most natural query setting, an algorithm has query access to an oracle that on being input a pair of vertices, outputs whether or not an edge exists between those vertices. In the basic communication complexity setup for graph problems, two parties, say Alice and Bob, are given the information about the edges in E_1 and E_2, respectively, where E_1 and E_2 are disjoint subsets of all possible edges in the underlying graph. Their task, just as in the query model, is to jointly solve a known graph problem on the graph formed by the edges in $E_1 \cup E_2$. Several interesting results are known in these basic query and communication settings in the deterministic, randomized, and quantum models, see, for example, [5, 27, 19, 29, 40, 9, 11] and the references therein.

A prime example of a graph problem whose query complexity and communication complexities have been widely studied is *Graph Connectivity*. The randomized and quantum communication complexities of this problem are known to be $O(n \log n)$ and $\Omega(n)$. This gap has been open for a long time, and the question of closing it has been explicitly asked [29, 27]. On the other hand, its deterministic communication complexity is known to be $\Theta(n \log n)$ [27].

A graph problem that has been extensively studied in the context of communication complexity is the Clique vs. Independent Set (CIS) problem [47, 25, 26, 8]. The CIS problem is so fundamental that it makes an appearance in the first chapter of standard textbooks on communication complexity [32, 41] (in fact, it is defined on the first page of the latter textbook). The CIS problem is parametrized by a graph $G = ([n], E)$, known to both Alice and Bob. Alice is given $C \subseteq [n]$ that forms a clique in G, Bob is given $I \subseteq [n]$ that forms an independent set in G, and their task is to determine whether or not $C \cap I = \emptyset$. Note that if $C \cap I \neq \emptyset$, then it must be the case that $|C \cap I| = 1$. It was long known that the communication complexity of CIS is $O(\log^2 n)$ for all graphs G. More than two decades after this upper bound was discovered, a near-matching lower bound of $\widetilde{\Omega}(\log^2 n)$ was shown to hold for a particular G, in a culmination of a long line of ground-breaking work [31, 28, 3, 45, 25, 26].

▶ **Theorem 1** ([47], [26, Theorem 1.2])**.** *Let G be an n-vertex graph. Then, $\mathsf{D}^{\mathsf{cc}}(\mathsf{CIS}_G) = O(\log^2 n)$. Furthermore, there exists an n-vertex graph G such that $\mathsf{D}^{\mathsf{cc}}(\mathsf{CIS}_G) = \widetilde{\Omega}(\log^2 n)$.*

This lower bound on the communication complexity of CIS also gives the currently-best-known lower bound for the exponent in the famous log-rank conjecture [35]. We remark that the upper bound above also holds if the task is to output the label of the unique intersection of C and I if $C \cap I \neq \emptyset$.

While not as well-studied as the undirected case, communication complexity of directed graph problems has also received some attention in the past (see, for example, [29, 6, 13]). In this work, we consider *tournaments*, which are directed graphs with exactly one directed edge between each pair of vertices (i.e. the underlying undirected graph is complete). We adopt the natural communication complexity setting where Alice knows the orientation of a subset E of the edges, Bob knows the orientation of the remaining edges, and their goal is to jointly solve a known task on the tournament.

A source of a tournament is a vertex with no in-neighbour. The first problem that we study is source-finding: finding the source of a tournament (if it exists, and reporting that no source exists otherwise). The source-finding problem has recently played a central role in the recent breakthrough by Chattopadhyay, Mande and Sherif that refuted the log approximate-rank conjecture [15] which is the randomized analog of the famous log-rank conjecture [35] of communication complexity. It was as also used in the follow-up results [4, 46] that refuted the quantum version of this conjecture. Source-finding has been studied in the context of query complexity and voting theory (see [18] and the references therein). In fact, the problem of finding a source in a tournament (in the bounded-round communication complexity setting) has been studied by Chakrabarti et al. [13, Sections 3, 4] with applications to streaming lower bounds. In a recent preprint, Ghosh and Kuchlous [24] studied the communication complexity of source-finding in general graphs. Interestingly, they showed that source-finding in general directed graphs can be exponentially harder than source-finding in tournaments as demonstrated by our results (Corollary 3).

We denote the source-finding problem in the specific communication setting discussed above by SRC_E (recall that E is the set of edges whose orientation is known to Alice). Perhaps surprisingly, we show that this task is *equivalent* to the CIS problem on undirected graphs.

▶ **Theorem 2.**
- *For all n-vertex graphs $G = ([n], E)$, $\mathsf{D}^{\mathsf{cc}}(\mathsf{CIS}_G) \leq \mathsf{D}^{\mathsf{cc}}(\mathsf{SRC}_E) + O(\log n)$.*
- *For all subsets of edges E of the complete n-vertex graph, there exists an n-vertex graph G such that $\mathsf{D}^{\mathsf{cc}}(\mathsf{SRC}_E) \leq \mathsf{D}^{\mathsf{cc}}(\mathsf{CIS}_G)$.*

Using known near-tight bounds on the communication complexity of CIS (Theorem 1), Theorem 2 immediately yields the following corollary which gives near-tight bounds on the communication complexity of finding a source in a tournament.

▶ **Corollary 3.** *For all subsets E of the edges of the complete n-vertex graph, the deterministic communication complexity of finding a source of a tournament if it exists, or outputting that there is no source is*

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{SRC}_E) = O(\log^2 n).$$

Furthermore, there exists a subset E of edges of the complete n-vertex graph such that the deterministic communication complexity of finding a source is

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{SRC}_E) = \widetilde{\Omega}(\log^2 n).$$

We believe that this equivalence between SRC and CIS will generate further insights into relationships among complexity measures in query and communication settings that are yet to be resolved. Recall that the source-finding function was also recently used to refute the randomized and quantum versions of the log-rank conjecture [15, 4, 46]. In particular, these works showed that the randomized and quantum communication complexities of finding a source in a tournament is polynomially large in the input size. However, in their settings, Alice and Bob each know a bit per edge, and that edge's direction is determined by the bitwise XOR of Alice and Bob's bits for that edge. In view of this, Corollary 3 demonstrates a fundamental difference between the communication complexities of the source-finding problem when the edge directions are partitioned between Alice and Bob, and when Alice and Bob jointly have partial information about each edge.

Motivated to find a "most-dominant vertex" in a tournament, Landau defined the notion of a *king* in a tournament [34]. A king in a tournament is a vertex v such that every other vertex w is either reachable via a path of length 1 or length 2 from v. While it is easy to see that there are tournaments that do not have a source, it is also easy to show that every tournament has a king [34, 38]. If a tournament has a source, then it is a unique king in the tournament. In view of this, a natural variant of SRC_E (and hence CIS, in view of Theorem 2) is the communication task of finding a king in a tournament.

We remark here that the deterministic query complexity of finding a king in an n-vertex tournament is still unknown, and the state-of-the-art bounds are $\Omega(n^{4/3})$ and $O(n^{3/2})$, and are from over 2 decades ago [44]. Recently, [37] essentially resolved the randomized and quantum query complexities of this problem: they showed that the randomized query complexity of finding a king in an n-vertex tournament is $\widetilde{\Theta}(n)$, and the quantum query complexity is $\widetilde{\Theta}(\sqrt{n})$. The complexity of finding a king and natural variants of it have also been fairly well-studied in different contexts [44, 2, 10, 33].

We consider the communication complexity of finding a king in an n-vertex tournament, denoting this task by KING_n. Perhaps surprisingly, while resolving the query complexity of finding a king in a tournament seems hard, we are able to essentially resolve its asymptotic deterministic, randomized, and quantum communication complexities.

▶ **Theorem 4.** *For all disjoint partitions E_1, E_2 of the edges of a tournament, the deterministic, randomized, and quantum communication complexities of finding a king (where Alice knows the edge directions of edges in E_1 and Bob knows the edge directions of edges in E_2) are as follows:*

$$\mathsf{D^{cc}}(\mathsf{KING}_n) = O(n), \quad \mathsf{R^{cc}}(\mathsf{KING}_n) = O(n), \quad \mathsf{Q^{cc}}(\mathsf{KING}_n) = O(\sqrt{n}\log n).$$

Furthermore, there exists a disjoint partition E_1, E_2 such that the deterministic, randomized, and quantum communication complexities of finding a king are as follows:

$$\mathsf{D^{cc}}(\mathsf{KING}_n) = \Omega(n), \quad \mathsf{R^{cc}}(\mathsf{KING}_n) = \Omega(n), \quad \mathsf{Q^{cc}}(\mathsf{KING}_n) = \Omega(\sqrt{n}).$$

In order to show our deterministic and randomized upper bounds, we give a $O(n)$ cost deterministic protocol. Our quantum upper bound follows from the upper bound in Theorem 5 (the upper bound in Theorem 5 is for the problem of finding a vertex of maximum out degree in the same setting, which is always a king [34]). Our lower bounds follow from a carefully constructed reduction from Set-Disjointness. We sketch our proofs in Section 1.1.

Interestingly, our lower bounds actually hold for tournaments that are promised to have exactly 3 kings. It is well known that a tournament cannot have exactly 2 kings [38]. Thus, the only "easier" case than this promised one is that where the input tournament is promised to have exactly one king. This case is handled in Corollary 3 (it is easy to see that a tournament has a unique king iff the unique king is a source in the tournament).

It is folklore [34] that a vertex with maximum out-degree in a tournament is also a king in the tournament. Thus, another natural question that arises is: what is the complexity of finding a maximum out-degree vertex? The deterministic and randomized query complexity of this task is known to be $\Theta(n^2)$, and its quantum query complexity is between $\Omega(n)$ and $O(n^{3/2})$ [7, 37]. Let MOD_n denote the search problem of finding a maximum out-degree vertex in an n-vertex tournament. We study the communication complexity of MOD_n, again in the natural setting where the edges of the tournament are partitioned between Alice and Bob. We show the following:

▶ **Theorem 5.** *For all disjoint partitions E_1, E_2 of the edges of a tournament, the deterministic, randomized, and quantum communication complexities of finding a maximum out-degree vertex (where Alice knows the edge directions of edges in E_1 and Bob knows the edge directions of edges in E_2) are as follows:*

$$\mathsf{D^{cc}}(\mathsf{MOD}_n) = O(n\log n), \quad \mathsf{R^{cc}}(\mathsf{MOD}_n) = O(n\log\log n), \quad \mathsf{Q^{cc}}(\mathsf{MOD}_n) = O(\sqrt{n}\log n).$$

Furthermore, there exist disjoint partitions such that the deterministic, randomized, and quantum communication complexities of finding a maximum out-degree vertex are as follows:[1]

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{MOD}_n) = \Omega(n \log n), \quad \mathsf{R}^{\mathsf{cc}}(\mathsf{MOD}_n) = \Omega(n), \quad \mathsf{Q}^{\mathsf{cc}}(\mathsf{MOD}_n) = \Omega(\sqrt{n}).$$

We direct the reader's attention to the similarity between our communication complexity bounds for MOD_n and known bounds for the communication complexity of Graph Connectivity mentioned earlier in this section: just like in that case we are able to give tight bounds on the deterministic communication complexity, but our bounds are loose by logarithmic factors in the randomized and quantum settings.[2] Our randomized and quantum lower bounds follow using exactly the same reduction from Set-Disjointness as in Theorem 4. Our deterministic lower bound follows by a carefully constructed *fooling set* lower bound. We give a sketch of our proofs in Section 1.1.

While most of the relevant literature of finding kings in tournaments deals with minimizing the number of queries to find a king (which is equivalent to minimizing the depth of a decision tree that solves KING), none deal with minimizing the *size complexity* of a decision tree that solves KING. Logarithm of decision tree size complexity is characterized, upto a log factor in the input size, by the *rank* of the underlying relation (see [36] for definition of size), and these are measures that have gained a significant interest in the past few years in various contexts (see, for instance, [14, 17, 16] and the references therein). While the decision tree depth complexity of KING_n lies between $\Omega(n^{4/3})$ and $O(n^{3/2})$, we show a tight bound of $n-1$ on $\mathrm{rank}(\mathsf{KING}_n)$, which implies and $\Omega(n)$ lower bound and an $O(n \log n)$ upper bound on the logarithm of decision tree size for KING_n. We omit the statement of this result and its proof due to lack of space, and refer the reader to the full version of the paper [36].

1.1 Sketch of proofs of main results

1.1.1 Equivalence of source-finding and CIS

We first sketch the proof of Theorem 2, which is the equivalence of finding a source in a tournament and the Clique vs. Independent Set problem. Below is a sketch of the proof of the first part of this theorem. Consider a graph $G = ([n], E)$, and an input C, I to the Clique vs. Independent Set problem. Here Bob is given $C \subseteq [n]$ which is a clique in G, and Alice is given $I \subseteq [n]$ which is an independent set in G (we switch the order of inputs for convenience). Alice and Bob construct the following instance to the source-finding problem:

- Alice has the edge directions of all edges in E, and Bob has the remaining edge directions in $\overline{E}$.
- Alice constructs her edge directions such that all vertices in I have in-degree 0 with respect to her edge directions in E. This is easy to do since there are no edges between any pair of vertices in I. She also ensures that all vertices in $[n] \setminus I$ have in-degree at least 1, with respect to her edge directions in E. She can ensure this if G is a connected graph. (see Section 3.)
- Just as the above, Bob ensures that all vertices in C have in-degree 0 w.r.t. $\overline{E}$, and all vertices in $[n] \setminus C$ have in-degree at least 1 w.r.t. $\overline{E}$.

[1] The edge partition we use to prove our deterministic lower bound is different from the partition we use to prove our randomized and quantum lower bounds.

[2] After a full version of our work appeared in the public domain [36], Ghosh [23] communicated to us a proof of a matching randomized $\Omega(n \log \log n)$ lower bound in Theorem 5, showing that our randomized upper bound is tight.

Using the properties above, it is not hard to show that $s = C \cap I$ iff s is a source in the tournament jointly constructed by Alice and Bob above. This concludes the reduction from CIS to source-finding.

In the other direction, if Alice is given edge directions for the subset E of edges of the complete n-vertex graph, then the underlying graph G that Alice and Bob construct for the CIS problem is $G = ([n], E)$. For the purpose of this reduction, we assume that Alice has an independent set as input to CIS, and Bob has a clique. Alice considers her input, an independent set, I to the CIS problem to be the set of all vertices with in-degree 0 w.r.t. E (note that these vertices must form an independent set in G), and Bob constructs his input clique C to be all vertices with in-degree 0 w.r.t. his edges (these form a clique w.r.t. E, and hence in G). Note that a source in the initial tournament, if it exists, must be a vertex in $I \cap C$ since it must have in-degree 0 both w.r.t. Alice's and w.r.t. Bob's edges. Moreover this is the only way in which I intersection C is non-empty. In other words, $I \cap C \neq \emptyset$ iff there is a source in the initial tournament. This concludes the reduction from source-finding to CIS, and hence Theorem 2. Known upper bounds and lower bounds on the communication complexity of the Clique vs. Independent Set problem (Theorem 1) then yield Corollary 3.

Some of our proofs of the lower bounds in Theorems 4 and 5 follow the same outline. In the next section, we sketch our upper bounds, and we sketch our lower bounds in the following section.

1.1.2 Upper bounds

We start with ideas behind the upper bounds in Theorem 4. Throughout this paper, we will view a n-vertex tournament as a string $G \in \{0,1\}^{\binom{n}{2}}$, where the indices are labeled by pairs $\{i < j \in [n]\}$ and $G_{i,j} = 1$ means the edge between vertices i and j is directed from i to j. Recall that the goal is to construct a communication protocol for finding a king in a tournament $G \in \{0,1\}^{\binom{n}{2}}$ whose edges are partitioned into E_1 (with Alice) and E_2 (with Bob).

Consider the deterministic communication model. At a high level, our protocol proceeds in rounds, and in each round Alice and Bob reduce the problem to king-finding in a smaller subtournament. In the beginning of each round assume without loss of generality that Alice has a larger number of edges. Alice sends Bob the label of a vertex v with maximum number of out-neighbours in E_1 along with the in-neighbourhood of v in E_1 as a bit-string (one bit for every other vertex u in the current subtournament for which Alice knows the direction of the edge between u and v). Upon receiving v, Bob also sends the in-neighbourhood of v in E_2 as a bit-string. Thus both players know the entire in-neighbourhood of v in the entire tournament by the end of the round. The communication cost so far is at most $2n + \log n = O(n)$, where n is the number of vertices in the current tournament. The players now reduce to finding a king in the in-neighbourhood of v, since by [38] (also see Lemma 11), this would give a king in the tournament G. Since $|E_1| \geq |E_2|$, the number of out-neighbours of v is at least $(n-1)/4$. This yields a communication protocol of cost $T(n)$ that is described by a recurrence of the form $T(n) \leq T(3n/4) + O(n)$, which is easily seen to give a solution of $T(n) = O(n)$. For the quantum upper bound, we note that a maximum out-degree vertex is always a king [34]. Our $O(\sqrt{n} \log n)$ quantum upper bound for finding a king then immediately follows from Theorem 5, which we describe shortly.

We now sketch proofs of the upper bounds in Theorem 5. Our upper bounds follow from communication protocols for the following problem: Alice and Bob are given $A \in [n]^n$ and $B \in [n]^n$, respectively. Their goal is to output an index $i \in [n]$ that maximizes $a_i + b_i$. We call this communication problem $\mathsf{MAXSUM}_{n,n}$. The reduction from MOD_n to $\mathsf{MAXSUM}_{n,n}$ is easy to see: Alice and Bob construct A, B to be the vector of out-degrees of all vertices w.r.t. their edges. Thus a deterministic communication protocol of cost $O(n \log n)$ immediately follows for MOD_n: Alice sends A to Bob, who then computes an answer. We now sketch the randomized upper bound. Let $S = (s_1, \ldots, s_n)$ where $s_i = a_i + b_i$. The first observation is that deciding $s_i \geq s_j$ is equivalent to deciding $a_i - a_j \geq b_j - b_i$. The latter can be done with cost $O(\log \log n)$ and error at most $1/3$ by using the communication protocol of Greater-Than due to [39, Theorem 1] (see Theorem 21). Thus Alice and Bob have access to a "noisy" oracle that decides whether $s_i \geq s_j$, for all $i, j \in [n]$, independently with probability at least $2/3$. Finding $\arg\max_{i \in [n]} s_i$ with error probability $1/3$ can be done by making $O(n)$ such queries (due to [21], see Theorem 20). This gives a protocol with an overall communication cost of $O(n \log \log n)$. The quantum communication protocol is an application of a result of [12], along with a quantum query upper bound for computing argmax (see Theorem 15), see Section 5 for details.

1.1.3 Lower bounds

Our intuition for the lower bounds is that a "hard" partition of edges between Alice and Bob should be such that every vertex has an equal number of incident edges with Alice and with Bob. One such natural partition of the edges is as follows: Alice receives the complete tournament restricted to the first $n/2$ vertices and the complete tournament restricted to the last $n/2$ vertices, and Bob receives all of the edges between these vertices. While we are unable to use this partition of edges to prove a lower bound for KING_n, we do use it to show a deterministic lower bound for MOD_n. Our approach to showing a deterministic communication lower bound for MOD_n is to construct a large *fooling set* (see Lemma 19). More precisely, for a permutation $\sigma \in S$, where S is a suitably chosen large (size $2^{\Omega(n \log n)}$) subset of $\mathcal{S}_n$, we construct inputs A_σ, B_σ to Alice and Bob such that vertex 1 is a unique maximum out-degree vertex for all $\sigma \in S$. We also ensure that "cross-inputs" $(A_\sigma, B_{\sigma'})$ with $\sigma \neq \sigma'$ lead to vertex 1 not being a maximum out-degree vertex as long as σ and σ' are far away in the ℓ_∞ norm, which we force to be true for all permutations in S by our construction. We refer the reader to Section 5 for technical details.

While we are unable to make the same reduction work to show the communication lower bounds for KING_n (and for good reason, since this argument gives an $\Omega(n \log n)$ lower bound, and there is an $O(n)$ upper bound for the communication complexity of KING_n) and randomized and quantum communication lower bounds for MOD_n, our partition constructed there has a similar flavor to that above. A key intermediate function that we consider for showing our remaining lower bounds is a variant of KING inspired by the well-studied Indexing function. Aptly, we name our variant $\mathsf{IndexKING}$, defined below. For a tournament $G \in \{0,1\}^{\binom{n}{2}}$ with vertex set $[n]$, and a set $S \subseteq [n]$, we use the notation $G|_S$ to denote the subtournament of G induced on the vertices in S.

▶ **Definition 6.** *Let $n > 0$ be a positive integer. Define the* $\mathsf{IndexKING}_n$ *communication problem as follows: Alice is given a set $S \subseteq [n]$ and Bob is given a tournament $G \in \{0,1\}^{\binom{n}{2}}$ on n vertices. Their goal is to output a king in $G|_S$.*

We consider the restriction of $\mathsf{IndexKING}$ to those inputs where Bob's tournament is a transitive tournament (see Definition 12). We denote this variant by $\mathsf{t\text{-}IndexKING}$. A moment's observation (see Observation 8) reveals that this problem is equivalently formulated as follows. We name this version the *Permutation Maximum Finding* problem, defined below, and we believe that this problem is of independent interest.

▶ **Definition 7** (Permutation Maximum Finding). *Let $n > 0$ be a positive integer. In the* Permutation Maximum Finding *problem,* PMF_n*, Alice is given as input a subset S of $[n]$, Bob is given a permutation $\sigma \in \mathcal{S}_n$, and their goal is to output*

$$\mathsf{PMF}_n(S, \sigma) = \begin{cases} \bot & S = \emptyset \\ \arg\max_{j \in S} \sigma(j) & S \neq \emptyset. \end{cases}$$

Unless explicitly mentioned otherwise, we assume that Alice's input S to PMF_n is always a non-empty set. In other words, in the PMF problem, Alice is given a subset of $[n]$, Bob is given a ranking of all elements in $[n]$ (here, $\sigma(i)$ denotes the rank of i), and their goal is to find the element in Alice's set that has the largest rank.

▶ **Observation 8.** *Let $n > 0$ be a positive integer. Then,* $\mathsf{cost}(\mathsf{PMF}_n) = \mathsf{cost}(\mathsf{t\text{-}IndexKING}_n)$*, where* $\mathsf{cost} \in \{\mathsf{D^{cc}}, \mathsf{R^{cc}}, \mathsf{Q^{cc}}\}$.[3]

We refer the reader to the full version [36] for a proof.

We show that Set-Disjointness reduces to PMF (see Lemma 28 and its proof). The lower bound results for PMF follow from known results for communication complexity of Set-Disjointness (see Theorem 17).

Next we reduce from PMF_n to KING. Our reduction ensures that an instance (S, σ) to PMF_n gives us a tournament $G_{S,\sigma}$ with the following properties:

- The tournament has $3n$ vertices, partitioned into V_1, V_2, V_3, of n vertices each, each labeled by elements of $[n]$. The internal edges (edges in $\binom{V_1}{2}, \binom{V_2}{2}$ and $\binom{V_3}{2}$) in each of the partitions are with Bob, and these correspond to transitive tournaments defined by σ.
- The remaining "cross" edges are all with Alice, and the directions of these are determined by S (see Figure 1 for details).
- The tournament $G_{S,\sigma}$ has exactly three kings (which are also the three unique maximum out-degree vertices), one in each V_i, and each of these is labeled by $\mathsf{PMF}_n(S, \sigma)$.

Thus finding a king or a maximum out-degree vertex in $G_{S,\sigma}$ amounts to Alice and Bob solving PMF_n, which we've already sketched to be hard via a reduction from Set-Disjointness. An interesting point to note is that this actually shows a lower bound on the communication complexity of finding a king, even when the input tournament is promised to have exactly three kings. Recall that we showed that finding a king can be done with $O(\log^2 n)$ deterministic communication when an input is promised to have exactly one king (Corollary 3). Also it is easy to show using Lemma 11 that there are no tournaments with exactly two kings. Thus, the "easiest" non-trivial case of a promised tournament with exactly three kings is already hard for communication.

2 Preliminaries

Let $[n] = \{1, \ldots, n\}$. We use the notation $\mathrm{polylog}(n)$ to denote $O(\log(n)^c)$ for some fixed constant c. For $f : \mathbb{N} \to \mathbb{N}$, we use the notation $\widetilde{O}(f)$ to denote $O(f \log^{c_1} f)$ and $\widetilde{\Omega}(f)$ to denote $\Omega(f/(\log^{c_2} f))$, for some constants c_1, c_2.

A tournament $G \in \{0,1\}^{\binom{n}{2}}$ is a complete directed graph on n-vertices. For $v, w \in [n]$ such that $v < w$, if $G_{v,w} = 1$ then there is an out-edge from v to w, i.e. $v \to w$ (otherwise there is an out-edge from w to v). In this case we say that v 1*-step dominates* w. Similarly, for $u, w \in [n]$, if there exists a $v \in [n]$ such that $u \to v$ and $v \to w$ then we say that u 2*-step*

[3] We actually prove the stronger statement that the problems PMF_n and $\mathsf{t\text{-}IndexKING}_n$ are equivalent, in the sense that Alice and Bob need not communicate to go one from one problem to another.

dominates w. Let $S \subseteq [n]$ be such that v 2-step (1-step) dominates w for all $w \in S$. We then say that v 2-step (1-step) dominates S. It is easy to see that there are tournaments where no vertex 1-step dominates all other vertices (such a vertex is called the *source* of G). However, it is now folklore that every tournament has a vertex v such that every vertex $w \neq v$ is either 1-step or 2-step dominated by v. Such a vertex is called a *king* of the tournament (see [34]).

▶ **Lemma 9** (Folklore). *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament. Then there exists a vertex $v \in [n]$ such that v is a king of G.*

For a vertex $v \in [n]$, let $N^-(v) = \{w \in [n] : w \to v\}$ and $N^+(v) = \{w \in [n] : v \to w\}$. Thus $N^-(v)$ and $N^+(v)$ denote the in-neighbourhood and out-neighbourhood of v in G, respectively. The in-degree of v, denoted by $d^-(v)$ is defined as $|N^-(v)|$, and similarly the out-degree of v is denoted by $d^+(v)$ and is defined as $|N^+(v)|$. If a vertex has maximum out-degree in the tournament, then that vertex is a king of the tournament (a proof can be found in [38]).

▶ **Lemma 10** ([34]). *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament and $v \in [n]$ be a vertex of maximum out-degree in G. Then v is a king in G.*

For $S \subseteq [n]$ let $G|_S$ be the tournament *induced on S by G*, i.e. $G|_S$ is a tournament with vertex set as S and direction of edges in S are same as that in G.

The following is an important lemma that we use often.

▶ **Lemma 11** ([38]). *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament and $v \in [n]$. If a vertex u is a king in $G|_{N^-(v)}$, then u is a king in G.*

A special class of tournaments is the class of transitive tournaments, which we define next.

▶ **Definition 12** (Transitive Tournament). *A tournament $G \in \{0,1\}^{\binom{n}{2}}$ is transitive if it satisfies the following property: for all $u, v, w \in [n]$, $u \to v$ and $v \to w$ implies $u \to w$.*

In other words, a transitive tournament is a tournament which is a directed acyclic graph.

▶ **Lemma 13** (Properties of Transitive Tournaments). *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a transitive tournament. There is an ordering $v_1, \dots, v_n$ of $[n]$ such that*

- *v_1 is a source vertex and hence a unique king in G, and*
- *for all $i \in \{2, \dots, n\}$, v_i is source vertex in $G|_{[n] \setminus \bigcup_{j=1}^{i-1} \{v_j\}}$.*

Proof. Since G is a directed acyclic graph, a topological sort on the vertices gives a source of the graph. Let this vertex be v_1. The vertex v_i is obtained by applying the same argument over the transitive tournament $G|_{[n] \setminus \bigcup_{j=1}^{i-1} \{v_j\}}$. ◀

2.1 Query and Communication Complexity

We refer the reader to the full version [36] of our paper for the formal setup of deterministic, randomized, and quantum query complexity.

▶ **Definition 14** ($\mathsf{ARGMAX}_{k,n}$). *Let k be a positive integer and let $a \in ([k])^n$. Given query access to a, find $i \in [n]$ such that $a_i \geq a_j$ for all $j \neq i \in [n]$.*

▶ **Theorem 15** ([20]). *There exists a quantum query algorithm for $\mathsf{ARGMAX}_{k,n}$ with query cost $O(\sqrt{n})$.*

We refer the reader to the full version [36] of our paper for the formal setup of deterministic, randomized, and quantum communication complexity.

▶ **Definition 16** (Set-Disjointness). *Let $n > 0$ be a positive integer. The Set-Disjointness problem is denoted by* $\mathsf{DISJ}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *and is defined by*

$$\mathsf{DISJ}_n(A,B) = 1 \iff A \cap B = \emptyset,$$

where $A, B \subseteq [n]$ are the characteristic sets of Alice and Bob's inputs, respectively.

The communication complexity of DISJ_n is extensively studied. We require the following known bounds on its communication complexity [5, 30, 43, 42, 1].

▶ **Theorem 17** (Communication complexity of Set-Disjointness). *The deterministic, randomized, and quantum communication complexity of* DISJ_n *is as follows:*

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{DISJ}_n) = n, \quad \mathsf{R}^{\mathsf{cc}}(\mathsf{DISJ}_n) = \Theta(n), \quad \mathsf{Q}^{\mathsf{cc}}(\mathsf{DISJ}_n) = \Theta(\sqrt{n}).$$

It is a folklore result that, classically, query algorithms for functions give communication protocols for these functions composed with small gadgets with very little blowup in the complexity. In the quantum setup we have the following theorem, that essentially follows from [12].

▶ **Theorem 18** ([12]). *Let $f \subseteq \mathcal{D}_f^n \times \mathcal{R}$ be a relation where $\mathcal{D}_f = [k]$ for some finite k, and let $g : \mathcal{D}_g \times \mathcal{D}_g \to \mathcal{D}_f$ be a function. For all $\varepsilon > 0$, if $\mathsf{Q}_\varepsilon(f) \leq T$ then $\mathsf{Q}^{\mathsf{cc}}_\varepsilon(f \circ g) \leq 2T(\lceil \log n \rceil + \lceil \log k \rceil + \lceil \log |\mathcal{D}_g| \rceil)$.*

We refer the reader to the full version [36] of our paper for a proof.

A *fooling set* for a communication problem $f \subseteq (\mathcal{X} \times \mathcal{Y}) \times \mathcal{R}$ is a set $S \subseteq \mathcal{X} \times \mathcal{Y}$ such that for all pairs $s_1 = (x_1, y_1)$ and $s_2 = (x_2, y_2)$ in S, we have

$$\{r \in \mathcal{R} | (x_1, y_1, r) \in f \wedge (x_1, y_2, r) \in f \wedge (x_2, y_1, r) \in f \wedge (x_2, y_2, r) \in f\} = \emptyset.$$

▶ **Lemma 19.** *Let $f \subseteq (\mathcal{X} \times \mathcal{Y}) \times \mathcal{R}$ be a communication problem, and let $S \subseteq \mathcal{X} \times \mathcal{Y}$ be a fooling set for f. Then, $\mathsf{D}^{\mathsf{cc}}(f) \geq \log |S|$.*

We refer the reader to standard texts for a formal proof [32, Lemma 1.20]. We remark that standard texts usually frame the fooling set lower bound as a lower bound technique for communication complexity of functions rather than relations, but the same proof technique is easily seen to show the statement above as well. A sketch of the proof is as follows: The leaves of a protocol tree of depth c yields a partition of the space $\mathcal{X} \times \mathcal{Y}$ into 2^c rectangles, each of which has at least one $r \in \mathcal{R}$ that is a valid output for all pairs of inputs in the rectangle. By the property of a fooling set, each element of it must belong to a different leaf. This implies the number of leaves in any protocol for f must be at least $|S|$, implying that the depth of any protocol must be at least $\log |S|$.

We require the following theorem that gives an algorithm to find the maximum in a list given noisy comparison oracle access. The formulation we use below follows easily from [21, Theorem 15].

▶ **Theorem 20** ([21, Theorem 15]). *Let $S = (s_1, \ldots, s_n)$ be a list of n numbers. Suppose we have access to a "noisy" oracle, that takes as input a pair of indices $i \neq j \in [n]$, and outputs a bit that equals $\mathbb{I}[s_i \geq s_j]$ with probability at least $2/3$, independent of the outputs to the other queries. Then there is an algorithm that makes $O(n)$ queries to the noisy oracle and outputs $\arg\max_{i \in [n]} s_i$ with probability at least $2/3$.*

▶ **Theorem 21** ([39, Theorem 1]). *Let $n > 0$ be a positive integer. The* $\mathsf{GT} : [n] \times [n] \to \{0,1\}$*, where Alice is given $x \in [n]$ and Bob is given $y \in [n]$ is defined as* $\mathsf{GT}(x,y) = 1$ *if and only if $x \geq y$. The randomized communication complexity of* GT *is $O(\log \log n)$.*

2.2 Formal definitions of graph problems of interest

For clarity and completeness, we include formal definitions of the tasks of finding a king and finding a maximum out-degree vertex in this section.

▶ **Definition 22.** *Let $n > 0$ be a positive integer. Define* $\mathsf{KING}_n \subseteq \{0,1\}^{\binom{n}{2}} \times [n]$ *to be*

$$(G, v) \in \mathsf{KING}_n \iff v \text{ is a king in the tournament } G.$$

▶ **Definition 23.** *Let $n > 0$ be a positive integer. Define* $\mathsf{MOD}_n \subseteq \{0,1\}^{\binom{n}{2}} \times [n]$ *to be*

$$(G, v) \in \mathsf{MOD}_n \iff v \text{ is a maximum out-degree vertex in the tournament } G.$$

When we give communication upper bounds for these problems, our upper bounds hold for all partitions of the input variables $\binom{n}{2}$ between Alice and Bob. When we give lower bounds, we exhibit specific partitions for which our lower bounds hold.

3 Communication complexity of finding a source

We consider the communication complexity of finding a source in a tournament if it exists. Alice knows the edge directions of a subset E_A of the edges of a tournament $T \in \{0,1\}^{\binom{n}{2}}$, Bob knows the directions of the remaining edges E_B, and their goal is to output the label of a source in the whole tournament if it exists, or output that the tournament has no source. Formally, for a partition of edges E_A, E_B of the complete n-vertex graph, define

$$\mathsf{SRC}_{E_A} \colon \{0,1\}^{E_A} \times \{0,1\}^{E_B} \to \{0,1,\ldots,n\} \quad (1)$$

to be $\mathsf{SRC}_{E_A}(a,b) = 0$ if there is no source in the tournament defined by edge directions a, b, and $\mathsf{SRC}_{E_A}(a,b) = i$ if vertex i is the (unique) source in the same tournament. We define the decision version of this problem to be $\mathsf{SRC}^{\mathrm{dec}}_{E_A} : \{0,1\}^{E_A} \times \{0,1\}^{E_B} \to \{0,1\}$. That is, $\mathsf{SRC}^{\mathrm{dec}}_{E_A}$ outputs 0 if there is no source in the tournament, and outputs 1 if there is a source.

Below, we define the celebrated Clique vs. Independent Set problem on an n-vertex graph G [47], which we henceforth abbreviate as CIS_G. The CIS_G problem is associated with an n-vertex undirected graph $G = (V, E)$. In this problem, Alice and Bob both know G. Alice is given as input a clique $x \subseteq [n]$ in G, Bob is given as input an independent set $y \subseteq [n]$, and their goal is to either output that $x \cap y = \emptyset$, or output the label of the (unique) vertex v with $\{v\} = x \cap y$.[4]

There has been a plethora of work on the Clique vs. Independent set problem, see for example, [47, 25, 26, 8]. Of relevance to us is Theorem 1, which gives near-tight bounds on the deterministic communication complexity of this problem.

Perhaps surprisingly, we show that the communication problem of finding a source in a tournament is *equivalent* to the Clique vs. Independent Set problem. Corollary 3 would then immediately follow. We now prove Theorem 2.

[4] Conventionally, the Clique vs. Independent Set problem is phrased as a decision problem, where the task is to determine if $x \cap y$ is empty or non-empty. The known bounds we state here are easily seen to hold for the "search version" that we consider as well.

Proof of Theorem 2. In this proof, we assume for convenience that in the Clique vs. Independent Set Problem, Alice is given an independent set and Bob is given a clique.

- Let $G = (V, E)$ be an n-vertex graph. Let $I, C \subseteq [n]$ be Alice and Bob's input to CIS_G, respectively. Recall that the vertices in I form an independent set in G and the vertices in C form a clique in G. We now describe the reduction from CIS_G to SRC_E. Before delving into the main reduction, we do a preprocessing of small communication cost to make sure that G is connected and the size of the independent set I is at least 3.

  Preprocessing: Bob sends the label of the connected component in G that his clique C is part of. Alice removes from her independent set I, all vertices that aren't part of this connected component. She now sends a bit to Bob to indicate whether $|I| \geq 3$. If not, she further sends labels of the two vertices in I to Bob who then responds with an answer. This requires a total of $O(\log n)$ communication cost. We can therefore assume that the graph G is connected and $|I| \geq 3$ for the rest of the reduction. Alice and Bob locally construct the following inputs to SRC_E (recall that Alice must construct edge directions in E, and Bob must construct the remaining edge directions).

  - Alice orients the edges in E, using Claim 24 and the fact that G is a connected graph, such that only the vertices in I have in-degree 0.
  - Bob orients the edges in $\overline{E}$ as follows. For vertices in C, he orients the edges in their connected components in $\overline{G}$, using Claim 24, such that only the vertices in C have in-degree 0. Next he orients the edges of connected components that don't contain vertices of C. If this connected component is not a tree, he uses Claim 25 to orient the edges such that no vertex has in-degree 0. If the connected component is a tree, he orients the edges in an arbitrary way.

  Let T denote the tournament constructed above. We next show that (I, C) is a 1-input to CIS_G iff there exists a source in T. This would prove the first part of the theorem. Moreover, we show that when there is a source in the constructed tournament, the source vertex is the same as the unique vertex in $I \cap C$.

  Let (I, C) be a 1-input to CIS_G and s be the unique vertex in $I \cap C$. We show that s is the source in the tournament T. By construction, the neighbours of s in E are the outneighbours of s in Alice's input, and the neighbours of s in $\overline{E}$ are the outneighbours of s in Bob's input.

  We prove the contrapositive for the other direction. Let (I, C) be a 0-input to CIS_G, i.e., $I \cap C = \emptyset$. We show that there is no source in T. Vertices in $\overline{I}$ are ruled out from being a source by the orientation of Alice's edges. Now the vertices of I forms a clique in Bob's input, thus they form a connected component that is not a tree (since $|I| \geq 3$). Since this connected component does not contain a single vertex from C (since we assumed $I \cap C = \emptyset$), the construction above (using Claim 25) implies that all vertices in I have in-degree at least 1 w.r.t. Bob's edge directions. Thus, there is no source in the entire tournament.

- In the other direction, let $\{0,1\}^{E_A}$ and $\{0,1\}^{E_B}$ be Alice and Bob's input to SRC_{E_A}, where E_A, E_B form a partition of the edges of the n-vertex complete graph. Say that the tournament formed by these inputs is T. Alice and Bob construct the following instance to the Clique vs. Independent Set problem.

  - The graph is $G = (V, E)$ with $V = [n]$ and $E = E_A$.
  - Alice constructs $I \subseteq [n]$ to be all of the vertices with in-degree 0 w.r.t. E_A. It is easy to see that I forms an independent set in G since any edge between vertices in I causes one of the vertices in I to have in-degree at least 1.
  - Bob constructs $C \subseteq [n]$ to be all of the vertices with in-degree 0 w.r.t. E_B. As in the previous bullet, it is easy to see that C forms an independent set in $\overline{G}$, and hence a clique in G.

Consider the input (I, C) to CIS_G as constructed above. We show now that $I \cap C \neq \emptyset$ iff there is a source in T, which would prove the second part of the theorem since (I, C) and G were constructed using no communication.

Suppose s is a source in T. Since s has in-degree 0 w.r.t. both E_A and E_B, we must have $s \in I \cap C$. Moreover, since every other vertex must have in-degree at least 1, such a vertex is either not in I or not in C. Thus, $s = I \cap C$. In the other direction, suppose $s = I \cap C$. By the construction above, s must have in-degree 0 w.r.t. both E_A and E_B, and hence is a source in T. ◀

▷ Claim 24. Let T be a tree, V be its vertex set and I be an independent set in T. Then there exists an orientation of the edges of T such that exactly the vertices in $V \setminus I$ have in-degree at least 1.

Proof of Claim 24. We now show a procedure to orient the edges such that the set of vertices with in-degree 0 equals the set I. Consider a (left-to-right) listing of subsets of vertices based on their distances from the set I. So if the listing looks like $V_0, V_1, \cdots, V_j, \cdots$, then $V_0 = I$, and $V_j \subseteq V \setminus I$ is the set of vertices such that the length of a shortest path to reach a vertex in I equals j. We orient the edges from $V_i \to V_{i+1}$ for $i \geq 0$. The edges within a partition, say V_i, are oriented arbitrarily. Now using the fact that tree is a connected graph, it is easily seen that every vertex in $V \setminus I$ has in-degree at least 1. Moreover, by our construction, all vertices in $V_0 = I$ has in-degree 0. ◁

▷ Claim 25. Let G be a connected graph that is not a tree. Then, there exists an orientation of the edges of G such that every vertex of G has in-degree at least 1.

Proof of Claim 25. Since G is connected but not a tree, it contains a cycle, say C. Orient the edges of C in a cyclic way to give in-degree 1 to every vertex in C, and then orient the edges "away" from the cycle C (in a manner similar to the proof in Claim 24 where $V_0 = C$ here) to add 1 to in-degrees of vertices in $V \setminus C$. Thus the directed graph so constructed has *no* vertex with in-degree 0. ◁

4 Communication complexity of KING

The proof of Theorem 4 is divided into two parts. We show the upper bounds in Section 4.1 and the lower bounds in Section 4.2.

4.1 Upper bounds on communication complexity of KING_n

We start by proving an $O(n)$ upper bound on the deterministic communication complexity which also implies an $O(n)$ upper bound on the randomized communication complexity.

▶ **Lemma 26.** *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament and let E_1, E_2 be a partition of the edges of G. The deterministic and randomized communication complexity of finding a king of G, where Alice is given E_1 and Bob is given E_2, is upper bounded as follows*

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{KING}_n) = O(n), \qquad \mathsf{R}^{\mathsf{cc}}(\mathsf{KING}_n) = O(n).$$

Proof. The proof follows via the Protocol in Algorithm 1.

Algorithm 1 Deterministic Communication Protocol for KING_n.

1: **Input:** Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament and $E_1, E_2 \subseteq \{(i,j) : i < j \in [n]\}$ be a partition of the edges of G. Alice (Player 1) is given $\{0,1\}^{E_1}$ and Bob (Player 2) is given $\{0,1\}^{E_2}$.
2: $S = [n]$
3: **while** $|E_1| > n$ **and** $|E_2| > n$ **do**
4: $\quad b \leftarrow \arg\max_{i \in \{0,1\}} |E_i|$ ▷ `Ties broken arbitrarily`
5: $\quad v \leftarrow \arg\max_{w \in [n]} \{\text{out-degree}(w) \text{ in } E_b\}$ ▷ `Ties broken arbitrarily`
6: $\quad$ Player b sends to Player $1-b$ the label of v along with a $|S|$-bit indicator vector of the in-neighbourhood of v in E_b
7: $\quad$ Player $1-b$ sends an $|S|$-bit indicator vector of the in-neighbourhood of v in E_{1-b}
8: $\quad S \leftarrow S \cap N^-(v)$
9: $\quad E_1 \leftarrow$ the edges of E_1 that are present in $G|_S$
10: $\quad E_2 \leftarrow$ the edges of E_2 that are present in $G|_S$
11: **end while**
12: **if** $|E_1| \leq n$ **then**
13: $\quad$ Alice sends E_1 to Bob
14: $\quad$ Bob outputs a king of the tournament.
15: **else if** $|E_2| \leq n$ **then**
16: $\quad$ Bob sends E_1 to Alice
17: $\quad$ Alice outputs a king of the tournament.
18: **end if**

Correctness. It is easy to see that in every iteration of the **while** loop, the size of either E_1 or E_2 decreases by at least 1. This shows that our algorithm always terminates.

Let $S^{(i)}$ denote the set S in i'th iteration of the **while** loop, where $S^{(1)} = [n]$. We maintain the invariant that in every iteration of the **while** loop, a king in $G|_{S^{(i+1)}}$ is also a king in $G|_{S^{(i)}}$. This follows easily from Lemma 11 since $S^{(i+1)}$ is obtained from $S^{(i)}$ by restricting to vertices in the in-neighbourhood of some vertex v in Line 8. Assume without loss of generality that the **while** loop terminates with $|E_1| \leq n$. In this case, in Line 13, Alice sends her edges to Bob who outputs a king of G.

Cost. We show that the cost of Protocol 1 is upper bounded by $O(n)$ for all tournaments $G \in \{0,1\}^{\binom{n}{2}}$. Suppose we enter the **while** loop with $|S| = k$. Let $c(k)$ be the number of bits communicated during the execution of the **while** loop. Consider Line 6, and assume without loss of generality that $|E_1| \geq |E_2|$, thus $|E_1| \geq (1/2 \cdot \binom{k}{2})$. Since every edge in E_1 is an out-edge for some vertex (note that E_1 and E_2 are subsets of edges of $G|_S$ due to Line 9 and Line 10) we have $\sum_{u \in S} d^+(v) \geq (1/2 \cdot \binom{k}{2})$ (where the out-degrees are only computed in E_1) and hence by an averaging argument there exists $v \in S$ such that the out-degree of v when restricted to E_1 (and therefore S) is at least $(k-1)/4$. Thus the in-degree of v in S is at most $(3/4 \cdot (k-1))$. Furthermore, in each iteration of the **while** loop, $\lceil \log k \rceil + k$ bits are communicated in Line 6 and k bits are communicated in Line 7. We have the following upper bound on $c(k)$: $c(k) \leq c(3k/4) + \lceil \log k \rceil + 2k$, and thus $c(n) = O(n)$. Also observe that either Line 13 or Line 16 is executed and in each case at most n bits are communicated. Thus the overall number of bits communicated in $O(n)$. ◀

Next, we give an $O(\sqrt{n}\log n)$ cost quantum communication protocol for KING_n. Our quantum communication upper bound is a corollary of Theorem 5 which gives a quantum communication protocol for finding a maximum out-degree vertex in a tournament (such a vertex is also a king, see Lemma 10).

▶ **Lemma 27.** *Let $G \in \{0,1\}^{\binom{n}{2}}$ be a tournament and let E_1, E_2 be a partition of E. The quantum communication complexity, where Alice is given E_1 and Bob is given E_2. Then*

$$\mathsf{Q}^{\mathsf{cc}}(\mathsf{KING}_n) = O(\sqrt{n}\log n).$$

4.2 Lower bounds on communication complexity of KING_n

Next, we prove the lower bound. In order to do this, we first give a lower bound on the communication complexity of PMF_n. Recall that, in this problem, Alice is given as input a subset S of $[n]$, Bob is given a ranking of elements of $[n]$ defined by σ, and their goal is to output the element in S that has the largest rank according to σ.

▶ **Lemma 28.** *The deterministic, randomized, and quantum communication complexity of PMF_n is as follows:*

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{PMF}_n) = \Omega(n), \quad \mathsf{R}^{\mathsf{cc}}(\mathsf{PMF}_n) = \Omega(n), \quad \mathsf{Q}^{\mathsf{cc}}(\mathsf{PMF}_n) = \Omega(\sqrt{n}).$$

Proof. We show that Set-Disjointness reduces to PMF_n and the lemma follows from Theorem 17. We describe the reduction next.

Consider an input to Set-Disjointness, $S, T \subseteq [n]$ where S is with Alice and T is with Bob. Alice and Bob locally construct the following instance of PMF_n: Alice retains her set S, and Bob creates an arbitrary σ such that the following holds:

$$\forall i \neq j \in [n], \qquad (T_i = 0) \wedge (T_j = 1) \implies \sigma(i) < \sigma(j).$$

In other words, Bob creates a permutation σ of $[n]$ that ranks all of the indices in T higher than all of the indices outside T. They then run a protocol for PMF_n with inputs S, σ, let k be the output of this protocol. If $k \in T$ then they return $S \cap T \neq \emptyset$ else they return $S \cap T = \emptyset$.

Correctness. If $\mathsf{PMF}_n(S,\sigma) = \perp$, then the players know (without any additional communication) that $S = \emptyset$ and hence $\mathsf{DISJ}_n(S,T) = 1$. Thus, we may assume $S \neq \emptyset$. Since any protocol for PMF_n must output an index $k \in S$. By Bob's construction of σ, the elements of T are ranked higher than elements that are not in T. Since k is the output of a protocol for PMF_n, k is the highest ranked element in S by σ. Thus if k is not among the top $|T|$ ranked elements, then all elements of S are ranked lower than all elements of T (by Bob's construction of σ) and $S \cap T = \emptyset$. On the other hand if k is among the top $|T|$ ranked elements then $k \in T \cap S$. These conditions can be checked by Bob who has σ and k. ◀

By the equivalence of PMF and the transitive variant of $\mathsf{IndexKING}$ (Observation 8), Lemma 28 implies the same lower bounds on $\mathsf{t\text{-}IndexKING}_n$.

We thus immediately conclude the same lower bounds on the general $\mathsf{IndexKING}$ problem (where Bob's tournament is arbitrary, and need not be transitive).

▶ **Corollary 29.** *The deterministic, randomized, and quantum communication complexity of $\mathsf{IndexKING}_n$ is as follows:*

$$\mathsf{D}^{\mathsf{cc}}(\mathsf{IndexKING}_n) = \Omega(n), \quad \mathsf{R}^{\mathsf{cc}}(\mathsf{IndexKING}_n) = \Omega(n), \quad \mathsf{Q}^{\mathsf{cc}}(\mathsf{IndexKING}_n) = \Omega(\sqrt{n}).$$

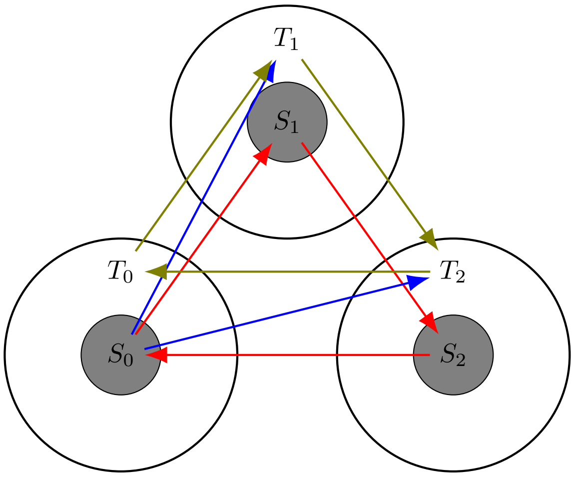

Figure 1 Visual depiction of $G_{S,\sigma}$. For each $b \in \{0,1,2\}$, S_b contains the vertices $\{i_b : i \in S\}$ and T_b contains the vertices $\{i_b : i \notin S\}$. There are four types of edges (also see Definition 30):

- Edges of Type 1 are those within each $T_b \cup S_b$, here $i_b \to j_b$ iff $\sigma(i) > \sigma(j)$.
- Edges of Type 2 are those between S_b and $T_{b'}$ for $b \neq b'$, here $i_b \to j_{b'}$.
- Edges of Type 3 are those between S_b and $S_{b'}$ for $b \neq b'$, here $i_b \to j_{b'}$ iff $b' = b + 1 \pmod 3$.
- Edges of Type 4 are those between T_b and $T_{b'}$ for $b \neq b'$, here $i_b \to j_{b'}$ iff $b' = b + 1 \pmod 3$.

We now give a lower bound on the communication complexity of KING_n. For this we first define a class of tournaments that we use in our proof.

4.3 A class of tournaments

In this section, we define a special class of tournaments on $3n$ vertices, that are parametrized by a subset $S \subseteq [n]$ and an ordering σ of $[n]$.

▶ **Definition 30.** *Given a set $S \subseteq [n]$ and $\sigma \in \mathcal{S}_n$, define the tournament $G_{S,\sigma}$ on $3n$ vertices as follows:*

- *The vertex set is $V = \{i_b : i \in [n], b \in \{0,1,2\}\}$.*
- *For each $b \in \{0,1,2\}$ and all $i \neq j \in [n]$, the direction of the edge between i_b and j_b is $i_b \to j_b$ iff $\sigma(i) > \sigma(j)$. We refer to these as Type 1 edges.*
- *For all $b \neq b' \in \{0,1,2\}$, all $i \in S$ and all $j \notin S$, $i_b \to j_{b'}$ is an edge. We refer to these as Type 2 edges.*
- *For all $b \neq b' \in \{0,1,2\}$ and all $i \neq j \in S$, the direction between the edge i_b and $j_{b'}$ is $i_b \to j_{b'}$ iff $b' = b + 1 (\text{mod } 3)$. We refer to these as Type 3 edges.*
- *For all $b \neq b' \in \{0,1,2\}$ and all $i \neq j \notin S$, the direction between the edge i_b and $j_{b'}$ is $i_b \to j_{b'}$ iff $b' = b + 1 (\text{mod } 3)$. We refer to these as Type 4 edges.*

We refer the reader to Figure 1 for a pictorial representation and some additional notation.

▶ **Lemma 31.** *Let $n > 0$ be a positive integer, $S \subseteq [n]$ and $\sigma \in \mathcal{S}_n$. Then, the tournament $G_{S,\sigma}$ has exactly three kings, namely k_0, k_1, k_2, where $k = \arg\max_{j \in S} \sigma(j)$. Moreover, k_0, k_1, k_2 are the only vertices with maximum out-degree in $G_{S,\sigma}$.*

Proof. We first show that k_0 is a king. The argument for k_1, k_2 being kings follows similarly. To show that k_0 is a king, we exhibit paths of length one or two from k_0 to all other vertices in the tournament.

- First note that for any element $j \in S$, there is an edge from k_0 to j_0 since $k = \arg\max_{j \in S} \sigma(j)$ (this is an edge of Type 1). Thus, k_0 1-step dominates S_0.
- For all $j \notin S$ and $b \in \{1, 2\}$, there is an edge (of Type 2) from k_0 to j_b. Thus, k_0 1-step dominates T_1 and T_2.
- For $j, j' \in S$, there is an edge (of Type 3) from k_0 to j_1. Thus k_0 1-step dominates S_1. There is also an edge (also of Type 3) from j_1 to j'_2. Thus, k_0 2-step dominates S_2.
- For an arbitrary $j \in S$, as noted above, there is an edge from k_0 to j_1. For $j' \notin S$, there is an edge (of Type 2) from j_1 to j'_0. Thus, k_0 2-step dominates T_0.

This shows that k_0 (and similarly k_1 and k_2) is a king in $G_{S,\sigma}$.[5] We next show that no other vertex is a king. We do this by showing for every other vertex k'_b, a vertex that is not 1-step or 2-step dominated by k'_b.

- Consider $k' \neq k \in S$ and $b \in \{0, 1, 2\}$. We now show that k'_b does not 1-step or 2-step dominate k_b.
  - Since k_b is the unique king in the transitive tournament $(G_{S,\sigma})|_{S_b}$ (see Lemma 13), k'_b does not 1-step dominate k_b via Type 1 edges. Moreover, the only vertices that are 1-step dominated by k'_b via Type 1 edges are a subset of vertices in $S_b \cup T_b$. None of these vertices can 1-step dominate k_b since $(G_{S,\sigma})|_{S_b \cup T_b}$ is a transitive tournament. This shows that k'_b cannot 1-step dominate or 2-step dominate k_b by first using an edge of Type 1.
  - The only other out-going edges from k'_b are either of Type 2 or Type 3.
  - Consider a Type 2 edge which goes from k'_b to $T_{b+1 \pmod 3}$ ($T_{b+2 \pmod 3}$ follows similarly). By construction, there is no edge from any vertex in $T_{b+1 \pmod 3}$ to k_b (see Figure 1).
  - Now consider a Type 3 edge which goes from k'_b to $S_{b+1 \pmod 3}$. By construction, there is no edge from any vertex in $S_{b+1 \pmod 3}$ to k_b (see Figure 1).
- Consider $k' \notin S$ and $b \in \{0, 1, 2\}$. We now show that k'_b does not 1-step or 2-step dominate $k_{b+2 \pmod 3}$.
  - The only out-going edges from k'_b are either of Type 1 or Type 4. On taking a Type 1 edge, k'_b can only 1-step dominate a subset of vertices of $S_b \cup T_b$. None of these vertices have an edge to $k_{b+2 \pmod 3}$ (see Figure 1). Thus, k'_b cannot 2-step dominate $k_{b+2 \pmod 3}$ by first taking a Type 1 edge.
  - A Type 4 edge goes from k'_b to a vertex in $T_{b+1 \pmod 3}$. By construction, no vertex in $T_{b+1 \pmod 3}$ has an edge to $k_{b+2 \pmod 3}$ (see Figure 1).

Finally, we observe that k_0, k_1, k_2 are the only three vertices with maximum out-degree in $G_{S,\sigma}$. Observe that the out-degrees of k_0, k_1, k_2 are all equal by symmetry. By Lemma 10, a vertex with maximum out-degree in $G_{S,\sigma}$ is a king in $G_{S,\sigma}$. This, along with the proof above that shows that k_0, k_1, k_2 are the only kings in $G_{S,\sigma}$, immediately implies that k_0, k_1, k_2 are the only three vertices with maximum out-degree in $G_{S,\sigma}$. ◀

4.4 Proof of Theorem 4

We now prove Theorem 4. The upper bounds follow from the arguments in Section 4.1. For the lower bounds, we do a reduction from PMF. The class of tournaments constructed in Section 4.3, and its properties, play a crucial role in the reduction.

[5] We remark here that there is an alternative proof that shows k_0 to be a king: consider an arbitrary j_1 for an arbitrary $j \in S$. The in-neighborhood of j_1 contains S_0 and a subset of $S_1 \cup T_1$. It can be verified that k_0 is a source (and hence a king) in the tournament restricted to the in-neighbourhood of j_1. Lemma 11 then implies that k_0 is a king. We choose to keep the current proof for clarity.

Proof of Theorem 4. The upper bounds follow from Lemma 26 and Lemma 27.

For the lower bounds, consider an input $S \subseteq [n]$ to Alice and $\sigma \in \mathcal{S}_n$ to Bob for PMF_n. Alice and Bob jointly construct the tournament $G_{S,\sigma}$. Note that this construction is completely local and involves no communication; Alice can construct all edges of Types 2, 3 and 4, and Bob can construct all edges of Type 1 (see Figure 1). By Lemma 31, there are exactly 3 kings in $G_{S,\sigma}$ and these are $\left\{i_b : b \in \{0,1,2\}, i = \arg\max_{j \in S} \sigma(j) = \mathsf{PMF}_n(S,\sigma)\right\}$ (recall Definition 7). Thus, running a protocol for KING_{3n} on input $G_{S,\sigma}$ (where Alice has edges of Types 2, 3 and 4, and Bob has edges of Type 1) gives the solution to $\mathsf{PMF}_n(S,\sigma)$ at no additional cost. Lemma 28 implies the required lower bounds. ◀

5 Communication complexity of MOD

Recall that in the MOD_n communication problem, Alice and Bob are given inputs in $\{0,1\}^{E_1}$ and $\{0,1\}^{E_2}$, respectively, where E_1 and E_2 form a partition of the edge set $\binom{n}{2}$. Their goal is to output a vertex v that has maximum out-degree in the tournament formed by the union of their edges. We next prove Theorem 5. In this theorem we settle the communication complexity of finding a maximum out-degree vertex in a tournament in the deterministic, randomized, and quantum models, up to logarithmic factors in the input size. In the deterministic model we are able to show a tight $\Theta(n \log n)$ bound.

We first define an intermediate communication problem, $\mathsf{MAXSUM}_{n,k}$, which seems independently interesting to study from the perspective of communication complexity.

▶ **Definition 32.** *Let $n, k > 0$ be positive integers. In the $\mathsf{MAXSUM}_{n,k}$ problem, Alice is given $A = (a_1, \ldots, a_n) \in [k]^n$, Bob is given $B = (b_1, \ldots, b_n) \in [k]^n$, and their goal is to output $\arg\max_{j \in [n]}(a_j + b_j)$ (if there is a tie, they can output any of the tied indices).*

$\mathsf{MAXSUM}_{n,k}$ is easily seen to be the composition of two problems: the outer problem is $\mathsf{ARGMAX}_{2k,n}$ (see Definition 14) and the inner function is SUM_k (which adds two integers in $[k]$, one with Alice and the other with Bob). It is also easy to see that MOD_n reduces to $\mathsf{MAXSUM}_{n,2n}$: Alice and Bob can locally construct $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ to be the out-degree vectors of all the vertices restricted to edges in their inputs. Thus, a cost-c protocol for $\mathsf{MAXSUM}_{n,2n}$ also gives a protocol for MOD_n.

We note here that our upper bounds (Theorem 5) actually give upper bounds for the more general $\mathsf{MAXSUM}_{n,k}$ problem; the deterministic, randomized, and quantum communication upper bounds here are $O(n \log k), O(n \log\log k)$ and $O(\sqrt{n} \log k \log n)$, respectively. Next, we proceed to give a proof of Theorem 5.

Proof of Theorem 5. For the upper bounds, we exhibit protocols of the required cost for $\mathsf{MAXSUM}_{n,n}$, which is only a (potentially) harder problem.

- For the deterministic upper bound, note that Alice can just send her input to Bob with cost $n \log n$, and Bob can output the answer.
- The randomized upper bound follows by using Theorem 20 with the list $s = (a_1 + b_1, \ldots, a_n + b_n)$, and observing that testing whether $a_i + b_i \geq a_j + b_j$ can be done with communication $O(\log\log n)$ and success probability at least $2/3$ (Theorem 21).
- For the quantum upper bound, recall that $\mathsf{MAXSUM}_{n,n}$ is the composition of $\mathsf{ARGMAX}_{2n,n}$ (with an input list in $[2n]^n$) and SUM (sum of 2 integers in $[n]$, one with Alice and the other with Bob). Here, $\mathsf{ARGMAX}_{2n,n}$ has query complexity $O(\sqrt{n})$, where query access is to the values of the elements of the list (see Theorem 15) and $\mathsf{SUM} : [n] \times [n] \to [2n]$. Setting $\mathcal{D}_g = [n]$, $\mathcal{D}_f = [2n]$, $g = \mathsf{SUM}_n : \mathcal{D}_g \times \mathcal{D}_g \to \mathcal{D}_f$, , $f = \mathsf{ARGMAX}_{2n,n} \subseteq \mathcal{D}_f^n \times [n]$ in Theorem 18, this gives a quantum communication upper bound of $O(\sqrt{n} \log n)$.

Randomized and quantum lower bounds. The randomized and quantum lower bounds follow the same proof as that of Theorem 4 (see Section 4.4) because the three kings in $G_{S,\sigma}$ are precisely the maximum out-degree vertices there as well (see Lemma 31). This argument also shows a deterministic lower bound of $\Omega(n)$.

Deterministic lower bound. We now turn our attention to the deterministic lower bound of $\Omega(n \log n)$, which does not use the same reduction as in the proof of Theorem 4. We show this via a fooling set argument (Lemma 19). Below, we assume that the first half of Alice's input corresponds to the out-degree sequence of a tournament on vertex set $L = \{1, 2, \dots, n/2\}$, the second half of her input corresponds to the out-degree sequence of a tournament on vertex set $R = \{1', 2', \dots, (n/2)'\}$, and Bob's input is the out-degree sequence of the complete bipartite tournament between L and R. We focus on inputs that are induced by tournaments of the following form, that are defined for a permutation $\sigma \in \mathcal{S}_{n/2-1}$ that acts in an identical fashion on $\{2, 3, \dots, n/2\}$ and $\{2', 3', \dots (n/2)'\}$. We call Alice and Bob's input constructed below A_σ and B_σ, respectively.

1. Vertex 1 is the source in L, and vertex $1'$ is the source in R. These edges are with Alice.[6]
2. Vertex 1 has edges towards $1'$ and $\sigma^{-1}(2')$. All other vertices in $\{3', 4', \dots (n/2)'\}$ have edges pointing towards vertex 1. These edges are with Bob.
3. For all $i, j \in \{2, 3, \dots, n/2\}$, there is an edge from i to j iff $\sigma(i) < \sigma(j)$. Similarly there is an edge from i' to j' iff $\sigma(i') < \sigma(j')$. These edges are with Alice.
4. For $i \in \{2, 3, \dots, n/2\}$, there is an edge from i to $1'$. These edges are with Bob.
5. For $i, j \in \{2, 3, \dots, n/2\}$, there is an edge from i to j' iff $\sigma(i) \leq \sigma(j)$. These edges are with Bob.

We now verify that vertex 1 is the unique vertex with maximum out-degree in the whole tournament (and hence the first coordinate must be output in the corresponding inputs to Alice and Bob for MOD_n).

- Items 1 and 2 above ensure that vertex 1 has out-degree $n/2 - 1 + 2 = n/2 + 1$.
- Item 1 and Item 4 ensure that the out-degree of vertex $1'$ is $n/2 - 1$.
- Item 1 and Item 5 ensure that vertex $\sigma^{-1}(2')$ has out-degree $n/2 - 2$.
- For $i \in \{2, 3, \dots, n/2\}$, the out-degree of vertex $\sigma^{-1}(i)$ is $n/2 - i$ from Alice's input (Item 3) plus i from Bob's input (Item 5), which gives a total of $n/2$.
- For $i \in \{3, 4, \dots, n/2\}$, the out-degree of vertex $\sigma^{-1}(i')$ is $n/2 - i$ from Alice's input (Item 3) plus $i - 1$ from Bob's input (Item 5), which gives a total of $n/2 - 1$.

These bullets verify that for input (A_σ, B_σ), vertex 1 is the unique maximum out-degree vertex. Our fooling set will be of the form $F = \{(A_\sigma, B_\sigma) : \sigma \in S\}$, where $S \subseteq \mathcal{S}_{n/2-1}$ is chosen appropriately. The property that S will satisfy is that for all $\sigma \neq \sigma' \in S$, at least one of the inputs $(A_\sigma, B_{\sigma'})$ or $(A_{\sigma'}, B_\sigma)$ will *not* have vertex 1 as a maximum out-degree vertex. We will also construct S such that $|S| = 2^{\Omega(n \log n)}$. Lemma 19 will then imply the required deterministic communication lower bound of $\Omega(n \log n)$.

It remains to construct $S \subseteq \mathcal{S}_{n/2-1}$, which we do in the remaining part of this proof. We construct S such that it satisfies the following property.

$$\forall \sigma \neq \sigma' \in S, \qquad \exists i \in \{2, 3, \dots, n/2\} : |\sigma(i) - \sigma'(i)| \geq 2.$$

[6] When we say "edges are with Alice/Bob", we actually mean Alice/Bob's out-degree of vertices is determined by the directions of the underlying edges. In this case we mean Alice's first coordinate is $n/2 + 1$ because vertex 1 is a source in L.

In the two bullets below, we first show why such an S satisfies the required fooling set property, and then show a construction of S of size $2^{\Omega(n \log n)}$.

- Let $\sigma \neq \sigma'$ be an arbitrary pair of elements of S. Without loss of generality, assume that $i \in \{2, 3, \ldots, n/2\}$ is such that $\sigma'(i) - \sigma(i) \geq 2$ (otherwise switch the roles of σ and σ' and run the same argument). Consider the input $(A_\sigma, B_{\sigma'})$. Note that the out-degree of vertex 1 remains $n/2 + 1$ because all edges incident on it are fixed for all inputs in our fooling set. Alice's contribution to the out-degree of vertex i is $n/2 - \sigma(i)$, and Bob's contribution is $\sigma'(i)$, which gives a total of $n/2 + \sigma'(i) - \sigma(i) \geq n/2 + 2$. Thus vertex 1 cannot be a maximum out-degree vertex in the input $(A_\sigma, B_{\sigma'})$.
- We construct such an S greedily one element at a time. At any step in the construction we maintain the invariant that the current set T satisfies

  $$\forall \sigma \neq \sigma' \in T, \qquad \exists i \in \{2, 3, \ldots, n/2\} : |\sigma(i) - \sigma'(i)| \geq 2.$$

  Additionally we maintain a "candidate" set of permutations in $\mathcal{S}_{n/2-1}$ that are not in T, and have the property that adding any of them to T will satisfy T's invariant. Initially we start with $T = \emptyset$ and the candidate set as $\mathcal{S}_{n/2-1}$, which clearly satisfies the required invariant. At any stage, after adding σ to T, we remove the set S_σ from the candidate set, where S_σ is defined as

  $$S_\sigma := \left\{\tau \in \mathcal{S}_{n/2-1} : |\tau(i) - \sigma(i)| < 2\right\} \ \forall i \in \{2, 3, \ldots, n/2\}.$$

  It is easy to verify by induction that T and the candidate set thus constructed always satisfy the required invariant. The initial size of the candidate set is $(n/2-1)! = 2^{\Omega(n \log n)}$, and at each step we are removing at most 3^n elements from the candidate set. This means that the number of iterations of this construction is at least $2^{\Omega(n \log n - n)} = 2^{\Omega(n \log n)}$, which is what we needed. ◀

We remark that while it may seem like the argument used in the previous proof may be adaptable to prove a deterministic communication lower bound of $\Omega(n \log n)$ for KING_n, this is not possible in view of our $O(n)$ deterministic communication upper bound for KING_n from Theorem 4. This shows an inherent difference between MOD_n and KING_n in the setting of deterministic communication complexity.

▶ Remark 33. We note that our $\Omega(n \log n)$ lower bound for MOD_n also solves Problem 2 in [22].

References

1 Scott Aaronson and Andris Ambainis. Quantum search of spatial regions. *Theory Comput.*, 1(1):47–79, 2005. doi:10.4086/TOC.2005.V001A004.

2 Miklós Ajtai, Vitaly Feldman, Avinatan Hassidim, and Jelani Nelson. Sorting and selection with imprecise comparisons. *ACM Trans. Algorithms*, 12(2):19:1–19:19, 2016. doi:10.1145/2701427.

3 Kazuyuki Amano. Some improved bounds on communication complexity via new decomposition of cliques. *Discrete Applied Mathematics*, 166:249–254, 2014. doi:10.1016/j.dam.2013.09.015.

4 Anurag Anshu, Naresh Goud Boddu, and Dave Touchette. Quantum log-approximate-rank conjecture is also false. In *Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 982–994, 2019. doi:10.1109/FOCS.2019.00063.

5 László Babai, Peter Frankl, and Janos Simon. Complexity classes in communication complexity theory (preliminary version). In *Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 337–347, 1986. doi:10.1109/SFCS.1986.15.

6 Yakov Babichenko, Shahar Dobzinski, and Noam Nisan. The communication complexity of local search. In *Symposium on Theory of Computing, (STOC)*, pages 650–661, 2019. doi:10.1145/3313276.3316354.

7 R. Balasubramanian, Venkatesh Raman, and G. Srinivasaragavan. Finding scores in tournaments. *J. Algorithms*, 24(2):380–394, 1997. doi:10.1006/JAGM.1997.0865.

8 Kaspars Balodis, Shalev Ben-David, Mika Göös, Siddhartha Jain, and Robin Kothari. Unambiguous DNFs and Alon-Saks-Seymour. In *Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 116–124. IEEE, 2021. doi:10.1109/FOCS52979.2021.00020.

9 Gal Beniamini and Noam Nisan. Bipartite perfect matching as a real polynomial. In *Symposium on Theory of Computing, (STOC)*, pages 1118–1131. ACM, 2021. doi:10.1145/3406325.3451002.

10 Arindam Biswas, Varunkumar Jayapaul, Venkatesh Raman, and Srinivasa Rao Satti. Finding kings in tournaments. *Discret. Appl. Math.*, 322:240–252, 2022. doi:10.1016/j.dam.2022.08.014.

11 Joakim Blikstad, Jan van den Brand, Yuval Efron, Sagnik Mukhopadhyay, and Danupon Nanongkai. Nearly optimal communication and query complexity of bipartite matching. In *Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 1174–1185, 2022. doi:10.1109/FOCS54457.2022.00113.

12 Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation. In *Symposium on the Theory of Computing, (STOC)*, pages 63–68, 1998. doi:10.1145/276698.276713.

13 Amit Chakrabarti, Prantar Ghosh, Andrew McGregor, and Sofya Vorotnikova. Vertex ordering problems in directed graph streams. In *Symposium on Discrete Algorithms, (SODA)*, pages 1786–1802, 2020. doi:10.1137/1.9781611975994.109.

14 Arkadev Chattopadhyay, Yogesh Dahiya, Nikhil S. Mande, Jaikumar Radhakrishnan, and Swagato Sanyal. Randomized versus deterministic decision tree size. In *Symposium on Theory of Computing, (STOC)*, pages 867–880, 2023. doi:10.1145/3564246.3585199.

15 Arkadev Chattopadhyay, Nikhil S. Mande, and Suhail Sherif. The log-approximate-rank conjecture is false. *J. ACM*, 67(4):23:1–23:28, 2020. doi:10.1145/3396695.

16 Arjan Cornelissen, Nikhil S. Mande, and Subhasree Patro. Improved quantum query upper bounds based on classical decision trees. In *Foundations of Software Technology and Theoretical Computer Science, (FSTTCS)*, volume 250, pages 15:1–15:22, 2022. doi:10.4230/LIPICS.FSTTCS.2022.15.

17 Yogesh Dahiya and Meena Mahajan. On (simple) decision tree rank. *Theor. Comput. Sci.*, 978:114177, 2023. doi:10.1016/J.TCS.2023.114177.

18 Palash Dey. Query complexity of tournament solutions. In *Conference on Artificial Intelligence, (AAAI)*, pages 2992–2998, 2017. doi:10.1609/AAAI.V31I1.10702.

19 Pavol Duris and Pavel Pudlák. On the communication complexity of planarity. In *Fundamentals of Computation Theory, (FCT)*, volume 380 of *Lecture Notes in Computer Science*, pages 145–147, 1989. doi:10.1007/3-540-51498-8_14.

20 Christoph Dürr and Peter Høyer. A quantum algorithm for finding the minimum. *CoRR*, quant-ph/9607014, 1996. URL: http://arxiv.org/abs/quant-ph/9607014.

21 Uriel Feige, David Peleg, Prabhakar Raghavan, and Eli Upfal. Computing with unreliable information (preliminary version). In *Symposium on Theory of Computing, (STOC)*, pages 128–137. ACM, 1990. doi:10.1145/100216.100230.

22 Maxime Flin and Parth Mittal. $(\Delta+1)$ vertex coloring in $O(n)$ communication. In *Proceedings of the 43rd ACM Symposium on Principles of Distributed Computing, PODC 2024, Nantes, France, June 17-21, 2024*, pages 416–424. ACM, 2024. doi:10.1145/3662158.3662796.

23 Prantar Ghosh. Private Communication, 2024.

24 Prantar Ghosh and Sahil Kuchlous. New algorithms and lower bounds for streaming tournaments. *CoRR*, abs/2405.05952, 2024. doi:10.48550/arxiv.2405.05952.

25 Mika Göös. Lower bounds for clique vs. independent set. In *Symposium on Foundations of Computer Science, (FOCS)*, pages 1066–1076, 2015. doi:10.1109/FOCS.2015.69.

26 Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM J. Comput.*, 47(6):2435–2450, 2018. doi:10.1137/16M1059369.

27 András Hajnal, Wolfgang Maass, and György Turán. On the communication complexity of graph properties. In *Symposium on Theory of Computing, (STOC)*, pages 186–191, 1988. doi:10.1145/62212.62228.

28 Hao Huang and Benny Sudakov. A counterexample to the Alon-Saks-Seymour conjecture and related problems. *Comb.*, 32(2):205–219, 2012. doi:10.1007/S00493-012-2746-4.

29 Gábor Ivanyos, Hartmut Klauck, Troy Lee, Miklos Santha, and Ronald de Wolf. New bounds on the classical and quantum communication complexity of some graph properties. In *Foundations of Software Technology and Theoretical Computer Science, (FSTTCS)*, volume 18, pages 148–159, 2012. doi:10.4230/LIPICS.FSTTCS.2012.148.

30 Bala Kalyanasundaram and Georg Schnitger. The probabilistic communication complexity of set intersection. *SIAM J. Discret. Math.*, 5(4):545–557, 1992. doi:10.1137/0405044.

31 Eyal Kushilevitz, Nathan Linial, and Rafail Ostrovsky. The Linear-Array Conjecture in Communication Complexity is False. *Comb.*, 19(2):241–254, 1999. doi:10.1007/S004930050054.

32 Eyal Kushilevitz and Noam Nisan. *Communication Complexity.* Cambridge University Press, 1997.

33 Oded Lachish, Felix Reidl, and Chhaya Trehan. When you come at the king you best not miss. In *Foundations of Software Technology and Theoretical Computer Science, (FSTTCS)*, volume 250, pages 25:1–25:12, 2022. doi:10.4230/LIPIcs.FSTTCS.2022.25.

34 H.G. Landau. On dominance relations and the structure of animal societies: III The condition for a score structure. *The bulletin of mathematical biophysics*, 15:143–148, 1953. doi:10.1007/BF02476378.

35 László Lovász and Michael E. Saks. Lattices, mobius functions and communications complexity. In *Symposium on Foundations of Computer Science, FOCS*, pages 81–90, 1988. doi:10.1109/SFCS.1988.21924.

36 Nikhil S Mande, Manaswi Paraashar, Swagato Sanyal, and Nitin Saurabh. On the communication complexity of finding a king in a tournament. *CoRR*, 2024. arXiv:2402.14751.

37 Nikhil S. Mande, Manaswi Paraashar, and Nitin Saurabh. Randomized and quantum query complexities of finding a king in a tournament. In *Foundations of Software Technology and Theoretical Computer Science, (FSTTCS)*, volume 284, pages 30:1–30:19, 2023. doi:10.4230/LIPICS.FSTTCS.2023.30.

38 Stephen B Maurer. The king chicken theorems. *Mathematics Magazine*, 53(2):67–80, 1980.

39 Noam Nisan. The communication complexity of threshold gates. *Combinatorics, Paul Erdos is Eighty*, 1:301–315, 1993.

40 Noam Nisan. The demand query model for bipartite matching. In *Symposium on Discrete Algorithms, (SODA)*, pages 592–599, 2021. doi:10.1137/1.9781611976465.36.

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

42 Alexander Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya of the Russian Academy of Sciences, mathematics*, 67(1):159–176, 2003.

43 Alexander A. Razborov. On the distributional complexity of disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992. doi:10.1016/0304-3975(92)90260-M.

44 Jian Shen, Li Sheng, and Jie Wu. Searching for sorted sequences of kings in tournaments. *SIAM J. Comput.*, 32(5):1201–1209, 2003. doi:10.1137/S0097539702410053.

45 Manami Shigeta and Kazuyuki Amano. Ordered biclique partitions and communication complexity problems. *Discrete Applied Mathematics*, 184:248–252, 2015. doi:10.1016/j.dam.2014.10.029.

46 Makrand Sinha and Ronald de Wolf. Exponential separation between quantum communication and logarithm of approximate rank. In *Annual Symposium on Foundations of Computer Science, (FOCS)*, pages 966–981, 2019. doi:10.1109/FOCS.2019.00062.

47 Mihalis Yannakakis. Expressing combinatorial optimization problems by linear programs. *J. Comput. Syst. Sci.*, 43(3):441–466, 1991. doi:10.1016/0022-0000(91)90024-Y.